DELIVERABLE SUBMISSION SHEET

To: Claude Poliart *(Project Officer)*

EUROPEAN COMMISSION
DG Infso E3, Cultural heritage and technology enhanced learning
EUFO 1154
Rue Alcide de Gasperi
L-2920 Luxembourg

From: Project name: Enabling Access to Sound Archives through Integration, Enrichment and Retrieval

Project acronym: EASAIER Project number: 033902

Person: Joshua Reiss

Organisation Queen Mary University of London

Date 10 September 2008

The following deliverable:

Deliverable title: Prototype transcription system

Deliverable number: D.4.2

is now complete.
- ☐ It is available for your inspection.
- ☐ Relevant descriptive documents are attached.
- ■1 copy herewith.
- ☐ 2 copies herewith

Tick all that apply

The deliverable is:) on paper

× on (url:www.easaier.org)

For all paper deliverables, *and other deliverables as appropriate:*

Date of del: 15 September 2008 Version: 1

Author: Ruohua Zhou No. of pages: 41

Status: x Public ☐ Restricted ☐ Confidential *(tick one)*

Sent electronically to Claude Poliart	Functional mail box	Claude.Poliart@ec.europa.eu	ON DATE	15/08/08
ATTESTATION I confirm that this electronic version is the same as the paper version submitted herewith.				
NAME	Betty Woessner	ORGANISATION	QMUL	
SIGNATURE				
DATE	15 September 2008			

D4.2

Prototype transcription system

Abstract

Deliverable 4.2 describes prototype transcription systems and the related key technologies (music onset detection and multiple pitch estimation).

The resonator time frequency image (RTFI) is the basic time-frequency analysis tool. The introduced onset detection method consists mainly of two elements: the time-frequency processing and the detection stages. Two detection algorithms have been developed: an energy-based algorithm and a pitch-based one. The energy-based detection algorithm exploits energy-change cues and performs particularly well for the detection of hard onsets. The pitch-based algorithm successfully exploits stable pitch cues for the onset detection in polyphonic music, and achieves much better performances than the energy-based algorithm when applied to the detection of soft onsets. Results for both the energy-based and pitch-based detection algorithms have been obtained on a large music dataset.

A computationally efficient method for multiple pitch estimation is also described. The method employs the Fast RTFI as the basic time-frequency analysis tool. First, a preliminary pitch estimation is obtained by means of a simple peak-picking procedure in the pitch energy spectrum. Such spectrum is calculated from the original RTFI energy spectrum according to harmonic grouping principles. Then the incorrect estimations are removed according to spectral irregularity and knowledge of the harmonic structures of the music notes played on commonly-used music instruments. The approach is compared with a variety of other frame-based polyphonic pitch estimation methods, and results demonstrate the high performance and computational efficiency of the approach.

Version: 1
Date: 12 September 2008
Editor: QMUL
Contributors: QMUL

Table of Contents

1. Document History

Version	Date	Author	Description
v1	Sep. 01, 2008	Ruohua Zhou (QMUL)	Created – Table of contents, and dedicated to Partners

2. Introduction

2.1. What is music transcription

Music transcription is here defined as an act of analyzing a piece of music signal and writing down the parameter representations, which indicate the pitch, onset time and duration of each pitch. Automatic music transcription is a process to convert an acoustical waveform into a musical notation (such as the traditional western music notation) by computer programming. In most cases automatic transcription of common monophonic music can be considered as a matured filed, however the computational transcription of polyphonic music has less relative success. Transcription of polyphonic music is a very difficult task for the average human without training, however human can improve their transcription ability by learning. Skilled musicians are able to transcribe polyphonies with much better performance than computational transcription system can achieve in current research phases. The automatic music transcription is a critical step for some higher level music analysis tasks such as melody extraction, rhythm tracking, and harmony analysis.

