International Journal of Signal Processing Systems Vol. 1, No. 2 December 2013

Music Similarity Analysis through Repetitions and Instantaneous Frequency Spectrum

Tatiana Endrjukaite and Yasushi Kiyoki
Keio University, Graduate School of Media and Governance, Fujisawa, Japan
E-mail: tatiana@sfc.keio.ac.jp, kiyoki@sfc.keio.ac.jp

***Abstract*—This paper presents a new approach for tunes similarity calculation based on repetitions. Information about repetitions in tunes is important since repetitions make very significant impression to listener. We are providing a way to describe tunes in descriptors which contain frequency information about repetitions of tunes. Frequency information is retrieved by means of a relatively new signal processing approach called instantaneous frequency spectrum (IFS). Further tunes comparison takes this repetitions frequencies information from one tune descriptor and compares to second tune's descriptor. As a result we obtain the similarity between compared tunes. We show that proposed approach gives meaningful information about music pieces similarity and it can successfully be used in music signal processing tasks.**

***Index Terms*—music similarity, repetitions, instantaneous frequency spectrum (IFS), empirical mode decomposition (EMD), Hilbert transforms (HT)**

I. Introduction

Very large number of music pieces exist in the world today. People listen to different genres of music, such as classical, popular, jazz, blues and others. It was observed, that people usually listen to the same kind of music they prefer. For example, same artist usually performs his singles in the same manner, mood, and musical genre. Although it is well established that people respond emotionally to music, little is known about precisely what it is in the music that they are responding to [1].

To find a new tune we will like, we have to listen to this music piece to find out whether we will like it or not. As soon as various structures in music influence on us, we have to consider something more than just separate characteristics such as tonality, timbre, speed, or pitch. Repetitive parts in music pieces contain the most representative and significant information about the tune. Generally speaking, the more repetitions and similar phases there are in a piece of music, the easier it is for people to have affinity for it. For example, in the popular music style choruses are repeated several times in the tune. That makes it possible to easily remember the tune and recall it from the memory when we want to listen something similar.

The ultimate goal of this research is to find and presents similarity of music tunes by means of repeated parts in it. That will help people to find music pieces they may be even do not know, but most probably will like. Such tunes can have different tonality and may be even have different genre, but sound very similar to tunes the person prefers to listen. Finding the similarity between music pieces contribute to the task of playlist suggestion for listeners.

However, there is no complete theory for music structure automatic analysis. At present there are encouraging researches on detecting the most frequently appearing component in a piece of music based on music structure analysis. Researches have made "music thumbnails" and "audio summarization" by detecting the most representative part of a piece of music. Some researchers have found repetitions by performing self-similarity calculations with Mel-Frequency Cepstral Coefficients (MFCCs) [2] and [3]; others have identified them based on approximate transcription results.

This paper is organized as follows: Section II describes main steps of the processing method. Section III presents tunes comparison approach. Section IV describes experimental results and discussions based on obtained outcomes. Section V summarizes the paper.

II. Approach

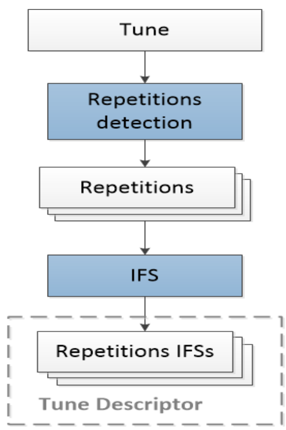

Figure 1. Tune's descriptor calculation.

The idea of music processing to get a tune descriptor is outlined in Fig. 1. White boxes show data and blue boxes show processing steps. Acoustic data is used as an input.

Manuscript received August 28, 2013; revised November 19, 2013.

doi: 10.12720/ijsps.1.2.170-176

We create a tune descriptor by using repetitions within tune. These repetitions are processed by means of empirical mode decomposition [4] and splits into a number of intrinsic functions. These functions are used to get an instantaneous frequency spectrum [5].

