USING OQSTFT AND A MODIFIED SHS TO DETECT THE MELODY IN POLYPHONIC MUSIC (MIREX 2009)

Morten Wendelboe
morten@morwen.dk

ABSTRACT

At the melody extraction task at MIREX, new and improved methods has been presented within the last few years. Some of the more successful methods are somewhat complicated. This paper briefly presents a very simple method, which seems to be able to compete with the more complicated methods. The method does not detect the presence of the melody — it is assumed that a melody is always present. The method is based on 2 modifications of the SHS method, and uses a simple time/frequency representation of the audio signal.

1 Introduction

Although improved methods have been presented at the MIREX melody extraction tasks within the last few years, it still seems to be a very difficult task to extract the fundamental frequency of the melody. At the melody extraction task it is possible to get a notion of how well a method performs compared to other methods. It is important to notice that the performance of the methods that participated at the previous competitions seems to depend a lot on the test files. The authors have done an excellent job but it seems to be very difficult to develop a method that performs well on a lot of different audio files. Even the best methods that participated in 2008 only achieved a raw pitch score of around 60% for one or more files.

This paper presents a very simple method that seems to be able to compete with some of the best methods from the MIREX 2008 competition. This could indicate that radical new methods need to be developed in order to achieve higher scores.

The method described in this paper only deals with the task of estimating the fundamental frequency of the melody and does not detect if the melody is present.

2 Method description

The method is briefly described in the following. For more details, see [7] (in Danish).

2.1 Time/frequency representation using OQSTFT

Often the melody consists of harmonic sounds. Two very used musical effects are vibrato and glissando, where the fundamental frequency changes over time. When the fundamental frequency of a harmonic sound vibrate, the higher harmonics vibrate over a wider frequency range than that of the lower harmonics. In order to get a relatively clear picture of how the higher harmonics change over time a STFT with a short window length i needed. But in the lower harmonics it would be preferable to use a STFT with a long window length, as this would facilitate accurate detection of the frequency of each harmonic. Judith Brown has suggested the use of a Constant Q spectral transform [1] where the window length is halved when the frequency doubles. Unfortunately, it seems like this makes the window length too short in the higher frequencies and too long in the lower frequencies. A good compromise between the constant window length in a normal STFT and the window length used by Brown is the window length used in Optimal Q-STFT (OQSTFT) [7], where the window length is halved when the square root of the frequency is doubled.

Let us think about a harmonic chirp and let us try to find a good time/frequency resolution for every harmonic. The uncertainty in time and frequency can't really be illustrated as a rectangular box but let us do it anyway, as it is a useful approximation. The size of the area of the rectangular box has a lower bound due to the uncertainty principle. In order to minimize the influence of frequency components from other sounds we want to fit the rectangular box as close to the chirp as possible. This is the case when the chirp represents the diagonal of the rectangular box. [1] So the chirpy-ness (the gradient) of the chirp determinates the ratio between the uncertainty in time (the length of the rectangular box) and the uncertainty in frequency (the height of the rectangular box). The area of the rectangular box is constant and is determinated by the uncertainty principle. The chirpy-ness of the chirp is called c, the area of the rectangular box is called A and the length of the box is called l. Remember that the average frequency of an arbitrary harmonic is calculated as $\nu_x = x\nu_1$, where ν_1 is the average frequency of the lowest harmonic and x states the number of the harmonic. So let us calculate the optimal window length at frequency ν_x under the assumption that we know the optimal window length at frequency ν_1. We start with the area of the rectangular boxes at frequency ν_1 and ν_x, as we know that the area is constant:

$$A_1 = A_x \quad \Rightarrow \quad cl_1^2 = xcl_x^2 \quad \Rightarrow \quad l_x = \frac{1}{\sqrt{x}} l_1. \quad (1)$$

[1] By using a chirplet transform it is also possible to rotate the rectangular box. The Matching Pursuit introduced by Mallat and Zhang in [6] is an example of an efficient implementation of a chirplet transform.

So if the optimal window length l_1 is known at frequency ν_1 we can calculate the optimal window length l_x at frequency $\nu_x = x\nu_1$ using equation 1. Now the problem is that we have to find the optimal window length at frequency ν_1. Experiments have shown that a window length of about 0.04 seconds at a frequency of about 1,000 Hz, generally serves as a good choice when analyzing music.