2.2. Significance of music transcription

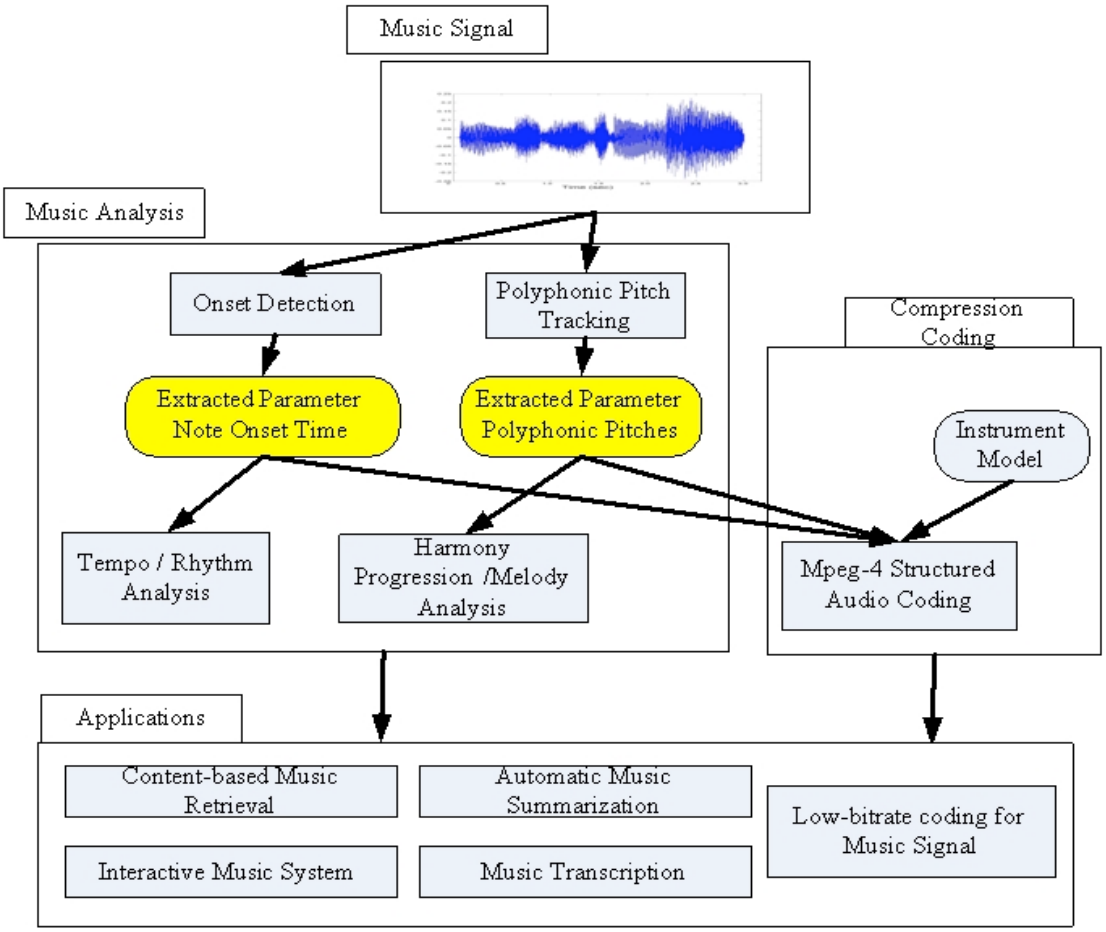

Figure 1 Essential role of the extracted parameters: onset time and polyphonic pitch in music analysis

The music consists of some basic elements. These are rhythm, melody, harmony, and timbre. The aim of the transcription system described in this document is to extract two important

parameters: note onset time and polyphonic pitch. As shown in Figure 1, the extraction of the both parameters plays an essential role in a music analysis. The extracted onset time of music notes can be used to chiefly determine rhythm. The extracted multiple pitches of music notes can be primarily employed for the detection of melody and harmony. Melodic lines are sequences of notes over time, and harmony is determined by the relationship between the multiple simultaneously occurring pitches of music notes.

Because the extraction of note onset time and polyphonic pitch is a fundamental stage of analyzing the basic elements of music signals, it can be utilized to support broad music applications. As shown in the Figure 1, except the automatic music transcription itself, the possible applications include content-based music retrieval, automatic music summarization, interactive music system, low-bitrate compression coding for music signal and so on. Multimedia music applications are nowadays rapidly moving from simple content related scenarios to more complex and sophisticated domains including content, interaction, related descriptions and annotations, item identification. In the case of content-based music retrieval, automatic onset and harmonic information extraction is crucial. Not only it is interesting to build measures of harmonic similarity between musical excerpts, but it can further help some other kind of analysis such as rhythm or instrument detection by finding onset time points where such events or instrument note starts are more likely to be observed. Another interesting point that is gaining importance in the domain is the automatic control of signal processing parameters according to content features and builds some music interactive systems. Another possible application of the extraction of polyphonic pitch is to assist for the low-bitrate coding for music signal. The MPEG-4 Structured Audio coding provides new methods for low-bitrate storage. In a framework of Structured Audio coding, automatic music transcription and music synthesis play chief role.