IFS spectrums of repetitions make the tune descriptor. To find similarity between two tunes, their descriptors spectrums are compared to each other by using sum of squared differences of every spectrum part. As a result, we get a similarity between two music pieces. We use PCM data in the mono WAV format with a 44100 Hz sampling and 16-bit quantization as input data.

A. Repetitions

Information regarding repetitions in a piece of music is important since it is related to affinity for music, though of course the relationship between repetitions and affinity varies from person to person.

In this work, we carefully selected repetitions within music pieces by music notes to create tunes descriptors. Those repetitions were then processed with IFS to create a descriptor for the considered tune. This idea is presented in Fig. 2.

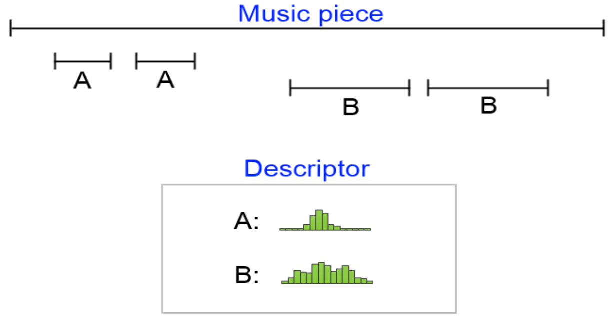

Figure 2. Tune repetitions and descriptor.

B. Empirical Mode Decomposition

The EMD is a way to decompose a signal into so-called intrinsic mode functions (IMFs) [6]. Since the decomposition is based on the local characteristic time scale of the data, it can be applied to nonlinear and non-stationary processes.

To extract IMFs from the signal *X(t)*, all local extrema should be found first. Then we should create an upper envelope $e_u(t)$ by local maxima and a lower envelope $e_l(t)$ by local minima. Envelopes are built by cube-spline interpolation. Using the upper and lower envelopes, the mean *m(t)* is calculated as (1). The result is shown in Fig. 3.

$$m(t) = \frac{e_u(t)+e_l(t)}{2} \quad (1)$$

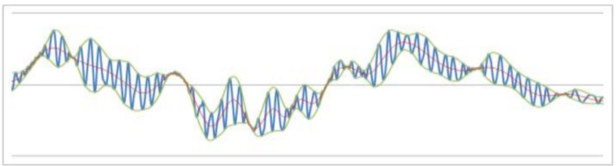

Figure 3. Signal (blue), its envelopes (green) and mean (red) by envelopes.

The difference between the data and *m(t)* is the first component $h_1(t)$, which represents *proto IMF*. An IMF is defined as a function that satisfies two requirements: First, the number of extrema and the number of zero-crossings must either be equal or differ at most by one. Second, at any point, the mean value of the envelope defined by the local maxima and the envelope defined by the local minima is zero.

Until $h_1(t)$ does not satisfy the definition of the IMF mentioned above, it should be iteratively refined using the same procedure. Thereby for $h_1(t)$ we get next component $h_2(t)$ and then $h_3(t)$ and so on until stop criteria (2) becomes true, where **ε** is a small number. In this work **ε** was set to *0.0001*.

$$\frac{\sum_t (h_k(t)-h_{k-1}(t))^2}{\sum_t (h_{k-1}(t))^2} < \varepsilon \quad (2)$$

After repeated refinement up to *k* times, $h_k(t)$ becomes the first IMF of the signal, called $c_1(t)$. Fig. 4 shows the first obtained IMF.

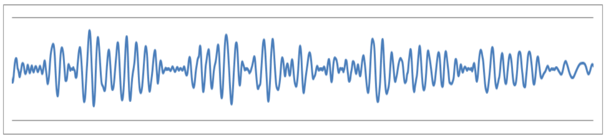

Figure 4. First IMF obtained from the signal.

