[2009]

Research report

Use of Variable Resolution Transform for Musical Descriptor Extraction

Aliaksandr Paradzinets

Liming Chen

[June 2009]

Use of Variable Resolution transform for musical descriptor extraction

Aliaksandr Paradzinets, Liming Chen
{aliaksandr.paradzinets, liming.chen}@ec-lyon.fr
Ecole Centrale de Lyon

As a major product for entertainment, there is a huge amount of digital musical content produced, broadcasted, distributed and exchanged. There is a rising demand for content-based music search services. Similarity-based music navigation is becoming crucial for enabling easy access to the ever-growing amount of digital music available to professionals and amateurs alike. This work presents new musical content descriptors and similarity measures which allow automatic musical content organizing (search by similarity, automatic playlist generating) and labeling (automatic genre classification). A novel variable resolution transform is presented and described in the context of music signal analysis. Higher level processing touches upon the musical knowledge extraction where the variable resolution transform is used in two algorithms – beat detection and multiple fundamental frequency estimation algorithms. The information issued from these algorithms is then used for building musical descriptors, represented in form of histograms (novel 2D beat histogram which enables a direct tempo estimation, note succession and note profile histograms etc.). A direct music information retrieval applications, namely music retrieval by similarity, which use aforementioned musical features are described and evaluated in this paper.

1. Introduction

As a major product for entertainment, there is a huge amount of digital musical content produced, broadcasted, distributed and exchanged. There is a rising demand for content-based music search services. Similarity-based music navigation is becoming crucial for enabling easy access to the ever-growing amount of digital music available to professionals and amateurs alike. A professional user, such as a radio programmer, may want to search for a different interpretation of one song to include in a radio playlist. In addition, a radio programmer has the need to discover new songs and artists to help his listeners to discover new music. The music amateur on the other hand has different needs, ranging from active music discovery for the fans, to the simple seed song playlist generation of similar items. Such ways to organize musical collections as genre classification and title structuring are important as they facilitate music navigation and discovery.

The primary stage in every kind of audio based music information retrieval is signal data analysis. Some algorithms perform analysis in the time domain as for example several beat detection algorithms. But the majority of music information retrieval algorithms perform their computation in the frequency domain, or a time-frequency representation, to be exact. So, the performance of all further steps of processing is strictly dependent on the initial data representation.

As compared to a vocal signal, a music signal is likely to be more stationary and posesses some very specific properties in terms of musical tones, intervals, chords, instruments, melodic lines and rhythms, etc. [1]. While many effective and high performance music information retrieval (MIR) algorithms have been proposed [2][3][4][5][6][7][8][9], most of these works unfortunately tend to consider a music signal as a vocal one and make use of MFCC-based features which are primarily designed for speech signal processing. Mel Frequency Cepstrum Coefficients (MFCC) was introduced in the 60's and used since that time for speech signal processing. The MFCC computation averages spectrum in sub-bands and provides the average spectrum characteristics. Whereas they are inclined to capture the global timbre of a music signal and claimed to be of use in music information retrieval [10][11], they cannot characterize the aforementioned music properties as needed for perceptual understanding by human beings and quickly find their limits [12]. Recent works suggest combining spectral similarity descriptors with high-level analysis in order to overcome existing ceiling [13].

The Fast Fourier Transform and the Short-Time Fourier Transform have been the traditional techniques in audio signal processing. This classical approach is very powerful and widely used owing to its great advantage of rapidity. However, a special feature of musical signals is the exponential law of notes' frequencies. The frequency and time resolution of the FFT is linear and constant across the frequency scale while the human perception of a sound is logarithmic according to Weber-Fechner law (including loudness and pitch perception). Indeed, as it is well known, the frequencies of notes in equally-tempered tuning system in music follow an exponential law (with each semi-tone the frequency is increased by a factor of $2^{1/12}$). If we consider a frequency range for different octaves, this frequency range is growing as the number of octave increases. Thus, to cover a wide range of octaves with a good frequency grid large sized windows are necessary in the case of FFT; this affects the time resolution of the analysis. On the contrary, the use of small windows makes resolving frequencies of neighboring notes in low octaves almost impossible. The ability of catching all octaves in music with the same frequency resolution is essential for music signal analysis, in particular construction of melodic similarity features. Hence, as the basis of our work in music feature based MIR, we propose a new music signal analysis technique by variable-resolution transform (VRT) particularly suitable to music signal.

Our VRT is inspired by Continuous Wavelet Transformation (CWT) introduced 20 years ago [14] and designed in order to overcome the limited time-frequency localization of the Fourier-Transform for non-stationary signals. Unlike classical FFT, our VRT depicts similar properties as CWT, i.e. having a variable time-frequency resolution grid with a high frequency resolution and a low time resolution in low-frequency area and a high temporal/low frequency resolution on the other frequency side, thus behaving as a human ear which exhibits similar time-frequency resolution characteristics [15].

1.1. Time-frequency transforms: FFT vs WT

There are plenty of works in the literature dedicated to musical signal analysis. The common approach is the use of FFT (Fast Fourier Transform) which has become a de-facto standard in music information retrieval community. The use of FFT seems straightforward in this field and relevance of its application for music signal analysis is almost never motivated.

There are some works in music information retrieval attempting to make use of wavelet transform as a novel and powerful tool in musical signal analysis. However, this new direction is not very well explored. [8] proposes to rely on discrete wavelet transform for beat detection. Discrete packet wavelet transform is studied in [16] to build time and frequency features in music genre classification. In [17], wavelets are also used for automatic pitch detection.

As it is well known, Fourier transform enables a spectral representation of a periodic signal as a possibly sum of a series of sines and cosines. While Fourier transform gives an insight into the spectral properties of a signal, its major disadvantage is that a decomposition of a signal by Fourier transform has infinite frequency resolution and no time resolution. It means that we are able to determine all frequencies in the signal, but without any knowledge about when they are present. This drawback makes Fourier transform to be perfect for analyzing stationary signals but unsuitable for irregular signals whose characteristics change in time. To overcome this problem several solutions have been proposed in order to represent more or less the signal in time and frequency domains.

One of these techniques is windowed Fourier transform or short-time Fourier transform. The idea behind is to bring time localization into classic Fourier transform by multiplying the signal with an analyzing window. The problem here is that the short-time discrete Fourier transform has a fixed resolution. The width of the windowing function is a tradeoff between a good frequency resolution transform and a good time resolution transform. Shorter window leads to smaller

frequency resolution but higher time resolution while larger window leads to greater frequency resolution but lower time resolution. This phenomenon is related to Heisenberg's uncertainty principle which says that

$$\Delta t \sim \frac{1}{\Delta f} \tag{1.1}$$

where Δt is a time resolution step and Δf is a frequency resolution step.

Remember that in our work the main goal is music analysis. In this respect, we consider a rather music-related example which illustrates specificities of musical signals. As it is known, the frequencies of notes in equally-tempered tuning system in western music follow a logarithmic law, i.e. adding a certain interval (in semitones) corresponds to multiplying a frequency by a given factor. For an equally-tempered tuning system a semitone is defined by a frequency ratio of $2^{1/12}$. So, the interval between two frequencies is

$$n = 12 \cdot \log_2 \left(\frac{f_2}{f_1} \right) \tag{1.2}$$

If we consider a frequency range for different octaves, it is growing as the number of octave is higher. Thus, applying the Fast Fourier Transform we either lose resolution of notes in low octaves (Figure 1.1) or we are not able to distinguish high-frequency events which are closer in time and have shorter duration.

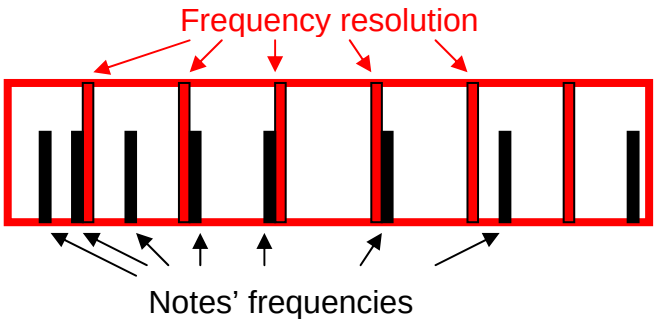


Figure 1.1. Mismatch of note frequencies and frequency resolution of the FFT.