3. Music onset detection

3.1. Introduction

A music signal can be considered as a succession of musical events (notes). Music onset detection aims at finding the starting time of each note. Music onset detection plays an essential role in music signal processing and has a wide range of applications such as music transcription, beat-tracking, and tempo identification. Many different onset detection systems have been described in the literature. Typically they consist of three stages: time-frequency processing, detection function generation, and peak-picking [1]. At first, a music signal is transformed into different frequency bands by using a filter-bank or a spectrogram. Then, the output of the first stage is further processed to generate a detection function at a lower sampling rate. Finally, a peak-picking operation is used to find onset times within the detection function, which is often derived by inspecting the changes in energy, phase, or pitch.

In the past, differences in a signal's envelop were used to detect note onsets. However, such approach has been proved to be inefficient. Some researchers have found it useful to separate the analyzed signal into several frequency bands and then detect onsets across the different frequency bands. This constitutes the key element of the so-called multi-band processing. The first-order difference of energy or amplitude has been utilized to derive a detection function. However, the first-order difference is usually not able to precisely mark onset times. According to psychoacoustic principles, a perceived increase in the signal amplitude is relative to its level. The same amount of increase can be perceived more clearly in a quiet signal. Consequently, as a refinement, the relative difference can be used to better locate onset times [2].

Phase-based approaches detect onsets by using phase information [3]. The STFT of the signal can be considered to be a group of sinusoid oscillators. In the steady-state parts of the signal, the frequency of each oscillator tends to remain constant. This is not the case in the transients.

Therefore, the change in frequency is an indicator of a possible onset. The second difference of the phase of the oscillator is able to identify the change in its frequency. Accordingly, statistics (e.g., mean, variance, kurtosis) on the second difference of the phase can be calculated across the range of frequencies and used to derive the detection function. To detect soft onsets, phase-based approaches perform better than standard energy-based approaches. However, they are susceptible to phase distortion and to noise introduced by the phases of low-energy components. The combination of phase and energy on the complex domain can provide more robust detection [4].

The approaches that only use the information of energy and/or phase are not satisfactory for the detection of soft onsets. Pitch-based detection appears as a promising solution for the problem. Pitch-based approaches can use stable pitch cues to segment the analyzed signal into transients and steady-state parts, and then locates onsets only in the transients. Such approaches are expected to greatly reduce false positives. A pitch-based onset detection system is described in [5]. In the system, an independent constant-Q pitch detector provides pitch tracks that are used to find likely transitions between notes. For the detection of soft onsets, such system performs better than other state-of-the-art approaches. However, it is designed only for the onset detection of monophonic music.

Different sound sources (instruments) have different types of onsets that are often classified as “soft” or “hard”. Hard onsets are characterized by sudden increases in energy, whereas soft onsets show more gradual changes. Hard onsets can be well detected by energy-based approaches, but the detection of soft onsets remains a challenging problem. Let us suppose that a note consists of a transient, followed by a steady-state part, and the onset of the note is at the beginning of the transient. For hard onsets, usually, energy changes are significantly larger in the transients than in the steady-state parts. Conversely, when considering the case of soft onsets, energy changes in the transients and the steady-state parts are comparable, and they do not constitute reliable cues for onset detection any more. Consequently, energy-based approaches fail to correctly detect soft onsets. Stable pitch cues enable to segment a note into a transient and a steady-state part, because the pitch of the steady-state part often remains stable. This fact can be used to develop appropriate pitch-based methods that yield better performances, for the detection of soft onsets, than energy-based methods. However, only a few pitch-based methods have been proposed in the literature, although many approaches have already used energy information. In the EASAIER research, we develop a new method that makes best use of both energy-change and pitch-change information.

3.2. Onset detection method

As shown in Figure 2, the method consists of three main stages: Time-frequency processing, onset type classification and detection algorithms.

3.2.1. Resonator Time-Frequency Image

A Frequency-Dependent Time-Frequency (FDTF) analysis may be defined as follows:

$$FDTF(t,\omega) = \int_{-\infty}^{\infty} s(\tau) w(\tau - t, \omega) e^{-j\omega(\tau - t)} d\tau \,. \quad (1)$$

Unlike the STFT, the window function w of an FDTF may depend on the analytical frequency ω. This means that time and frequency resolutions can be tuned according to the analytical frequency.