The final definition of the OQSTFT is:

$$\mathbf{oqstft}\{\mathbf{s}\}(t,\nu) = \sqrt{\nu}\int_{-\infty}^{\infty}\mathbf{s}(\tau)\mathbf{w}(\sqrt{\nu}(\tau-t))e^{i2\pi\nu(\tau-t)}\mathrm{d}\tau \tag{2}$$

where OQSTFT is calculated for the signal $\mathbf{s}$ at the time t and the frequency ν, and where a window $\mathbf{w}$ is used. In the integration we integrate over time τ.

OQSTFT can be calculated fast by using the method that Brown [2] uses to calculate the Constant Q transform.

Implementation First the audio signal is downsampled to 14,700 Hz so that further calculations can be based on the frequencies up to approximately 7,000 Hz. Then a normal STFT is calculated. The window length is 92.88 ms (4,096 samples at a sample rate of 44,100 Hz), so the distance between 2 frequency bins is 10.767 Hz. Using the spectrum of a Hann window, the OQSTFT is calculated by convoluting the STFT with Hann windows of different lengths. To get an artificially high frequency resolution, OQSTFT is calculated at a frequency distance of 5.833 Hz. For frequencies below 172.27 Hz the window length is constant.

The implemented OQSTFT runs about 5 times slower than a normal STFT, so in order to speed up the calculations a multi-resolution STFT that approximates the OQSTFT could be used. A good example of a multi-resolution STFT is the one used by Dressler in [4].

Using OQSTFT instead of a normal STFT seems to improve the identification of the melody by a few percentage points especially when the fundamental frequency vibrate.

2.2 Spectral analysis using a modified SHS

The spectral analysis is based on a few modifications to the subharmonic summation (SHS) method which was introduced by Dik Hermes [5]. Inspired by the spectral normalized factor used by Cao *et al.* in [3], a simpler spectral normalized factor introduced by Wendelboe in [7] is used. Another modification called the smoothing factor is also used to prevent too many octave errors. The smoothing factor was also introduced in [7]. In the following sections the new spectral normalized factor and the smoothing factor are briefly described.

2.3 The spectral normalized factor

The idea behind the spectral normalized factor is, that a frequency component is important if it is stronger than the surrounding frequency components. So the spectral normalized factor tells how strong a frequency component is relatively to the surrounding frequency components. A function called $\boldsymbol{\rho}$ determines the width of the surroundings. For low frequencies the surroundings are narrow and for high frequencies the surroundings are wide.

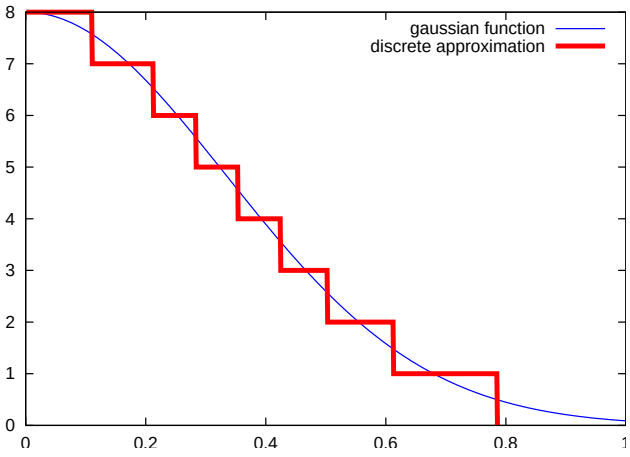

Figure 1. Here a gaussian function and a discrete approximation with 8 levels are shown.

In the following equation the spectral normalized factor is defined. The calculations are based on the power spectrum which is indicated by the symbol $\mathbf{P}$. As a symbol for the frequency, ν is used. In the integration we integrate over the frequency which is indicated by the symbol ξ. When integrating over the frequencies a window $\mathbf{w}$ is used. The window is localized at the frequency ν and has a width determined by the function $\boldsymbol{\rho}$.

$$\mathbf{P}_{norm}(\nu) = \frac{\mathbf{P}(\nu)}{2\boldsymbol{\rho}(\nu)}\int_{\nu-\boldsymbol{\rho}(\nu)}^{\nu+\boldsymbol{\rho}(\nu)}\mathbf{w}(\nu,\xi)\mathbf{P}(\xi)\mathrm{d}\xi \tag{3}$$

The integration term divided by the term $2\boldsymbol{\rho}(\nu)$ states a weighted power average. In order to speed up the calculations of the integration, a summed power spectrum is calculated and an approximation to a Gaussian function is used as window — see figure 1. For every level in the discrete approximation to the Gaussian function (the window $\mathbf{w}$ in the equation), a sort of power average is calculated by a difference in the summed power spectrum. By summing the averages for every level a sort of weighted power average is calculated. For more details see [7].