Time-frequency representation, which can overcome resolution issues of the Fourier transform is ***Wavelet transform***. Wavelets (literally "small waves") are a relatively recent instrument in modern mathematics. Introduced about 20 years ago, wavelets have made a revolution in theory and practice of non-stationary signal analysis [14][18]. Wavelets have been first found in the literature in works of Grossmann and Morlet [19]. Some ideas of wavelets partly existed long time ago. In 1910 Haar published a work about a system of locally-defined basis functions. Now these functions are called Haar wavelets. Nowadays wavelets are widely used in various signal analysis, ranging from image processing, analysis and synthesis of speech, medical data and music [17][20].

Continuous wavelets transform of a function $f(t) \in L^2(R)$ is defined as follows:

$$W(a,b) = \frac{1}{\sqrt{|a|}} \int_{-\infty}^{\infty} f(t) \psi^* \left(\frac{t-b}{a} \right) dt \tag{1.3}$$

where $a, b \in R, a \neq 0$.

In the equation (1.3) $\psi(t)$ is called basic wavelet or mother wavelet function (* stands for complex conjugate). Parameter *a* is called wavelet scale. It can be considered as analogous to frequency in the Fourier transform. Parameter *b* is localization or shift. It has no correspondence in the Fourier transform.

One important thing is that the wavelet transform does not have a single set of basis functions like the Fourier transform. Instead, the wavelet transform utilizes an infinite set of possible basis functions. Thus, it has an access to a wide range of information including the information which can be obtained by other time-frequency methods such as Fourier transform.

As explained in brief introduction on music signal, a music excerpt can be considered as a sequence of note (pitches) events lasting certain time (durations). Beside beat events, singing voice and vibrating or sweeping instruments, the signal between two note events can be assumed to be quasi-stationary. The duration of a note varies according to the main tempo of the play, type of music and type of melodic component the note is representing. Fast or short notes usually found in melodic lines in high frequency area while slow or long notes are usually found in bass lines with rare exceptions. Let's consider the following example in order to see the difference between the Fourier transform and wavelet one. We construct a test signal as containing two notes E1 and A1 playing simultaneously during the whole period of time (1 second). These two notes can represent a bass line, which, as it is well known, does not change quickly in time. At the same time, we add 4 successive notes B5 with small intervals between them (around 1/16 sec). These notes can theoretically be notes of the main melody line. Let's see now the Fourier spectrogram of the test signal with a small analyzing window.

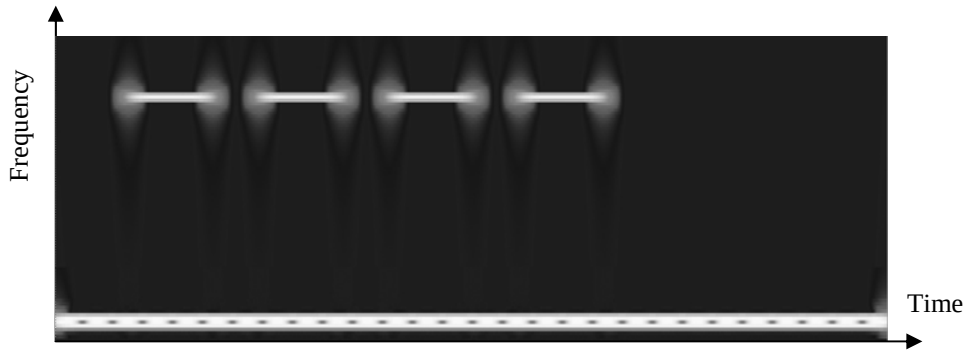


Figure 1.2. Small-windowed Fourier transform (512 samples) of the test signal containing notes E1 and A1 at the bottom and 4 repeating B5 notes at the top.

As we can see from Figure 1.2, while high-octave notes can be resolved in time, two bass notes are irresolvable in frequency domain. Now we increase the size of the window in the Fourier transform. Figure 1.3 illustrates the resulting spectrogram.

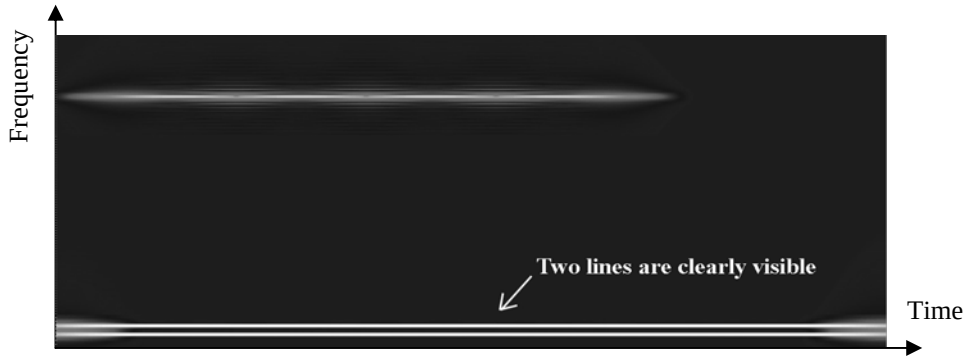


Figure 1.3. Large-windowed Fourier transform (≥1024 samples) of the test signal containing notes E1 and A1 at the bottom and 4 repeating B5 notes at the top.

As we can see, two lines at the bottom of the spectrogram are now clearly distinguishable while the time resolution of high-octave notes has been lost.

Finally we apply wavelet transform to the test signal. Figure 1.4 shows such Morlet-based wavelet spectrogram of our test signal.

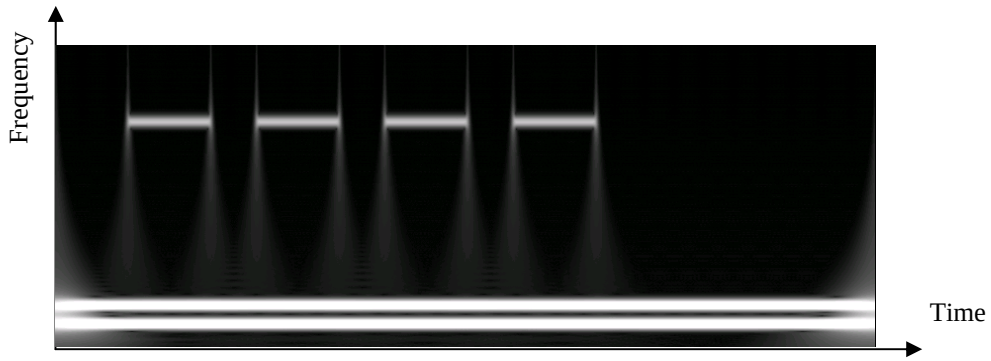


Figure 1.4. Wavelet transform (Morlet) of the test signal containing notes E1 and A1 at the bottom and 4 repeating B5 notes at the top.

Of course, the given example is quite artificial; however it explains well our motivation for a wavelet like time-frequency representation of a signal. It is also known, that human ear exhibits time-frequency characteristic closer to that from wavelet transform [15].

1.2. Other transforms and filter banks

The idea to adapt the time/frequency scale of a Fourier-related transform to musical applications is not completely novel. A technique called **Constant Q Transform** [21] is related to the Fourier transform and it is used to transform a data series to the frequency domain. Like the Fourier transform a constant Q transform is a bank of filters, but contrary to the Fourier transform it has geometrically spaced center frequencies $f_k = f_0 \cdot 2^{\frac{k}{b}}$ (k = 0; …), where b is the number of filters per octave. In addition it has a constant frequency resolutions ratio $R_{f/\Delta} = \left(2^{\frac{1}{b}} - 1\right)^{-1}$. Choosing appropriately k and f_0 makes central frequencies to correspond to the frequencies of notes.