By subtracting $c_1(t)$ from initial data we get the residue *r(t)*, as shown in (3) and in Fig. 5.

$$r(t) = X(t) - c_1(t) \quad (3)$$

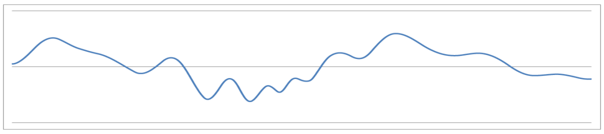

Figure 5. Residue after subtracting first IMF c_1.

In the next round of the sifting process the residue *r(t)* is considered as a signal *X(t)* and the sifting procedure is repeated the same way to obtain $c_2(t)$, then $c_3(t)$, and so on until residue becomes a monotonic function without extrema. When we sum all obtained IMFs with the last residue, we get initial data signal as (4).

$$X(t) = \sum_{i=1}^{n} c_i + r_n .... \quad .(4)$$

The good feature of such decomposition is that each IMF represents an intrinsic component of the real physical effect. Fig. 6 shows the original signal and IMFs obtained by means of EMD.

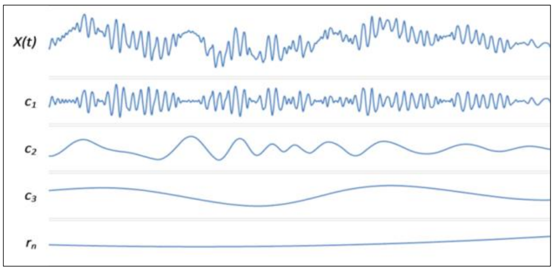

Figure 6. The resulting empirical mode decomposition components from the music data: the original data X(t) and the components c1 – c3; rn is a trend.

C. Hilbert Transform

The Hilbert transform can be interpreted as a phase shifter, which changes the phase of all frequency components of a signal to π/2. To shift a phase, the initial signal is processed with a Fourier transform (FT) and then every component of the resultant spectrum is multiplied by imaginary *i* and the spectrum is converted back to signal using the inverse Fourier transform (IFT).

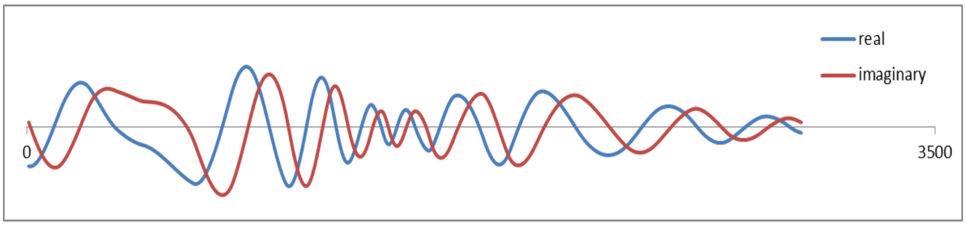

Figure 7. Initial signal (blue) and obtained imaginary signal after HT (red).

An example of the original signal and derived signal with shifted phase are shown in Fig. 7.

The imaginary signal $\widetilde{X}(t)$ is orthogonal to original signal $X(t)$. This feature allows us to develop from $\widetilde{X}(t)$ and $X(t)$ a complex analytical signal $H(t)$ (5). $H(t)$ is described as a vector on the complex plane where $X(t)$ and $\widetilde{X}(t)$ are projections to real and imaginary axes, respectively.

$$H(t) = X(t) + i\widetilde{X}(t) \tag{5}$$

The advantage of this representation is that we have an opportunity to determine instantaneous parameters of the signal $H(t)$, i.e., the amplitude and frequency, where the radius of each circle represents the amplitude and the space between circles means the frequency.

Instantaneous amplitude is calculated as complex number length in (6).

$$A(t) = \sqrt{(realH(t))^2 + (imagH(t))^2} \tag{6}$$

Instantaneous frequency is calculated as instantaneous phase derivative of a signal (7). Where phase φ is calculated as (8)

$$f(t) = \frac{1}{2\pi}\varphi'(t) \tag{7}$$

$$\varphi(t) = \tan^{-1}\frac{realH(t)}{imagH(t)} \tag{8}$$

D. Instantaneous Frequency Spectrum

The IFS calculation method is outlined in the Fig. 8. As inputs, we use a number of IMFs that represent intrinsic functions of the same signal. White boxes show data and blue boxes show processing steps. As an output, we get a histogram of amplitudes by frequencies. For each IMF, we get instantaneous frequencies and instantaneous amplitudes using the Hilbert transform. These frequencies and amplitudes are used to create a histogram. Formally, this is described in (9.1) - (9.2).