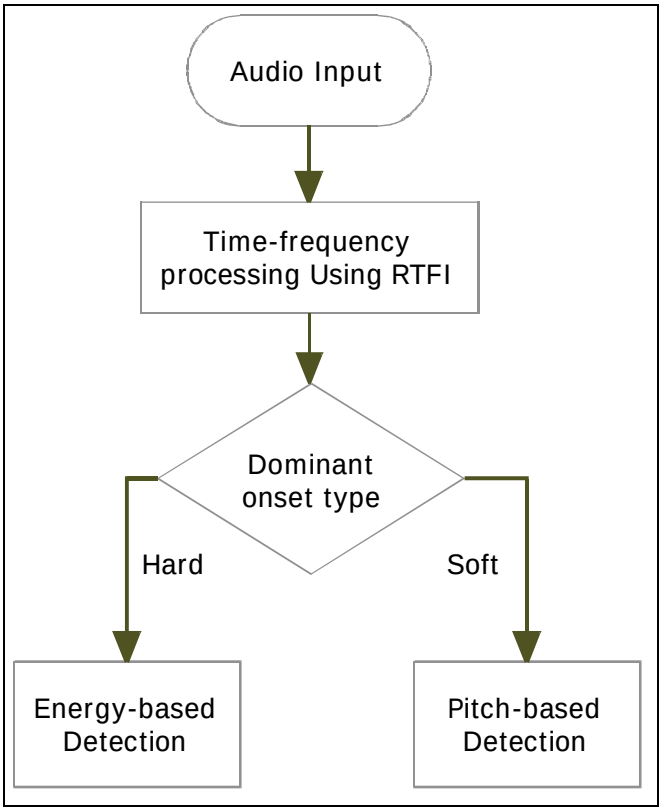

Figure 2 System overview of Music onset detection combining energy-based and pitch-based approaches

Equation (1) can also be expressed as:

$$FDTF(t,\omega) = s(t) * I(t,\omega) \tag{2}$$

where

$$I(t,\omega) = w(-t,\omega)e^{j\omega t}. \tag{3}$$

Equation (1) is more suitable to express a transform-based implementation, whereas equation (2) leads to a straightforward implementation of a filter bank with impulse response functions expressed by equation (3).

A novel time-frequency representation, known as the Resonator Time-Frequency Image (RTFI), has been developed. Its main feature is that it selects a first-order complex resonator filter bank to implement a frequency-dependent time-frequency analysis. This was chosen due to the flexibility with regards to time and frequency resolution, and the simplicity and computational efficiency of an implementation based on first order filters.

The Resonator Time-Frequency Image (RTFI) can be described as follows:

$$RTFI(t,\omega) = s(t) * I_R(t,\omega) \tag{4}$$

$$= r(\omega)\int_0^t s(\tau)e^{r(\omega)(\tau-t)}e^{-j\omega(\tau-t)}d\tau$$

where

$$I_R(t,\omega) = r(\omega)e^{(-r(\omega)+j\omega)t}, \qquad t > 0. \tag{5}$$

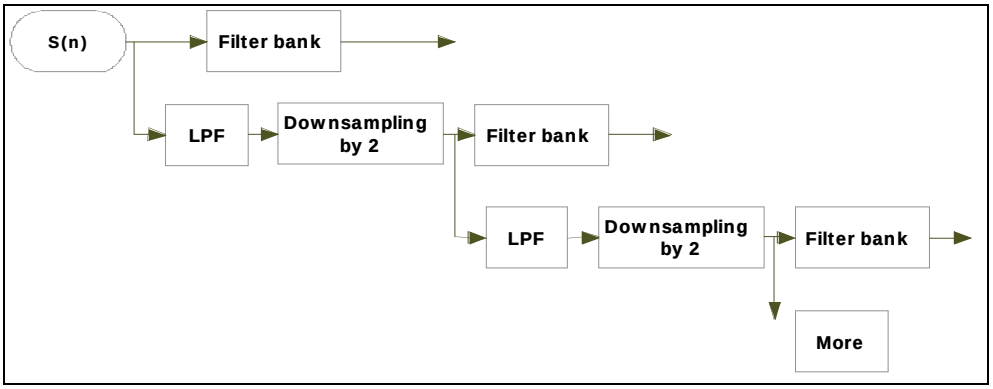

Figure 3 Block diagram of the multi-resolution implementation

In the above equations, I_R denotes the impulse response of the first-order complex resonator filter with oscillation frequency ω, and the factor $r(\omega)$ before the integral in the equation (4) is used to normalize the gain of the frequency response when the resonator filter's input frequency is the oscillation frequency. The decay factor r is dependent on the frequency ω and determines the exponent window length and the time resolution. It also determines the bandwidth (i.e. the frequency resolution).