In general, the transform is well suited to musical data (see e.g. [22], in [23] it was successfully used for recognizing instruments), and this can be seen in some of its advantages compared to the Fast Fourier Transform. As the output of the transform is effectively amplitude/phase against log frequency, fewer spectral bins are required to cover a given range effectively, and this proves useful when frequencies span several octaves. The downside of this is a reduction in frequency resolution with higher frequency bins.

Besides constant Q transform there are bounded version of it (BQT) which use quasi-linear frequency sampling when frequency sampling remains linear within separate octaves. This kind of modification allows construction of medium complexity computation schemes in comparison to standard CQT. However, making the frequency sampling quasi-linear (within separate octaves) renders the finding of harmonic structure much more complex task.

Fast Filter Banks are designed to deliver higher frequency selectivity maintaining low computational complexity. This kind of filter banks inherits all disadvantages of FFT in music analysis applications.

More advanced techniques, described for example in [24] are medium-complexity methods which aim to overcome disadvantages of FFT and try to follow note system frequency sampling. However, octave-linear frequency sampling keeps the same disadvantage as in the case of bounded Q transforms.

2. Variable Resolution Transform

Our Variable Resolutions Transform (VRT) is first derived from the classic definition of Continuous Wavelet Transform (CWT) in order to enable a variable time-frequency coverage which should fit to music signal analysis better. The consideration of specific properties of music signal finally leads us to change the mother function as well and thus our VRT is not a true CWT but a filter bank.

We start the construction of our VR Transform from Continuous Wavelet Transform defined by (1.3). Thus, we define our mother function as follows

$$\psi(t) = H(t,l)e^{j\cdot 2\pi\cdot t} \qquad \textbf{(2.1)}$$

where $H(t,l)$ is the Hann window function of a length l with $l \in Z$ as defined by (2.2). In our case l will lie in a range between 30-300 ms. Notice that using different different length values l amounts to change the mother wavelet function Ψ.

$$H(t,l) = \frac{1}{2} + \frac{1}{2}\cos\frac{2\pi t}{l} \qquad \textbf{(2.2)}$$

Once the length l is fixed, function (2.1) becomes much more similar to a Morlet wavelet. It is an oscillating function, a flat wave modulated by a Hann window. The parameter l defines the number of periods to be present in the wave. Figure 2.1 illustrates such a function with *l=20 waves*.

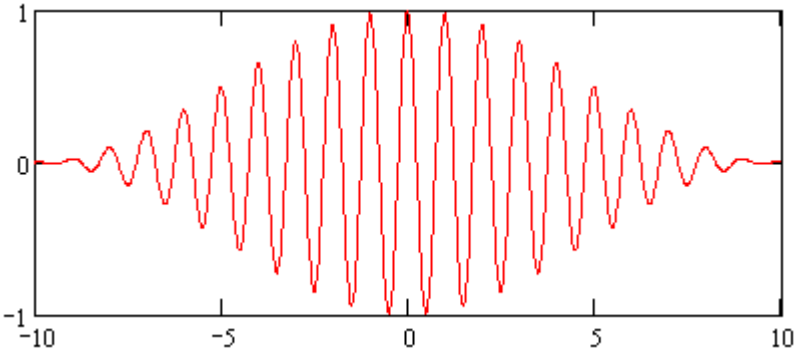


Figure 2.1. Our mother wavelet function. A flat wave modulated by a Hann window with *l=20*.

We can write according to the definition of the function (since $l < \infty$):

$$\int_{-\infty}^{\infty} |\psi(t)| dt < \infty \text{ and } \int_{-\infty}^{\infty} |\psi(t)|^2 dt < \infty \qquad \textbf{(2.3)}$$

The function is oscillating symmetrically around its 0 value, hence

$$\int_{-\infty}^{\infty} \psi(t)dt \to 0 \qquad \textbf{(2.4)}$$

Using (1.3) we write a discrete version of the transform for a sampled signal between the instants of time form *t–l/2* to *t+l/2.* Applying the wavelet transform to the signal, we are interested in spectrum magnitude

$$W(a,b) = \frac{1}{\sqrt{a}} \sqrt{\left(\sum_{t=-l/2}^{l/2} s[t+b] \cdot H\left[\frac{t}{a}, l\right] \cdot \cos\left(2\pi \frac{t}{a}\right)\right)^2 + \left(\sum_{t=-l/2}^{l/2} s[t+b] \cdot H\left[\frac{t}{a}, l\right] \cdot \sin\left(2\pi \frac{t}{a}\right)\right)^2} \qquad \textbf{(2.5)}$$

Here *W(a,b)* is the magnitude of the spectral component for the signal *s[t]* at time instant *b* and wavelet scale *a.*

The value of *W(a,b)* can be obtained for any *a* and *b* provided that *b* does not exceed the length of the signal. The equation (2.5) thus defines a Continuous Wavelet Transform for a discrete signal (time sampling).

The scale of wavelet *a* can be expressed in terms of central frequency corresponding to it since our mother function is a unit oscillation:

$$a = \frac{f_S}{f} \qquad \textbf{(2.6)}$$

where f_S is the sampling frequency of the signal.

A higher value of *a* stands for a lower central frequency.

2.1. Logarithmic frequency sampling

First of all, the sampling of the scale axis is chosen to be logarithmic in the meaning of frequency. It means that each musical octave or each note will have an equal number of spectral samples. Such a choice is explained by the properties of a music signal, which is known to have frequencies of notes to follow a logarithmic law (following the human perception). Logarithmic frequency sampling also simplifies harmonic structure analysis and economizes the amount of data necessary to cover the musical tuning system effectively.

A voiced signal with single pitch is in the general case represented by its *fundamental frequency* and the fundamental frequency's *partials (harmonics)* with the frequencies equal to the fundamental frequency multiplied by the number of a partial. Hence the distances between partials (harmonic components) and f_0 (basic frequency) in logarithmic frequency scale are constant independently from f_0. Such harmonic structure looks like a "fence", depicted on Figure 2.2.

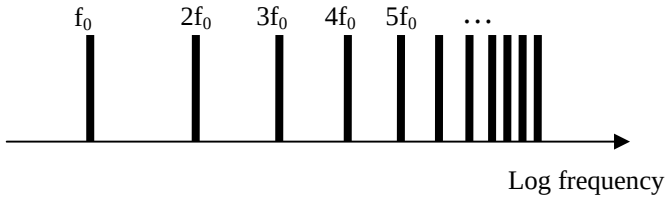


Figure 2.2. Harmonic structure in logarithmic frequency scale.

In order to cover the frequency axis form f_{min} to f_{max} with N frequency samples with a logarithmic law we define a discrete function $a(n)$, which denotes the scale of wavelet and where n stands for a wavelet bin number ranging in the interval 0..N-1.

$$a(n) = \frac{f_S}{f_{\min} e^{\frac{n}{N}\ln\left(\frac{f_{\max}}{f_{\min}}\right)}} \tag{2.7}$$

Now the transform (2.5) sampled in both directions gives

$$W(n,b) = \frac{1}{\sqrt{\frac{f_S}{f_{\min} e^{n \cdot C}}}} \left| \sum_{-l/2}^{l/2} s[t+b] \cdot H\left[\frac{t \cdot f_{\min} \cdot e^{n \cdot C}}{f_S}, l \right] \cdot e^{-i \frac{t f_{\min} \cdot e^{n \cdot C}}{f_S}} \right| \tag{2.8}$$

where the constant $C = \frac{1}{N} \ln\left(\frac{f_{\max}}{f_{\min}}\right)$.

Expression (2.8) is the basic expression to obtain an N-bin spectrogram of the signal at time instant b. Thus, for a discrete signal of length S, expression (2.8) provides $S \times N$ values for each instant of time, N being the number of frequency samples. The expression (2.8) is still a sampled version of the Continuous Wavelet Transform where the sampling of the scale axis has been chosen logarithmic for N samples.