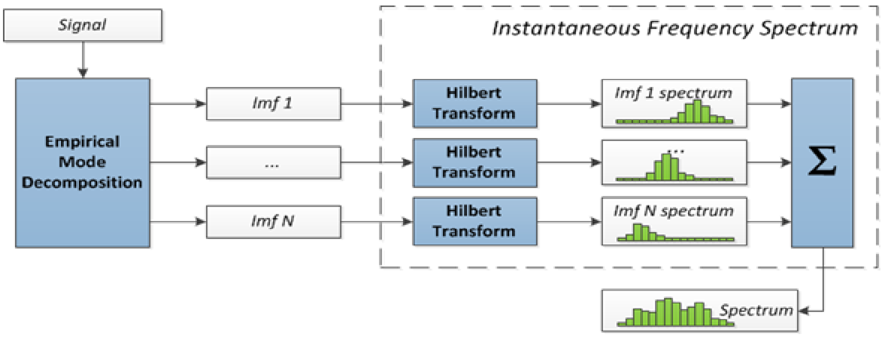

Figure 8. Scheme for calculating the IFS.

$$b_i = \sum_t A(t), \; t: \beta(i-1) \le \log f(t) \le \beta(i) \tag{9.1}$$
$$i = \overline{1, N}$$

$$\beta(i) \stackrel{\text{def}}{=} \frac{i}{N}\log F_{max} \tag{9.2}$$

where b_i is height of *i*-th bar of the histogram,

$A(t)$ is an instantaneous amplitude at time *t*,

$f(t)$ is an instantaneous frequency at time *t*,

$\beta(i)$ is a frequency upper boundary for *i*-th histogram bar,

N is a number of bars in the histogram,

F_{max} is maximal frequency.

For this paper, a *100*-bar histogram was used ($N = 100$) with maximal frequency of *20* kHz ($F_{max} = 20\,000$).

III. Tunes Comparison

To compare two tunes we have to compare every repetition of first tune to every repetition of second tune. Those pairwise repetitions comparison results have to be considered to calculate the tunes difference. Schematically it is displayed on Fig. 9.

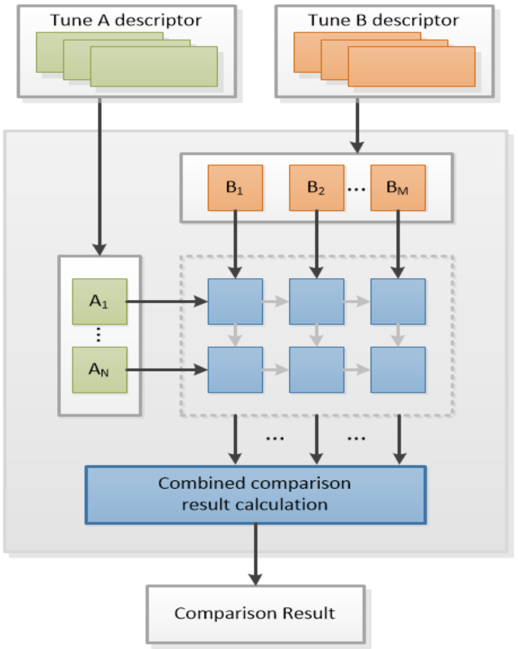

Figure 9. Tunes comparison method.

There are different approaches to combine multiple results from repetitions comparison into one single number for tunes difference. We give three approaches how it can be done. Hereafter we reference them as AVG, MIN and AVG_k.