Since the RTFI has a complex spectrum, it may be expressed as follows:

$$RTFI(t,\omega) = A(t,\omega)e^{j\varphi(t,\omega)} \tag{6}$$

where $A(t,\omega)$ and $\varphi(t,\omega)$ are real functions. The energy of the signal may then be given by

$$RTFI_{Energy}(t,\omega) = \left|A(t,\omega)\right|^2. \tag{7}$$

In this work, it is proposed to use the first order complex resonator digital filter bank to implement a discrete RTFI. To reduce the memory requirements needed to store the RTFI values, the RTFI is separated into different time frames and the average RTFI values are calculated in each frame. Finally the average RTFI energy is used to track the time-frequency characteristics of the music signal. The average RTFI energy spectrum can be expressed as follows:

$$ARTFI(g, f_k) = dB\left(\frac{1}{M}\sum_{n=J_g}^{J_g+M-1}\left|RTFI(n, f_k)\right|^2\right) \tag{8}$$

where M is the number of sample in the time frame, g is the index of frame, dB() converts the value to decibels and the ratio of M to sampling rate is the duration time of the frame in the averaging process. RTFI(n, f_k) denotes the value of the discrete RTFI at sampling point n and frequency f_k, and J_g denotes the frame which begins at the J_g^{th} sample of the analyzed signal.

The Fast RTFI is used to reduce the redundancy in computation. In some cases it is not necessary to keep the same sampling frequency of the input for every filter in the filter bank. For the filters with lower center frequencies, the sampling rate can be decreased. In the fast implementation, the filter bank is separated into different octave frequency bands. The inputs of the filter banks in the same frequency band maintain the same sampling rate. The input signal is recursively low-pass filtered and down sampled by a factor of 2 from the highest to the lowest frequency band according to the scheme depicted in Figure 3.

This subsection has briefly introduced the basic idea behind RTFI analysis. A more detailed description of the discrete RTFI and its fast implementation can be found in [6, 7].

3.2.2. Time-Frequency Processing

The monaural music signal is used as the input signal at a sampling rate of 44.1 kHz. The method utilized RTFI as the basic tool for time-frequency processing. The center frequencies of discrete RTFI are according to logarithmic scale and the resolution is selected as constant-Q. 10 filters are used to cover the frequency band of one semitone and there is a total of 960 filters in the analyzed frequency range, which extends from 46 Hz to 6.6 kHz. The average RTFI energy spectrum is calculated in unit of 0.01 second and it is used to track the time-frequency character of music signal.

To remove the low-frequency noise, the RTFI energy spectrum is first transformed into the adjusted energy spectrum (AES) according to the Robinson and Dadson equal-loudness contours, which have been standardized in the international standard ISO-226. In order to simplify the transformation, only an equal-loudness contour corresponding to 70db is used to adjust the RTFI energy spectrum. The standard provides equal-loudness contour limited to 29 frequency bins. Then, this contour is used to obtain the equal-loudness contour of 960 frequency bins by cubic spline interpolation in the logarithmic frequency scale. Let us define this equal-loudness contour as $Eq(\omega_m)$ in db. Then, the adjusted energy spectrum (AES) can be expressed as follows:

$$AES(k,\omega_m) = ARTFI_E(k,\omega_m) - Eq(\omega_m) \tag{9}$$

In the adjusted energy spectrum, one can select a threshold value of the energy spectrum below which it will be considered as a noise spectrum. Then adjusted energy spectrum is further transformed into the pitch energy spectrum Y , smoothed pitch energy spectrum R, difference pitch energy spectrum D and the normal pitch energy spectrum F according to the following equations:

$$R(k,\omega_m) = \frac{1}{5}\sum_{i=1}^{5} AES(k, i\cdot\omega_m) \tag{10}$$

$$S(k,\omega_m) = \frac{1}{25}\sum_{i=k-2}^{k+2}\sum_{m-2}^{m+2} R(k,\omega_m) \tag{11}$$

$$D(k,\omega_m) = S(k,\omega_m) - S(k-n,\omega_m) \tag{12}$$

$$F(k,\omega_m) = S(k,\omega_m) - \max((S(k,\omega_m))_{m=1:N}) \tag{13}$$

where n is the difference order and N is the total number of frequency bins in the spectrum F .