Frequency dependency on the bin number has the following form (with $f_{min}=50$, $f_{max}=8000$, $N=1000$).

$$f(n) = f_{\min} e^{\frac{n}{N}\ln\left(\frac{f_{\max}}{f_{\min}}\right)} = f_{\min} e^{nC} \tag{2.9}$$

In order to depict the time/frequency properties of music signals by this discretized wavelet transform with a fixed length value (l=20), let's consider wavelet spectrograms of several test signals. Figure 2.3 shows the wavelet spectrogram $W(n,b)$ of a piano recording. One can observe single notes on the left and chords on the right. Fundamental frequency (f_0) and its harmonics can be observed in the spectrum of each note. As we can see from the Figure 2.3, up to 5 harmonics are resolvable. Higher harmonics after the 5th one become indistinguishable especially in the case of chords where the number of simultaneously present frequency components is higher.

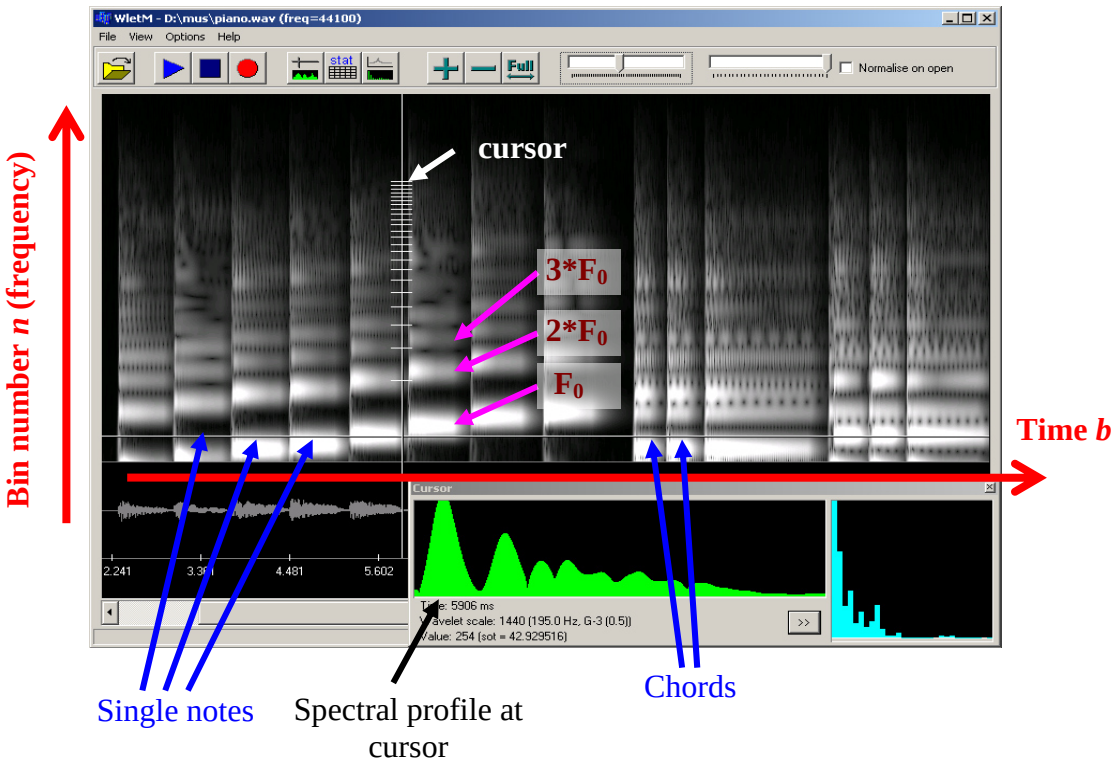


Figure 2.3. Wavelet spectrogram of a piano recording (wavelet (2.1)). Single notes on the left and chords on the right. Up to 5 harmonics are resolvable. Higher harmonics after the 5th one become indistinguishable especially in the case of chords where the number of simultaneous frequency components is higher.

Good time resolution is important in such tasks as beat or onset detection for music signal analysis. The next example serves to illustrate the time resolution properties of the Variable Resolution Transform we are developing. In this example we examine a signal with a series of delta-pulses (Dirac) as illustrated in Figure 2.4 which is a wavelet spectrogram of 5 delta-pulses (1 on the left, 2 in the middle and 2 on the right). As we can see from this figure, Delta-pulses on the picture are still distinguishable even if the distance between them is only 8 ms (right case). In the case of FFT one need 64-sample window size in order to obtain such time resolution.

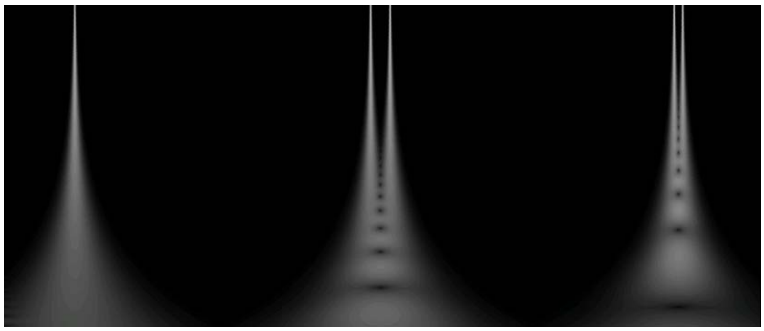


Figure 2.4. Wavelet transform of a signal containing 5 delta-pulses. The distance between two pulses on the right is only 8 ms.

A quite straightforward listening experiment that we have carried out reveals that the human auditory system is capable to distinguish delta-pulses when a distance between them is around 10 ms. On the other hand, the human auditory system is also able to distinguish very close frequencies - 4Hz in average[1], and down to 0.1Hz

[1] http://tonometric.com/adaptivepitch/

2.2. Varying the mother function

However, music analysis requires good frequency resolution as well. As we can see from the spectrogram in Figure 2.3, neither high-order partials nor close notes are resolvable, because the spectral localization of the used wavelet is too wide. Increasing the length parameter *l* in (2.1) or (2.8) of the Hann window would render our wavelet transform unusable in low-frequency area since the time resolution in low-frequency area would rise exponentially. Thus, we propose in this work to make dynamic parameter *l* **with** a possibility to adjust its behavior across the scale axis**.** For such a purpose we propose to use the following law for parameter *l* in (2.8) instead of applying scale *a(n)* to parameter *t* in *H(t,l)*:

$$l(n) = L \cdot \left(1 - k_1 \frac{n}{N}\right) \cdot e^{-k_2 \frac{n}{N}} \qquad \textbf{(2.10)}$$

where L is the initial window size, k_1 and k_2 – adjustable parameters

The transform (2.8) becomes:

$$W(n,b) = \frac{1}{\sqrt{\frac{f_S}{f_{\min} e^{n \cdot C}}}} \left| \sum_{-l/2}^{l/2} s[t+b] \cdot H\left[t, L \cdot \left(1 - k_1 \frac{n}{N}\right) \cdot e^{-k_2 \frac{n}{N}}\right] \cdot e^{-i \frac{t f_{\min} \cdot e^{n \cdot C}}{f_S}} \right| \qquad \textbf{(2.11)}$$

The expression (2.10) allows the effective"wavelet" width to vary in different ways: from linear to completely exponential to follow the original transform definition. When $L = \frac{f_S}{f_{\min}}$, $k_1=0$ and $k_2=C \cdot N$, (2.11) is equivalent to (2.8).