2.4 The smoothing factor

The idea behind the smoothing factor is motivated by the supposition that the frequency envelope curve of a harmonic tone often follows a smooth curve. Actually the smoothing factor is a function that calculates an estimated strength of a frequency. The function takes 2 arguments: The first argument, ν_0, is the fundamental frequency of an assumed harmonic tone, and the second argument, n, is the number of the harmonic. The function uses 2 constants, u_- and u_+, which tell how much the strength of a given harmonic depends on the strength of the neighboring

harmonics. The function is defined as:

$$\begin{aligned}\mathbf{P}_{avg}(v_0,n) =& u_{-}\mathbf{P}\big((n-1)v_0\big) \\ &+ (1-u_{-}-u_{+})\mathbf{P}\big(nv_0\big) \\ &+ u_{+}\mathbf{P}\big((n+1)v_0\big) \\ \mathbf{P}_{smooth}(v_0,n) =& \mathbf{min}\Big(\mathbf{P}_{avg}(v_0,n),\mathbf{P}(nv_0)\Big) \end{aligned} \quad (4)$$

2.5 Putting it together

First we introduce a special notation. We want to be able to specify that calculations of the smoothing factor are based on the normalized spectrum. This is specified by $\mathbf{P}_{smooth}\{\mathbf{P}_{norm}\}(v_0,n)$.

So, first we calculate a time/frequency representation by using OQSTFT. Then for every time step the spectrum, $\mathbf{P}$, is extracted and a normalized spectrum, $\mathbf{P}_{norm}$, is calculated. Using the smoothing factor as a function, the modified SHS is then calculated as:

$$\mathbf{shs}_{modified}(v_0) = \sum_{n=1}^{N} \mathbf{h}(n)\mathbf{P}_{smooth}\{\mathbf{P}_{norm}\}(v_0,n) \quad (5)$$

The modified SHS is calculated for frequencies between 100 and 1,200 Hz, at a distance of 1.3458 Hz. The function $\mathbf{h}$ is defined as: $\mathbf{h}(n) = b^{n-1}$, where $b = 0.95$. N specifies the number of harmonics and is equal to 20, because the human voice often has strong harmonics in the frequency band around 2,000 to 3,000 Hz, even if the fundamental frequency is low and around 150 Hz.

The frequency which gives the highest SHS value is chosen as the estimated fundamental frequency of the melody. The method does not detect if a melody is present in the audio signal.

3 Results

This year 4 different datasets were used in the competition. One of the datasets (ADC 2004) is public. The datasets can be described as:

- ADC 2004: 20 excerpts of about 20 sec. each from the following genres: Jazz, Midi, Pop, Opera.
- MIREX 2005: 25 phrase excerpts of 10-40 sec. from the following genres: Rock, R&B, Pop, Jazz, Solo classical piano.
- MIREX 2008: 4 excerpts of 1 min. from "north Indian classical vocal performances".
- MIREX 2009: Karaoke recordings of Chinese songs with a singing voice (male, female) and synthetic accompaniment.

Table 1, 2, 3 and 4 states the results for Raw Pitch and Raw Chroma, and table 5 states the running time of the participating implementations. The participants listed in the tables are:

- cl1: Chuan Cao, Ming Li
- dr2: Jean-Louis Durrieu, Gaël Richard (SIMM)
- hjc1: Chao-Ling Hsu, Jyh-Shing Roger Jang, Liang-Yu Chen (DP)
- jjy: Sihyun Joo, Seokhwan Jo, Chang D. Yoo
- kd: Karin Dressler
- mw: Morten Wendelboe
- pc: Pablo Cancela
- rr: Vishweshwara Rao,Preeti Rao
- toos: Hideyuki Tachibana, Takuma Ono, Nobutaka Ono, Shigeki Sagayama

This year the competition was very close. For each dataset the best performing algorithms achieved about the same score. Ranked by the Raw Chroma score the algorithm presented in this paper was among the 3 best performing algorithms when dealing with the 3 first datasets. Unfortunately the algorithm had problems with last dataset. The implementation was about 5 times slower than the fastest participating implementation, but it was still much faster than the slowest implementations.

Again this year Karin Dressler delivered the fastest and best performing implementation.