In practical cases, instead of using equation (3), the spectrum R can be easily calculated in the logarithm scale by the following approximation:

$$R(k,\omega_m) \approx \frac{1}{5}\sum_{i=1}^{5} Y(k,\omega_{m+A[i]}) \tag{14}$$

$$A[5] = [0,120,190,240,279] \tag{15}$$

The difference pitch energy spectrum makes the energy change more obvious, and the normal pitch energy spectrum makes pitch change more clearly.

3.2.3. Detection Algorithms

3.2.3.1. Onset Type Classification

A simple way is used to classify the dominant onset type of the analyzed input file. The measure of the onset ‘hardness’ is defined as follows:

$$Q(k) = mean(H(D(k,\omega_m))) \tag{16}$$

$$HM = mean(Q(k)) \tag{17}$$

where $H(x) = (x + |x|)/2$ is the half-wave rectifier function, and spectrum D is calculated with first order difference. The hardness measure HM is used to classify the dominant onset type. If the HM of the analyzed input file is more than a threshold, the onset type of this input is considered as hard, otherwise it is considered to be soft.

For the input with hard onset type, the energy-based algorithm is used to find onsets; conversely, the pitch-based algorithm is utilized.

3.2.3.2. Energy-based detection

A music signal is assumed into two parts - a transient part and a steady-state part. The difference pitch energy spectrum D can be used to track the transient information and generate an energy-based detection function as follows.

$$L(k,\omega_m) = H(D(k,\omega_m) - \theta_1), \quad \theta_1 > 0 \tag{18}$$

$$DF(k) = mean(L(k,\omega_m)) \tag{19}$$

where $H(x) = (x + |x|)/2$ is the half-wave rectifier function, and DF represents the energy-based detection function. The spectrum D is calculated with 3-order difference.

In the energy-based algorithm, firstly the difference pitch energy spectrum is limited by a threshold θ_1 so that only the energy-change values that exceed threshold θ_1 are considered to be possible transient clues; and then it is averaged across all frequency channels to generate the detection function. The detection function is further smoothed by a moving-average filter and a simple peak-picking operation is used to find the note onsets. In the peak-picking, another threshold θ_2 needs to be set and only the peaks having values greater than threshold θ_2 are considered as the possible onset candidates. In the final step, if there are two onset candidates and the position difference between them is smaller than or equal to 50ms, then only the onset candidate with the greater value will be kept.

3.2.3.3. Pitch-based detection

Generally speaking, energy-based detection methods are not good at detecting soft onsets. Consequently, a pitch-based algorithm has been developed. In the proposed pitch-based detection algorithm, the music signal is first divided into transient and stable parts by the stable pitch cue, and then the onset is located in the transient part by energy-change. As the output of RTFI time-frequency processing, the spectrum D and the spectrum F are used together as the input for this detection algorithm. The algorithm can be separated into two steps:

1) Searching the possible note onsets with the approximate fundamental frequency ω_{m1}.

2) Combining the detected onset candidates across all of the frequency channels and generating the final result for onset detection.

In the first step, the algorithm searches possible note onsets in every frequency channel. It is emphasized that, when searching in a certain frequency channel with frequency ω_{m1}, the detection algorithm tries to find only the onset where the new occurred pitch rightly has an approximate fundamental frequency ω_{m1}.

If a pitch with a fundamental ω_{m1} occurs in a certain time segment, then there is often a peak line in this time segment around the frequency ω_{m1} in the spectrum F and its value is nearly equal to 0db and relatively larger than the other frequency bins. This fact has been observed in our experiments.

When searching for onsets in a certain frequency channel with frequency ω_{m1}, the detection algorithm first tries to find the "stable time segment T ", which corresponds to the steady-state part of a music note. Let us suppose the time segment $T[k_1,k_2]$ represents a time duration from k_1 to k_2 in units of 10ms. Given a time-frequency spectrum $F(k, \omega_m)$, if a time segment $T[k_1,k_2]$ meets with the following three conditions, the time segment T is assumed to be stable.

$$(F(k,\omega_m))_{m=m1,k=k1:k2} > \alpha_1 \tag{20}$$

$$\max((F(k,\omega_m))_{m=m1,k=k1:k2}) > \alpha_2 \tag{21}$$

$Sum(\omega_m)$ has a spectral peak at the frequency ω_{m1},

$$Sum(\omega_m) = \sum_{k=k_1}^{k_2} F(k,\omega_m) \tag{22}$$

For each stable time segment $T[k_1,k_2]$, the algorithm looks backward from beginning of the stable time segment T and locates the onset time in 300ms window by searching salient energy-increasing in the duration $[k_1\text{-}300, k_1]$. The salient energy-increasing is defined by peak-picking in the different pitch spectrum D in the duration $[k_1\text{-}300\text{ms}, k_1]$ at the frequency ω_{m1}. The threshold α_3 of the peak-picking process is the third important parameter for this algorithm.