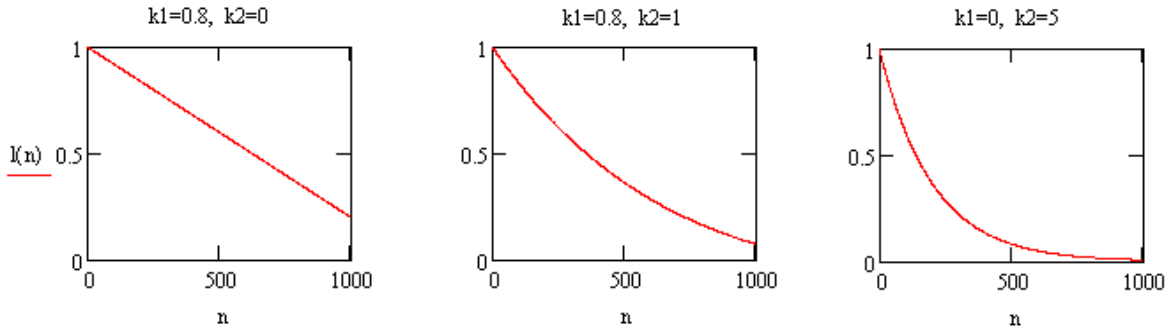


Figure 2.5. Various l(n), depending on parameters. From linear (left) to exponential (right).

Doing so, we are now able to control the time resolution behavior of our transform. In fact, such transform is not anymore a wavelet transform since the mother-function changes across the scale axis. For this reason we call the resulted transform as *variable resolution transform* (VRT). It can be also referred as a custom filter bank.

As the effective mother-function width (number of wave periods) grows in high-frequency relatively to the original mother-function, the spectral line width becomes more narrow, and hence the transform allows to resolve harmonic components (partials) of the signal. An example of the spectrogram with new variable resolution transform is depicted in Figure 2.6.

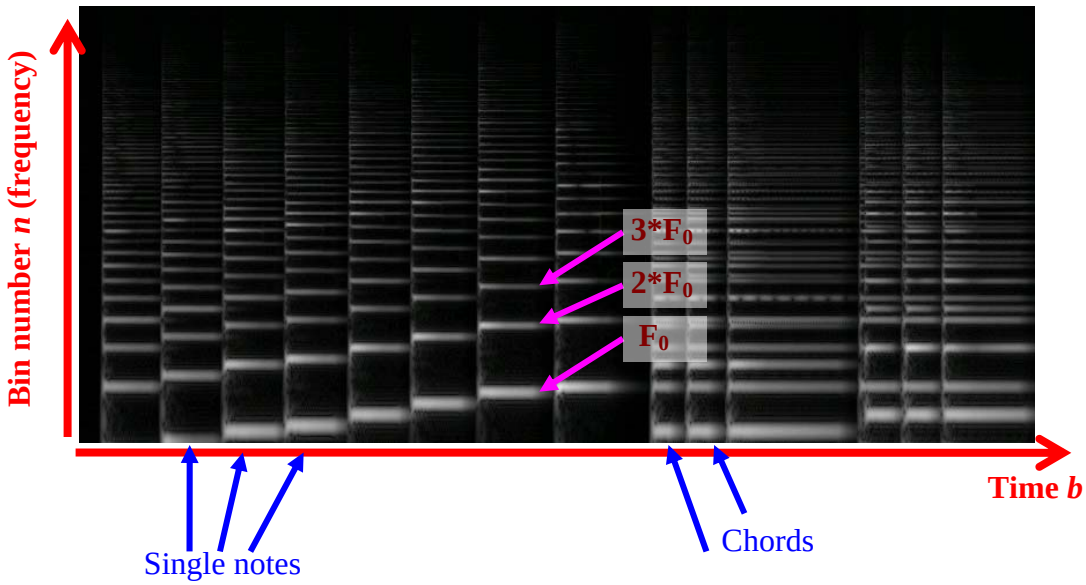


Figure 2.6. VRT spectrogram of the piano recording used in the previous experiment. Fundamental frequencies and partials are distinguishable (k1=0.8, k2=2.1).

2.3. Properties of the VR transform

Here we proceed to study the properties of our VR transform within the scope of the present work, i.e., with regard to music signals.

A music signal between 50 and 8000 Hz contains approximately 8 octaves. Each octave consists of 12 notes, leading to a total number of notes around 100. A filterbank with 100 filters would be enough to cover such octave range. In reality, frequencies of notes may differ from the theoretical note frequencies of equal-tempered tune because of recording and other conditions. Therefore for music signal analysis considered here, we are working with spectrogram size of 1024 bins – 10 times the amount necessary which covers the note scale by 10 bins per note. Timbre is a one of major properties of music signal along with melody and rhythm. Let's consider now a structure of partials of a harmonic signal (harmonic structure). In Figure 2.2 we have depicted an approximate view of such structure in logarithmic frequency scale. According to the definition of the function *f(n)* (2.9), the distance between partial *i* and partial *j* in terms of number of bins is independent of the absolute fundamental frequency value.

Indeed, according to (2.9) $n(f)=\frac{1}{C}\ln\frac{f}{f_{\min}}$ and taking into account $f_i=i*f_0$ and $f_j=j*f_0$ we obtain:

$$n(f_j)-n(f_i)=\frac{1}{C}(\ln(f_0\cdot j)-\ln f_{\min})-\frac{1}{C}(\ln(f_0\cdot i)-\ln f_{\min})=\frac{1}{C}(\ln(f_0\cdot j)-\ln(f_0\cdot i))=\frac{1}{C}\ln\frac{j}{i}$$

An accurate harmonic analysis of music signal implies that frequency resolution in terms of spectrogram bin number, expressed by the spectral dispersion, should be always below the distance between neighboring components under consideration.

Having the total width of 20-partial harmonic structure to be a constant around 600 points in terms of number of bins ($n(f_{20})$ - $n(f_0)$), we can establish that the frequency resolution of the obtained transform is large enough to resolve high-order partials we are interested in at all positions of the VRT spectrogram, especially for low octave notes. It means that a 20-partial

harmonic structure starting from the beginning of the spectrogram will always lie *above* the dispersion curve. If we consider now the time resolution of the transform, we must recall Figure 2.5, where various dependencies on the effective width of filter were given. If we define the maximum effective window size to be 180ms (recall our musical signal properties) we obtain the following time resolution grid as illustrated in Figure 2.7.

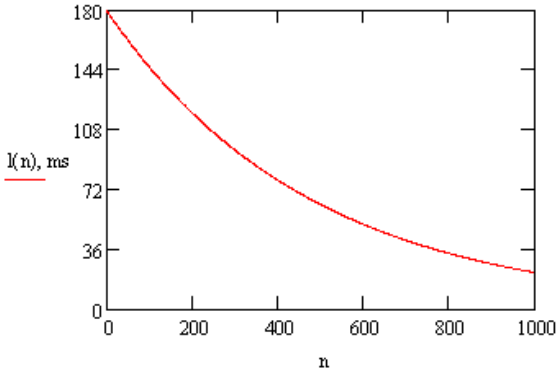


Figure 2.7. Time resolution dependency of VR transform with $k_2=0.8$, $k_2=2.1$.

2.4. Discussion

Our Variable Resolution Transform is derived from the classic definition of Continuous Wavelet Transform given in 1.1. In our previous work, we referred to our VRT as "Wavelet-Like" or "Pseudo-Wavelet" transform [25][26]. Actually, our VRT is not a CWT even though they have many similarities. The main difference between VRT and CWT consists in the frequency axis sampling, as well as in the mother wavelet function which is changing its form across the scale (or frequency) axis in the case of VRT in order to have enough resolution details for high order frequency partials. This last property is not a wavelet transform, because in the true wavelet transform the mother function is only scaled and shifted making a discrete tiling of the time-frequency space in the case of DWT or infinite coverage in the case of CWT. Our VRT can be also referred to as a specially crafted filter bank. Major differences between our VRT and a wavelet transform are:

- no 100% space tiling
- no 100% signal reconstruction (depending on parameters)
- mother function changes

Major similarities between our VRT and a wavelet transform are the following:

- They are based on specially sampled version of CWT
- with certain parameters they can provide 100% signal reconstruction
- low time resolution and high frequency resolution in low frequency area and high time with low frequency resolution in high frequency area

3. Applications: VRT-based similarity features

The most known acoustic characteristics generally used for audio similarity measurements are MFCC, fluctuation patterns, "gravity" [27], etc. In this paper we propose several new acoustic features – 2D beat histogram, timbre histogram as well as note profile and note succession histograms. Unlike simple spectral features, these new measurements take into account semantic information such as rhythm, tonality etc.