ADC 2004		
Participant	Raw Pitch	Raw Chroma
kd	87.1 %	87.6 %
jjy	83.3 %	87.0 %
mw	**82.3** %	**86.4** %
cl1	85.1 %	86.3 %
rr	76.9 %	85.1 %
dr2	81.2 %	83.6 %
pc	82.9 %	83.4 %
hjc1	63.9 %	73.6 %
toos	61.0 %	71.8 %

Table 1. Results for **ADC 2004** dataset, ordered by Raw Chroma.

MIREX 2005		
Participant	Raw Pitch	Raw Chroma
kd	76.4 %	80.9 %
mw	**75.0** %	**80.6** %
jjy	69.5 %	76.5 %
rr	69.0 %	76.4 %
dr2	70.4 %	76.0 %
toos	67.5 %	74.0 %
cl1	70.1 %	73.5 %
hjc1	59.1 %	70.8 %
pc	68.0 %	70.4 %

Table 2. Results for **MIREX 2005** dataset, ordered by Raw Chroma.

MIREX 2008		
Participant	Raw Pitch	Raw Chroma
mw	**86.0** %	**88.9** %
kd	87.8 %	88.8 %
dr2	86.6 %	86.8 %
rr	86.2 %	86.7 %
toos	79.8 %	83.7 %
pc	81.8 %	82.0 %
jjy	68.3 %	81.9 %
hjc1	67.6 %	74.9 %
cl1	50.8 %	51.3 %

Table 3. Results for **MIREX 2008** dataset, ordered by Raw Chroma.

MIREX 2009 mixed at 0dB		
Participant	Raw Pitch	Raw Chroma
toos	82.3 %	85.7 %
kd	80.5 %	81.9 %
jjy	75.9 %	80.2 %
hjc1	72.7 %	75.3 %
rr	68.6 %	71.4 %
mw	**67.3** %	**71.0** %
dr2	66.6 %	70.8 %
cl1	59.1 %	63.0 %
pc	50.9 %	53.4 %

Table 4. Results for **MIREX 2009** dataset, ordered by Raw Chroma.

4 Conclusion

Given the simplicity of the algorithm the achieved results are good. The algorithm is relatively fast and performs well on 3 out of the 4 datasets. During the development of the algorithm it was tested on a dataset consisting of western popular music, so it was not expected that the algorithm performed well on the dataset consisting of karaoke recordings of Chinese songs.

5 References

[1] J. C. Brown, "Calculation of a constant q spectral transform," *Journal of Acoustic of Society of America*, vol. 89(1), pp. 425–434, 1991, http://www.wellesley.edu/Physics/brown/pubs/cq1stPaper.pdf.

[2] J. C. Brown and M. S. Puckette, "An efficient algorithm for the calculation of a constant q transform," *Journal of Acoustic of Society of America*, vol. 92(5), pp. 2698–2701, 1992, http://www.wellesley.edu/Physics/brown/pubs/effalgV92P2698-P2701.pdf.

[3] C. Cao, M. Li, J. Liu, and Y. Yan, "Multiple f0 estimation in polyphonic music (mirex 2007)," Thinkit Speech Lab., Institute of Acoustics, Chinese Academy of Sciences, Tech. Rep., 2007, http://www.music-ir.org/mirex/2007/abs/F0_cao.pdf.

Running time	
Participant	time
kd	24 min
rr	26 min
cl1	28 min
mw	**132 min**
hjc1	344 min
dr2	524 min
toos	1468 min
jjy	3726 min
pc	4677 min

Table 5. Ordered running time. Notice that different machines have been used, but the machines have comparable speed.

[4] K. Dressler, "Sinusoidal extraction using an efficient implementation of a multi-resolution fft," Fraunhofer Institute for Digital Media Technology, Tech. Rep., 2006, http://www.dafx.ca/proceedings/papers/p_247.pdf.

[5] D. J. Hermes, "Measurement of pitch by subharmonic summation," *Journal of Acoustic of Society of America*, vol. 83, pp. 257–264, 1988.

[6] S. G. Mallat and Z. Zhang, "Matching pursuits with time-frequency dictionaries," *IEEE Transactions on Signal Processing*, vol. 41, no. 12, pp. 3397–3415, 1993, http://www.cmap.polytechnique.fr/~mallat/papiers/MallatPursuit93.pdf.

[7] M. Wendelboe, "Bestemmelse af melodien i polyfon musik," Master's thesis, Datalogisk Institut, Københavns Universitet, 2009, http://morwen.dk/university/master-thesis/speciale.pdf.