After all frequency channels have been searched, the pitch onset candidates can be found and expressed as follows:

$$Onset\_C(k,\omega_m) \geq 0\ ,\ m=1,\ 2,\ 3,\ \ldots N, \tag{23}$$

where k denotes the time frame and N denotes the total num of the pitch channels. If $Onset\_C(k, \omega_m)=0$, no onset exists in the k_{th} time-frame and m_{th} frequency channel. If the $Onset\_C(k, \omega_m)>0$, there is an onset candidate in the k_{th} time-frame and m_{th} frequency channel, and the value of $Onset\_C(k, \omega_m)$ is equal to the value of spectrum $D((k, \omega_m)$.

Finally the detection algorithm combines the pitch onset candidates across all the frequency channels to get the final onset. If two onset candidates are neighbours in a 0.05 second time window, then only the onset candidate with the greater value will be kept.

3.2.4. *Evaluation*

The described onset detection method was submitted to MIREX 2007 for evaluation [8]. According to the overall performance, our method wins this contest (reported in Table 1). Different methods can perform significantly better for different classes. Our method performs better than the other methods for the classes: solo drum, solo brass and solo wind.

In the MIREX 2005~2007 onset detection contests, most of the submitted methods are energy-based. The results suggest that a common difficulty exists in the onset detection of the classes: solo brass, solo wind, solo sustained string and solo singing voice. These classes usually contain a large number of soft onsets. Energy-based approaches are based on the assumption that there are relatively more salient energy changes at the onset times than in the steady-state parts. In case of soft onsets, the assumption can not stand. The significant energy changes in the steady-state parts can mislead energy-based approaches and cause many false positives. According to our previous experiments, the pitch-based detection can clearly outperform the energy-based detection for the detection of soft onsets.

For the solo brass and solo wind, our method outperforms the second best methods by about 8% and 9% respectively. Such performances can be contributed to the combination of the pitch-based detection. For the single sustained string class, the method is the second best one and Lee's method is better. This can be explained that Lee's method is not only energy-based but also

combines a phase-based detection, which uses the stable pitch cues indirectly [9]. For the solo singing voice class, our method performs not well. The reason is that the method is developed on the music datasets played by musical instruments. In the steady-state parts, the pitch variation of instrumental music is minor, but the singing voice′s pitch variation relatively larger. The method need be improved to achieve a better performance on the onset detection of signing voice.

Table 1: Overall results of Mirex 2007 onset detection task for the submitted method	
Overall Average F-measure	80.8%
Overall Average Precision	85.7%
Overall Average Recall	78.2%
Total Correct	7225
Total False Positives	1186
Total False Negatives	2130
Total Merged	189
Total Doubled	49
Runtime (s)	1399

4. Multiple Pitch Estimation

4.1. Introduction

Polyphonic pitch estimation plays an important role in music signal analysis. It can be essentially used for the detection of musically relevant features such as melody and harmony. In the case of content-based music retrieval, the "automatic" extraction of melody information is a crucial element for any music retrieval system. Another potential application is assisting the structured audio coding.

A number of approaches have been proposed in the literature. Klapuri proposed a polyphonic pitch estimation algorithm based on an iterative method, which was further explored for music transcription. In such method, first the predominant pitch of concurrent musical sound is estimated. Then the spectrum of the sound with the predominant pitch is estimated and subtracted from the mixture. The estimation and subtraction is repeated iteratively on the residual signal.

Recognizing a note in note-mixtures is a typical pattern recognition problem. Therefore, some approaches transform the polyphonic pitch estimation into a pattern recognition problem, which is then solved by employing machine learning methods such as neural networks and support vector machines. Other methods such as Bayesian inference, sparse coding, and nonnegative matrix factorization have also been investigated.