3.1. 2D beat histogram for rhythmic similarity

The idea of building a beat histogram is not novel [8]. Simple 1D beat histogram can be used in genre classification, tempo induction as well as music similarity search. In our work we propose a modified histogram – a two-dimensional one. Unlike 1D histogram this 2D histogram is free from beat detection threshold issue.

The beat/onset detection algorithm being used in this work is based on Variable Resolution Transform as all other algorithms in our work. Here the signal processed by VRT is treated a grayscale image. Thus, we apply image treatment operators like Sobel. In the resulting spectrogram image (Figure 3.1) distinct vertical lines are likely to represent beats or onsets.

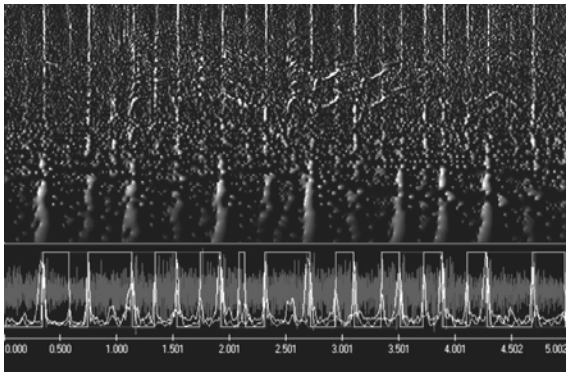


Figure 3.1. Treated wavelet spectrogram of musical excerpt.

Further, the enhanced spectrogram $W^*(t, scale)$ is processed by calculating a beat curve in the following way. A small 5-sample window together with preceding large 100-sample window is moved across the enhanced spectrogram. The value of the beat curve in each time moment is the number of points in the small window with values higher than a threshold which is obtained from the average value of points in the large window. Numerous beat curves may be computed separately by dividing the spectrum into bands. For the general question of beat detection the only one beat curve is used.

The probable beats are situated in beat curve's peaks. However, the definition of final beat threshold for the beat curve is problematic. Adaptive and none-adaptive algorithms for peak detection may be unstable. Many weak beats can be missed while some false beats can be detected.

Recall that our aim is the use of the rhythmical information for music similarity estimation. One of rhythmical information representation is the beat histogram. A classical one-dimensional beat histogram provides some knowledge only about the different beat periods while the distribution of beats in the meaning of their strength is not clear. At the same time beat detection algorithm and its parameters affect the form of the histogram. In order to avoid the dependency from the beat detection algorithm parameters we propose a 2D form of beat histogram, which is built with a beats period on the X axis and with amplitude (strength) of a beat on the Y axis (Figure 3.2). The information about beat strength in the proposed histogram is implicit since the histogram is computed upon the threshold used in beat detection. It is hence possible to avoid the disadvantage of recording conditions dependency (e.g. volume) and peak detection method. The range of threshold variation is taken from 1 to the found maximum-1. Thus, the beat strength is taken relatively and the volume dependency is avoided.

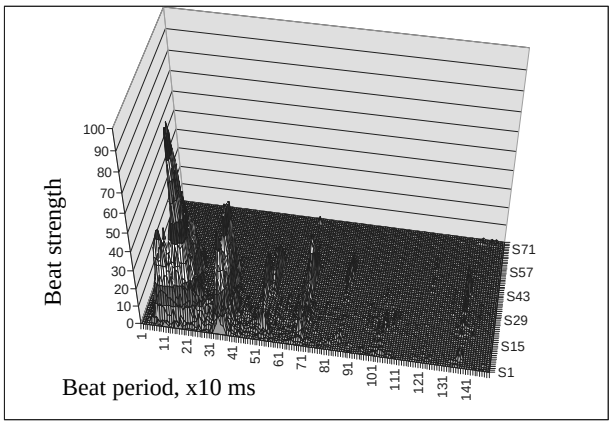


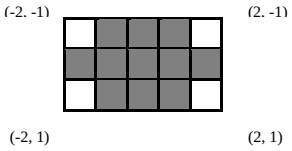
Figure 3.2. A 2-D beat histogram.

Such histogram can likely be a feature vector for example in genre classification or music matching.

The measure of rhythmic distance can be defined in numerous ways. In our experiments we have find out the following equation which takes into account slight variation of rhythm of musical pieces being compared.

$$Dist_{H1,H2} = \sum_{x=1,y=1}^{N,M} \frac{1}{2}\left(\min_{R}\left(\left|H1_{x,y} - H2_{(x,y)+R}\right|\right) + \min_{R}\left(\left|H1_{(x,y)+R} - H2_{x,y}\right|\right)\right) \quad \textbf{(3.1)}$$

where
H1, H2 – beat histograms to compare
N, M – beat histogram size
R – an area of the following form (to allow slight variations)



3.2. Transcription-derived similarity features

This paragraph covers aspects of higher level musical similarity metrics. Algorithms described in the paragraph are based on automated transcription (multiple F0 estimation) of polyphonic music with the use of VRT (former Continuous Wavelet-like Transform) described in [25].

The transcription algorithm issues for each window a list of detected f0`s together with relative amplitudes of their partials. This information is then used for building several kinds of statistical characteristics (histograms).

The simplest way to calculate a similarity distance is to calculate a distance between note histograms. Note histogram (profile) is computed across the whole musical title or its part and serves for estimation of musical similarity by tonality as well as tonality (musical key) itself. Tonality in music is a definition of note set used in a piece which is characterized by tonic or key note and mode (e.g. minor, major). Each tonality has its own distribution of notes involved in a play and it can be obtained from the note histogram [28]. To compare two musical titles in the meaning of tonal similarity we calculate a similarity of two note profiles. These profiles must be either aligned by the detected tonality's key note (e.g. by Re for D-dur or D-mol) or a maximal similarity across all possible combinations of tonalities must be searched.

Another musical similarity metric we propose in the current work is a similarity based on note successions histogram. Here probabilities of 3-note chains are collected and their histogram is then used as a "fingerprint" of musical title. A musical basis of such similarity metric is that if

same passages are frequent in two musical compositions, it gives a chance that these two compositions have similarities in melody or harmony.

The procedure is note successions histogram calculation is following. First, note extraction over the whole piece is carried out with a step of 320 samples (20ms). Then detected notes are grouped in local note histograms in order to find a dominant note in each grouping window. The size of the grouping window may vary from 100ms to 1 sec. Finally, all loudest notes are extracted from local histograms and their chains are collected in the note successions histogram. The resulting histogram is 3-dimensional histogram where each axe is a note of 3-note chain found in the musical piece being analyzed (Figure 3.3).

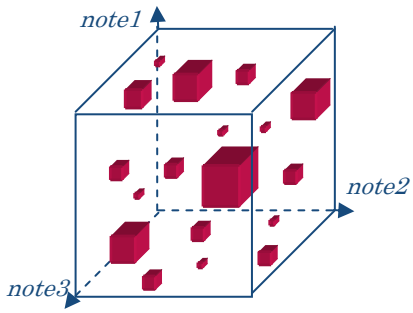


Figure 3.3. Note successions histogram example in 3D.

The third characteristic we extract from a musical piece is a timbre histogram. In general, "voiced" instruments differ from each other also by their timbre – profile of their partials. In our work we collect all detected notes with relative amplitude of their harmonics. Further, relative amplitudes of harmonics are reduced to 3-4 bits and attached together in order to form a number. Histogram of these numbers is then computed. Comparing of such histograms gives one more possibility of a similarity measurement.

3.3. Combining of similarity types

While pure similarity metrics could be interesting for exact matching of musical pieces by certain criteria, a combination of similarities have in goal building of "general" similarities like human listener could do (e.g. finding a piece with the same rhythm and key type could issue two slow sad melodies which are judged similar by a human listener).