AVG approach requires taking average of all repetition comparison as shown in (10).

$$AVG(A,B) = \frac{1}{N*M}\sum_{i=1}^{N}\sum_{j=1}^{M} comp(A_i,B_j) \quad (10)$$

where $comp(A_i,B_j)$ is a comparison result for i-th repetition from tune A and j-th repetition from tune B.

MIN approach requires taking the smallest value, which means that tunes become as different as their most similar repetitions difference is. Equation (11) shows the way how to calculate it.

$$MIN(A,B) = \min_{\substack{i=1..N \\ j=1..M}} comp(A_i,B_j) \quad (11)$$

AVG_k approach requires taking average of k smallest values from all repetition pairs comparisons. k should be calculated from tunes repetitions count as in (12).

$$AVG_k(A,B) = \frac{1}{k}\sum_{p=1}^{k} ord\{x, x = comp(A_i,B_j), i = \overline{1,N}, j = \overline{1,M}\}_p, \quad k = int\left(\sqrt{N*M}\right) \quad (12)$$

where k is the number of elements we take into account; $ord\{x\}_p$ is a p-th element of set of x elements ordered in ascending order. In other words, we take k minimal elements from all items.

A. Two Different Tunes Comparison

Here we consider example of two tunes comparison: Tune A and Tune B. Both have three repetitions. Let's assume that we compared all repetitions of those tunes. Fig. 10 shows the results of those comparisons.

		Tune A		
		Rep 1	Rep 2	Rep 3
Tune B	Rep 1	3.0	1.0	2.0
	Rep 2	2.5	2.0	3.5
	Rep 3	2.0	1.0	1.5

Figure 10. Two different tunes comparison example.

TABLE I. METHODS RESULTS COMPARISON

Type	Result	Comment
AVG	2.06	All information is taken into account
MIN	1.00	Too few information is considered
AVG_k	1.17	Part of information is considered

In Table I methods results comparison are represented. For different tunes comparison AVG method looks as the best because it uses all of the repetitions data to give the overall tunes similarity result. MIN approach for the same data gives the minimal value, which means that only one repetitions pair comparison is used, even when there are lots of repetitions in tunes being compared, it is too few to give a comprehansive result. AVG_k approach uses more than one pair of repetitions comparison results, but not all of them.

B. Tune Comparison to Itself

Here we consider another example when tune A is compared to itself. Fig. 11 shows the results of those comparisons. In Table II methods results comparison three methods results comparison are represented.

		Tune A		
		Rep 1	Rep 2	Rep 3
Tune A	Rep 1	0.0	3.0	2.0
	Rep 2	3.0	0.0	4.0
	Rep 3	2.0	4.0	0.0

Figure 11. Tune comparison to itself example.

TABLE II. METHODS RESULTS COMPARISON

Type	Result	Comment
AVG	2.00	INACCEPTABLE
MIN	0.00	OK
AVG_k	0.00	OK

When tune is compared to itself, AVG method gives a non-zero result which is inacceptable, because tune is always maximally similar to itself, so the difference must be zero. MIN approach always will give zero value because values on main diagonal always equal to zero. AVG_k approach also always gives zero because k elements that are being taken into account are exactly all zeros from main diagonal.

As a conclusion we can say that AVG_k approach is the best among described three approaches because it behaves as a trade-off, trying to consider as much repetitions comparisons as possible, and not breaking the requirement of zero difference for tune comparison to itself.

IV. Experiments and Discussions

We selected seven famous classical music pieces for our experiments. Table III shows the selected tunes and their short names. We use the short names hereafter in this paper. Seven pieces of music were processed with the method described in this paper.

TABLE III. Target Music Pieces Used in the Work

Name	Description
V	Four Seasons: Summer 3rd movement by Vivaldi, [7, 8]
B	Air on G string Orchestral Suite No3 in D major by Bach, [9]
C	Etude Op. 25 No. 11 by Chopin, [10, 11]
S	Gnossienne 4 by Satie, [12, 13]
M	Meditation for Thais by Massenet, [14]
AH	Bugler's Holiday by Anderson, [15]
AP	Plink Plank Plunk by Anderson, [16]

Table IV shows the repetitions in the musics scores we used to calculate tunes descriptors for terget music pieces and their performances time in CD. For example, for V tune the number of bars and the performance times of each part are as follows: from the 10th to 17th bar and from the 101th to 108th bar are the same. Performed from 12.5 to 22.0 seconds and from 122.0 to 132.0 seconds accordingly.