In our work we have studied to variant of combining. A liner combining is simple weighted sum of distances

$$D = \sum k_i d_i \tag{3.2}$$

Another version of liner combining is a weighted sum of ratings. In this case for every kind of similarity being combined its rating or position in a sorted list of similar titles is obtained. Final distance is computed as a weighted sum of ratings. In some cases when distances which are being combined have different natures and cannot be combined linearly.

More sophisticated means of combining can involve such techniques as neuron networks trained on user feedback data.

3.4. Experiments

Recall, that that our paper is dedicated to similarity estimation of musical pieces. Our main experiments have in aim an estimation of musical similarity accuracy. They consist of two evaluation parts – listening test and reinterpreted pieces search.

3.4.1. Listening test

Preliminary experiments with musical similarity search were carried out. A database of approximately 1000 musical composition of different artists, genres and rhythms has been processed. For each type of similarity metric (rhythmic, tonality, timbre and melodic) a similarity matrix 1000x1000 has been created. Then the system retrieved by different combinations of similarity metrics the 5 most similar songs from the database for a given example. Researches from the laboratory (not necessarily working with music) were taken as listeners. They were proposed to rate random queries from the database with scores from 0 (not similar) to 5 (very similar) according to shown similarity type. Neither songs' titles nor artist names were provided to listeners. Also with a probability of 50% listeners were provided by random and not similar music pieces without being notified of this fact in order to avoid prejudgements.

In our experiments we have used 4 pure similarity metrics: rhythmic, tonality, timbre and melodic; and 4 mixtures where *comb1* was a combination of tonality and rhythm metrics, *comb2* – timbre and rhythm, comb3 – tonality + melody + rhythm, comb4 –timbre + melody + rhythm. For the mentioned mixtures both liner and rating combinations were applied.

Evaluation results obtained in our experiments are presented in the Table 3.1. Here for each similarity type there is mean and median value of totality of votes. The column "corresponding random" shows the mean and median of listeners' votes for those cases when listeners were proposed random songs as similar. Since listeners were not notified about this fact, they still had to evaluate how similar were the proposed songs. These data are used as background un-truth. All found multiple interpretation of songs were not filtered out and considered as 5 – very similar.

Table 3.1. Listening test results (mean / median).

Similarity type	Linear combination or single	Rating combination	Corresponding random
rhythmic	2.92 / 2	n/a	0.40 / 0
tonality	3.16 / 3		2.41 / 3
timbre	2.16 / 2		0.81 / 0
melodic	2.23 / 2		1.60 / 2
comb1	3.55 / 4	2.06 / 3	0.94 / 1
comb2	2.78 / 3	3.75 / 4	0.97 / 0
comb3	3.85 / 5	1.80 / 1	0.75 / 0
comb4	2.49 / 3	2.26 / 3	1.01 / 0

Figure 3.4 shows normalized distributions of votes for mentioned single similarity metrics. The upper histograms stand for distributions of votes for "random" similar songs.

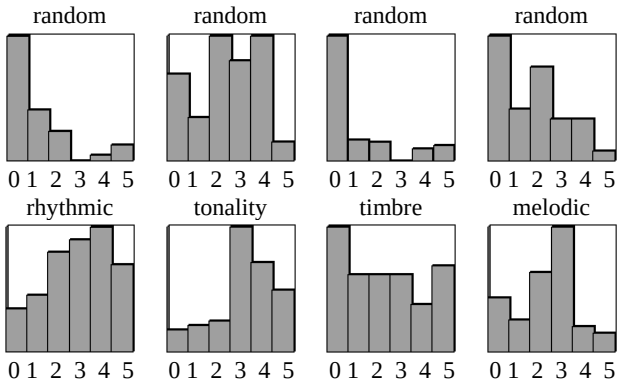


Figure 3.4. Histograms of listeners' votes for pure similarity metrics.

On the Figure 3.5 normalized histograms of votes for composite similarities are depicted. Here the upper row is showing histograms of votes for random songs. Two other rows include results for linear (lin) and rating (rt) combinations.

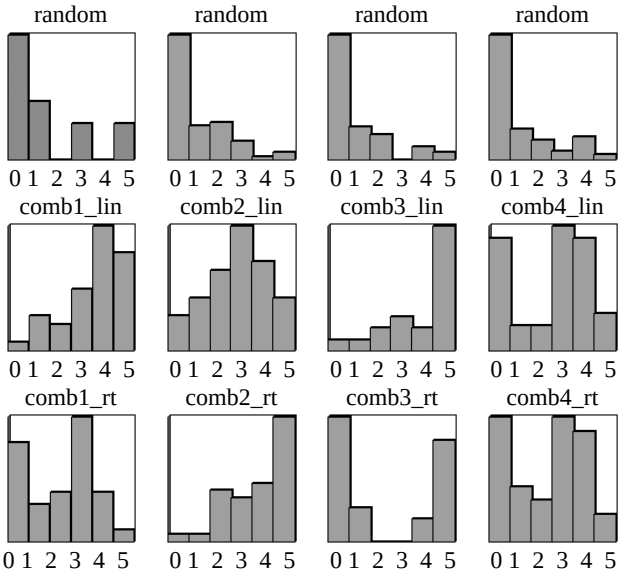


Figure 3.5. Histograms of listeners' votes for combined similarity metrics.

3.4.2. Reinterpreted songs search

Evaluation of melodic similarity metrics was based on composing of similarity playlists for musical titles that have multiple reinterpretations. The database of these titles used in this work is a certain number of musical files in MP3 format. The list is as follows.

1. Ennio Morricone – "Chi Mai", 3 interpretations
2. Roxette – "Listen to Your Heart", DHT – "Listen to Your Heart", DHT – "Listen to Your Heart" (dance)
3. Rednex – "Wish You Were Here", Blackmore's Night – "Wish You Were Here"
4. Tatu – "Not Gonna Get Us" (Eng), Tatu – "Nas Ne Dogonyat" (Rus)
5. Tatu – "All the Things She Said" (Eng), Tatu – "Ya Soshla s Uma" (Rus), Tatu – Remix
6. Tatu – "30 minutes" (Eng), Tatu – "Pol Chasa" (Rus)
7. Archie Shep, Benny Golson, Dexter Gordon, Mike Nock Trio, Ray Brown Trio – "Cry Me a River" (ver.1 jazz instrumental)
8. Diana Krall, Tania Maria, Linda Ronstadt, Bjork, Etta James, July London – "Cry Me a River" (ver. 2. vocal)

In this experiment the different interpretations of the same title are considered as "similar".

In the experiment playlists with 30 similar titles corresponding to each musical title in the database were built. Appearance of “a priori” similar titles at the top of playlist was considered as successful similarity output. The following table shows the result of playlist composition. It gives the information about position of appearance of similar titles in the associated playlist (1 – is the original music file).

Table 3.2. Objective evaluation results of music similarity measurements.

Original music composition	**Positions of appearance of similar titles**
Chi Mai	(1), 2, 3
Listen To Your Heart	(1), 3, 12
Wish You Were Here	(1), 2
Not Gonna Get Us	(1), 2
All the Things She Said	(1), 2, 3
30 minutes	(1), 2
Cry Me a River (ver. 1)	(1), 2, 3, 4, 6
Cry Me a River (ver. 2)	(1), 2, 4, 7, 8, n/a

Presence of similar songs in first positions of playlists signifies good performance of given melodic similarity metrics.

3.4.3. Playlist relevance evaluation

Finally we proceed on analysis of relevance of top 5 songs in playlists generated for seed songs. We considered two types of relevance: number of songs from the same genre and number of songs from the same artists. For the database we took ISMIR2004 genre classification database based on *Magnatune* collection. The database contained totally 729 titles of 128 artists in 6 genres.

The obtained results are as following (Table 3.3).

Table 3.3. Average number songs in the same genre or from the same artist.