TABLE IV. Repetitions in Target Music Pieces

Tune	Nr.	Repetition (bar)	Performance (seconds)
V	1	10 – 17; 101 – 108	12.5 – 22.0; 122.0 – 132.0
B	1	1 – 6; 7 – 12	0 – 43.2; 47.2 – 90.1
	2	13 –24; 25 –36	94.6 – 185.0; 186.5 – 278.3
C	1	5 – 7.5; 13 – 15.5; 69 – 71.5; 77 – 79.5; 23 – 25.5; 31 – 33.5	23.9 – 30.5; 38.6 – 46.5; 151.6 – 159.1; 167.8 – 176.0; 58.5 – 66.5; 75.0 – 83.0
	2	5 – 15.5; 69 – 79.5	23.9 – 46.5; 151.6 – 176.0
S	1	11 – 12; 24 – 25	58.1 – 68.6; 133.4 – 146.7
	2	13; 15	69.2 – 74.5; 80.6 – 85.7
	3	26 – 27; 28 – 29	147.4 – 159.2; 160.0 – 172.7
	4	19 – 20; 31 – 32	102.5 – 114.4; 180.1 – 195.4
	5	14; 17	74.5 – 79.7; 91.3 – 96.8
	6	18–20; 30–32	97.1 – 113.2; 173.6 – 195.1
M	1	3 – 10; 40 – 47	11.3 – 47.1; 174.5 – 211.1
	2	15 – 20.5; 52 – 57.5	63.9 – 96.7; 229.9 – 258.8
	3	3 – 4; 11 – 12; 40 – 41	11.3 – 19.0; 47.8 – 56.4; 175.1 – 182.1
AH	1	9 – 22.5; 35 – 38.5	5.7 – 16.1; 16.4 – 28
	2	59 – 74; 141 – 156	43.0 – 54.6; 104.0 – 116
	3	75 – 88; 157 – 170	54.8 – 64.8; 116.3 – 126.3
	4	97 – 104; 113 – 120	70.1 – 75.7; 83.0 – 87.9
AP	1	4.5 – 19; 20.5 – 19; 37.5 – 52; 37.5 – 52; 75.5 – 90; 107.5 – 122; 123.5 – 122	3.1 – 15.4; 15.9 – 27.2; 40.1 – 50.7; 62.4 – 72.9; 99.6 – 110.5; 122.0 – 133.4; 133.5 – 143.1
	2	21 – 36; 53 – 36; 91 – 106	27.4 – 38.9; 51.0 – 62.0; 110.9 – 121.7
	3	55 – 70; 55 – 70	74.0 – 85.0; 85.2 – 96.4

Fig. 12 represents the color scale, where red color means dissimilarity and green color means high similarity. Fig. 13 shows repetition to repetition comparison results in color.

Figure 12. Color scale.

As expected, the main diagonal is green because every repetition is maximally similar to itself. By solid lines we separate tunes. Dotted lines separate repetitions within tune.

In Fig. 13 we can see repetitions comparison between tunes, as well as within tunes. We may notice that some music pieces have dissimilar repetitions within. For example, it can be seen inside (6x6) cells rectangle for S tune comparison to itself: there are red and yellow cells. For other tunes comparisons to themselves we mainly see green cells, that means that repetititons within them are very similar.

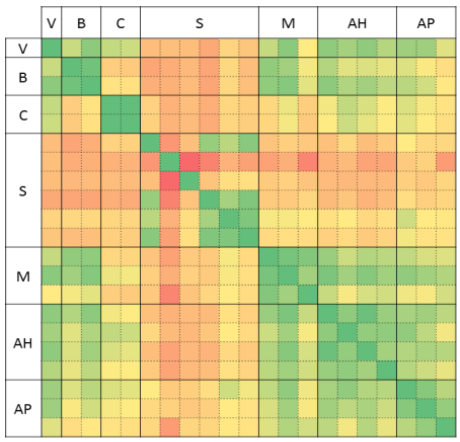

Figure 13. Tunes descriptors comparison.

According to comparison between two different tunes, we can see various results. Some rectangles mostly consist of green cells meaning that tunes are alike, some consist mostly of yellow and red cells meaning tunes dissimilarity.

A. Tunes Comparison Results

To get overall result for tunes comarison we consider all cells within every rectangle and calculate a single difference by means of AVG_k method. Those tune to tune similarity results are shown in Table V and it's colored presentation is shown in Fig. 14.

Main diagonal is green meaning that every tune is similar to itself. Tune S is very different from all other tunes, as we can see by red cells on row and line that correspond to that tune. S tune is very slow and contains many silent fragments and pauses. This fact makes that tune very different from all other tunes.

The same as S, tune C is also somehow different to most of the tunes. This etude is performed by piano and the tempo is very fast. Such piano recording sounds not very similar to orchestral performances as in B and M. But it is somehow similar to V and AH, as they sound similar because of the speed. Tunes comparison values are 0.0029 and 0.0030 correspondigly, and displayed as yellow cells.

Music pieces V, B, M, AH, and AP are very similar to each other as we can see from comparison values that are between 0.0012 and 0.0019, displayed as green cells. These tunes sound similar because they are played by orchestra and by quartet, they have similar energetic mood. The only exception is B to AP pair (0.0029) that sounds just somehow similar because of performance mood and small speed difference.

TABLE V. COMPARISON RESULTS

	V	B	C	S	M	AH	AP
V	0.0000	0.0013	0.0029	0.0082	0.0012	0.0013	0.0018
B	0.0013	0.0000	0.0057	0.0080	0.0012	0.0016	0.0029
C	0.0029	0.0057	0.0000	0.0073	0.0042	0.0030	0.0037
S	0.0082	0.0080	0.0073	0.0000	0.0049	0.0058	0.0042
M	0.0012	0.0012	0.0042	0.0049	0.0000	0.0012	0.0019
AH	0.0013	0.0016	0.0030	0.0058	0.0012	0.0000	0.0018
AP	0.0018	0.0029	0.0037	0.0042	0.0019	0.0018	0.0000

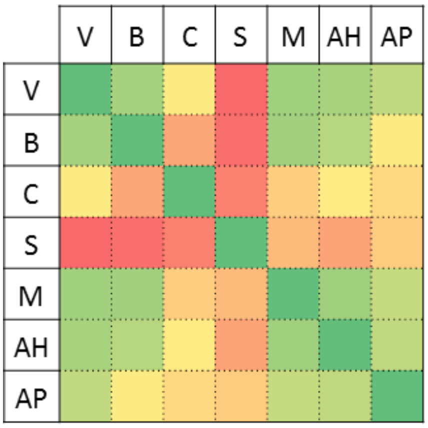

Figure 14. Target music analysis similarity result.

V. CONCLUSIONS AND FUTURE WORK

In this work, we proposed an approach for tunes comparison based on repetitions. Information regarding repetitions in a piece of music is important since repetitions make very significant impression while tunes are listened by a human. We provided the way to describe tunes in descriptors which contain frequency information about repetitions of tunes. Frequency information is retrieved by means of a relatively new signal processing approach called instantaneous frequency spectrum. Further tunes comparison takes this repetitions frequencies information from one tune descriptor and compares to second tune's descriptor. As a result we obtain the similarity between compared tunes. We used seven famous classical music pieces for experiments. Results of our experiments show that proposed approach gives meaningful information about music pieces similarity and it can successfully be used in music signal processing tasks, such as preparing a playlist of suggested similar musical records, based on recently played tunes.