Similarity type	**Same genre**	**Same artist**
Comb2_lin	3.58	0.99
Comb2_rt	3.48	0.89
Comb3_lin	3.07	0.86

The next picture (Figure 3.6) depicts distribution histograms of number of songs in the same genre and from the same artist for the best combination which in this case is comb2_lin.

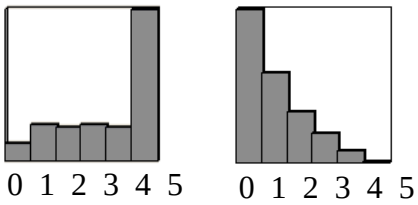


Figure 3.6.. Histogram of number of songs in the same genre in TOP-5 (left), and histogram of number of songs from the same artist in (TOP-5) (right).

Results of relevance analysis reported in literatures includes such numbers as average 1.43 songs in TOP-5 with the same genre as the query [29], average 3.44 of similar genres and 1.17 of similar artist [3]. A result obtained from the same ISMIR'2004 database found in literature is an average 3.4 songs (67.9%) in TOP-5 with the same genre [30].

4. Conclusion

In this paper we have considered the problem of automatic music analysis within such music information retrieval applications as music search by similarity (intelligent navigation) and automatic genre classification.

We have proposed an appropriate tool of musical signal analysis. We presented the variable resolution transform as such a tool. We have shown it to be better suited for our applications. The goal we have achieved is to obtain a single transform which can simultaneously cover the whole time-frequency scale in such a way that both pitch and rhythm information is gathered at the same time. The advantage of the tool we have proposed is that it has logarithmic frequency sampling in order to follow musical notes. In comparison to some classical approaches where the frequency sampling is also logarithmic, we have an improved frequency resolution in high frequency area, allowing us to better distinguish the high-order harmonics of the signal.

Several musical features and corresponding similarity measures have been proposed in this paper. Some of them were already known in literature (pitch class or note profile, 1D beat histogram), some of them are newly presented in the paper (note succession histogram, timbre histogram, 3D beat histogram). All these music features are closely related to musical content.

In this paper we have also described a direct application of music features and the associated similarity measures – music search by similarity. The evaluation we have carried out consisted of subjective judgment (human feedback) and objective evaluation such as relevance analysis. Objective evaluation showed quite good, but rather unstable results when using linear or rating combination of similarity measure. We have also found the best two combined similarity measures which were combinations of rhythm/tonality/melody and rhythm/timbre. A surprising result was observed when putting timbre similarity measure instead of tonality in the first combination, producing lower results. However, putting all distances together in a neuron-network combination mechanism showed stable results but not higher than in the case of linear combinations.

The objective analysis of similarity retrieval algorithm have shown very good similar genre rate – 3.58 against the best rate found in literature (3.4) in TOP-5 playlists analysis based on the ISMIR'04 corpus. Promising results were also achieved in search for pieces with multiple interpretations.

5. References

[1] Tanguiane A.S.. Artificial perception and music recognition (lecture notes in computer science). . Springer, October 1993.

[2] Casagrande N., Eck D., Kegl B.. Frame-level audio feature extraction using adaboost. *Proceedings of the ISMIR International Conference on Music Information Retrieval (London)* (2005) : pp. 345-350.

[3] Logan B. SA. A music similarity function based on signal analysis.. *In Proceedings of IEEE International Conference on Multimedia and Expo ICME 01* (2001)

[4] Mandel M. ED. Song-level features and support vector machines for music classification. *Proceedings of the ISMIR International Conference on Music Information Retrieval (London)* (2005) : pp. 594-599.

[5] McKinney M.F. BJ. Features for audio and music classification. *Proceedings of the ISMIR International Conference on Music* (2003) : pp. 151-158.

[6] Meng A., Shawe-Taylor J.,. An investigation of feature models for music genre classification using the support vector classifier. *Proceedings of the ISMIR International Conference on Music Information Retrieval (London)* (2005) : pp. 604-609.

[7] Scaringella N. ZG. On the modeling of time information for automatic genre recognition systems in audio signals. *Proceedings of the ISMIR International Conference on Music Information Retrieval (London)* (2005) : pp. 666-671.

[8] Tzanetakis G. CP. Automatic musical genre classification of audio signals. *IEEE Transactions on Speech and Audio Processing 10* (2002) : p. no 5, 293-302.

[9] West K. CS. Features and classifiers for the automatic classification of musical audio signals. *Proceedings of the ISMIR International Conference on Music Information Retrieval (Barcelona, Spain)* (2004) : pp. 531-536.

[10] Foote Jonathan T.. Content-based retrieval of music and audio. *Proceedings of SPIE Multimedia Storage and Archiving Systems II (Bellingham,WA) vol. 3229, SPIE* (1997) : pp. 135-147.

[11] Logan B.. Mel frequency cepstral coefficients for music modeling. *Proceedings of the ISMIR International Symposium on Music Information Retrieval (Plymouth, MA)* (2000)

[12] Aucouturier J.J. PF. Timbre similarity: how high is the sky?. *In JNRSAS* (2004)

[13] Pampalk E.. Computational models of music similarity and their application in music information retrieval. PhD thesis at Technishen Universitaet Wien, Fakultaet fuer Informatik.2006.

[14] Kronland-Martinet R. MJAGA. Analysis of sound patterns through wavelet transform. *International Journal of Pattern Recognition and Artificial Intelligence,Vol. 1(2)* (1987) : pp. 237-301.

[15] Tzanetakis G., Essl G., Cook P.. Audio analysis using the discrete wavelet transform. . *WSES Int. Conf. Acoustics and Music: Theory 2001 and Applications (AMTA), Skiathos, Greece* (2001)

[16] Grimaldi M., Kokaram A., Cunningham P.. Classifying music by genre using the wavelet packet transform and a round-robin ensemble. (2002)

[17] Kadambe S. FBG. Application of the wavelet transform for pitch detection of speech signals. *IEEE Transactions on Information Theory* (1992) 38, no 2: pp. 917-924.

[18] Mallat S.G.. A wavelet tour of signal processing. . Academic Press, 1999.

[19] Grossman A. MJ. Decomposition of hardy into square integrable wavelets of constant shape. *SIAM J. Math. Anal.* (1984) 15: pp. 723-736.

[20] Lang W.C. FK. Time-frequency analysis with the continuous wavelet transform. *Am. J. Phys.* (1998) 66(9): pp. 794-797.

[21] Brown J. C.. Calculation of a constant q spectral transform. *J. Acoust. Soc. Am.* (1991) 89(1): pp. 425-434.

[22] Nawab S.H., Ayyash S.H., Wotiz R.. Identification of musical chords using constant-q spectra.. *In Proc. ICASSP* (2001)
[23] Essid S.. Classisfication automatique des signaux audio-fréquences : reconnaissance des instruments de musique. PhD thesis Informatique, Telecommunications et Electronique, ENST.2005.
[24] Diniz F.C.C.B, Kothe I, Netto S.L., Biscainho L.P.. High-selectivity filter banks for spectral analysis of music signals. *EURASIP Journal on Advances in Signal Processing* (2007)
[25] Paradzinets A., Harb H., Chen L.,. Use of continuous wavelet-like transform in automated music transcription. *Proceedings of EUSIPCO* (2006)
[26] Paradzinets A., Kotov O., Harb H., Chen L.,. Continuous wavelet-like transform based music similarity features for intelligent music navigtion. *In proceedings of CBMI* (2007)
[27] Pampalk E., Flexer A., Widmer G.. Improvements of audio-based music similarity and genre classification. *In proceedings of ISMIR* (2005)
[28] Chuan C.-H. CE. Polyphonic audio key finding using the spiral array ceg algorithm. *Proceedings of ICME* (2005)
[29] Aucouturier J.J. PF. Music similarity measures : what's the use?. *Proceedings of ISMIR* (2002)
[30] Pohle T.. Post processing music similarity computation. *MIREX* (2006)