In future work we are planning to include repetitions lengths to tune's descriptor and use them as weights while calculating tunes similarity. It will prioritize repetitions, giving more attention for longer and more frequent repetitions.

REFERENCES

[1] J. Sloboda, *Exploring the Musical Mind: Cognition, Emotion, Ability, Function*, Oxford University Press, 2005, pp. 468.

[2] J. Foote, "Visualizing music and audio using self-similarity," in *Proc. ACM Multimedia '99 SIGIR'03*, 1999, pp. 77-80.

[3] B. Logan, "Mel frequency cepstral coefficients for music modeling," presented at 1st International Symposium on Music Information Retrieval, Plymouth, Massachusetts, October 23-25, 2000.

[4] N. Huang, *et al.,* "The empirical mode decomposition and the Hilbert spectrum for nonlinear and non-stationary time series analysis," in *Proc. R. Soc. Lond. A*, March 1998, pp. 903-995.

[5] T. Endrjukaite and N. Kosugi, "Time-dependent genre recognition by means of instantaneous frequency spectrum based on Hilbert-Huang transform," in *Proc. 14th International Conference on Information Integration and Web-Based Applications & Services*, 2012, pp. 64-71.

[6] C. Zong, "Hilbert-Huang transform based physiological signals analysis for emotion recognition," in *Proc. IEEE International Symposium on Signal Processing and Information Technology.* 2009, pp. 334-339.

[7] Vivaldi. Four seasons. Pina Carmirelli–Violin. PHILIPS. CD number: PHCP – 9001.

[8] Vivaldi. Four seasons, summer III. Presto, Music Score.

[9] Bach, Johann Sebastian. Air on the G String (Aria). Licence: Art Libre. Music Score.

[10] Chopin 12 Etudes. Performance by Pollini Maurizio. Piano, Universal Music Classics, CD number: UCCG – 7071.

[11] Chopin. Etude Opus 25 No. 11 – Allegro con brio, in A Minor, Berlin: Bote & Bock, music score.

[12] Erick Satie. Performance by Roger Pascal. Piano, DECCA-RECORDER, CD number: POCL – 4697.
[13] Satie. Gnossienne No. 4. Rouart, Lerolle & Cie. music score.
[14] Massenet. Meditatiion de “THAIS”. Nippon Gakufu Shuppannsha, music score.
[15] Leroy Anderson. Burghley’s Holiday. Alfred Publishing Co., Inc, music score.
[16] Leroy Anderson. Plink, Plank, Plunk! Alfred Publishing Co., Inc, music score.

Tatiana Endrjukaite was born in Riga, Latvia in 1986. She received her B.E. and M.E. degrees in computer science faculty from Transport and Telecommunication Institute in 2009 and 2011 respectively. In 2012 she worked at Communications Science Laboratory of NTT in Japan. Currently studying for a PhD at Keio University, Graduate School of Media and Governance faculty, Fujisawa, Japan. She has publication about "Time-dependent Genre Recognition by means of Instantaneous Frequency Spectrum based on Hilbert-Huang Transform", published in iiWAS 2012, and publication about "Music visualization technique of repetitive structure representation to support intuitive estimation of music affinity and lightness" in Journal of Mobile Multimedia. Current research interests are signal processing, music data mining, music visualization and music in social environment and humans emotions.

Yasushi Kiyoki received his B.E., M.E. and Ph.D. degrees in electrical engineering from Keio University in 1978, 1980 and 1983, respectively. In 1983 and 1984, he worked at Electrical and Communication Laboratory of NTT. From 1984 to 1996, he was with Institute of Information Sciences and Electronics, Univ. of Tsukuba, as an assistant professor and then an associate professor. Since 1996, he has been with Department of Environmental Information at Keio University, where he is currently a professor. His research addresses multidatabase systems, knowledge base systems, semantic associative processing, and multimedia database systems. He serves as the chair on Special Interest Group of Database Systems and as the editor-in-chief on Transactions on Databases in Information Processing Society of Japan. He also serves as the program chair for the 7th International Conference on Database Systems for Advanced Applications.