This subsection describes a computationally efficient method for polyphonic pitch estimation. The method consists of time-frequency analysis and post-process phases. For both phases, novel techniques are used to increase computational efficiency. In the post-process phase, neither iterative processing nor machine learning is needed. First, a preliminary estimation is used to find all possible pitch candidates, which may include extra estimations. Then the incorrect estimations are removed according to the spectral irregularity and knowledge of the harmonic structures. The post-process phase mainly involves pick-peaking, addition and subtraction operations, and the computational overload is negligible. Accordingly, the computational cost of the

method chiefly depends on the time-frequency analysis part. The constant-Q Fast Resonator Time-Frequency Image (RTFI) has been selected as the basic time-frequency analysis tool. RTFI is employed here mainly because it can be implemented by the simplest filter banks. In addition, fast implementations of such filter banks can also further improve the computational efficiency.

As a result, the overall approach is 3 times faster than real time on a standard PC equipped with a 2.0 GHz Pentium processer. The method was also evaluated in the multiple fundamental frequency frame level estimation task of MIREX 2007. The achieved results demonstrate the high performance and computational efficiency of the new approach. The method was the fastest and ranks third place in overall performance of the 16 submitted systems. Compared to the state-of-the-art approaches, it is more than 13 times faster, and has only slightly worse performance. (the accuracy of state-of-the-art method is 60.5%, whereas our method's accuracy is 58.2%).

4.2. Motivation for Selecting Constant-Q Time-Frequency Analysis

Resolution is a key factor of any time-frequency analysis. In the following, it is explained how it may be reasonable to select a nearly constant-Q resolution for a general-purpose music analysis system. Using the MIDI (Music Instrument Digital Interface) note numbers, the fundamental frequency and corresponding partials of a music note *k'* can be described as:

$$f_{k'}^{0} = 440 \cdot (2^{\frac{k'-69}{12}}) \quad \text{and} \quad f_{k'}^{m} = m \cdot f_{k'}^{0},\ k' \geq 1. \tag{24}$$

Supposing that the energy of every music note is mainly distributed over the first 10 partials, thus $Energy(f_{k'}^{m}) \approx 0$ for $m \geq 11$, the frequency ratio between the partials of one note and the fundamental frequencies of other notes can be expressed as follows:

$$2f_{k'}^{0} = f_{k'+12}^{0}\ , \quad 3f_{k'}^{0} / f_{k'+19}^{0} = 0.9989\,, \quad 4f_{k'}^{0} = f_{k'+24}^{0},\ 5f_{k'}^{0} / f_{k'+28}^{0} = 1.0079\,, \quad 6f_{k'}^{0} / f_{k'+31}^{0} = 0.9989$$
$$7f_{k'}^{0} / f_{k'+34}^{0} = 1.018,\,, \ 8f_{k'}^{0} = f_{k'+36}^{0},\ 9f_{k'}^{0} / f_{k'+38}^{0} = 0.9977\,,\ 10f_{k'}^{0} / f_{k'+40}^{0} = 1.008$$

This means that the first 10 partials always overlap with another fundamental frequency. Since the fundamental frequencies follow an exponential law (9), most of the energy is concentrated in frequency bins that are evenly spaced on a logarithmic axis. This is the reason for which the required resolution is constant-Q.

4.3. Motivation for Selecting Fast RTFI

The proposed method is manly used for polyphonic pitch tracking, where a joint time-frequency analysis is first needed. Either filter bank or constant-Q transform can be used to compute constant-Q time-frequency spectrum. As RTFI is implemented by the simplest filter bank, it is faster than any other filter-bank-based implementation. The Fast RTFI is also compared with transform-based implementations as follows.

So as to use a constant-Q transform for a joint time-frequency analysis, the time signal needs to be cut into different frames and then a constant-Q transform is performed in each frame [20]. It is assumed that the pitch tracking can report pitches every 10ms, so the time interval between two successive frames is set as 10ms. To perform a constant-Q time-frequency analysis for a 1-second signal, the constant-Q transform needs to be calculated 100 times, and the required number of complex multiplies can be expressed as:

$$N_{cq} = 100 \cdot \frac{Qf_s}{f_{\min}} \frac{1 - 0.5^{N_1}}{1 - 0.5^{1/N_2}} \tag{25}$$

where Q is the constant ratio of frequency to resolution, f_s is the sampling rate, f_{min} is the lowest analytical frequency, N_1 is the number of octave bands, and N_2 is the number of frequency components in one octave band. A fast constant-Q transform has been proposed in [21]. It employs an FFT to calculate constant-Q transform. When the fast constant-Q transform is used for time-frequency analysis of a 1-second signal, the required number of complex multiplies can be roughly expressed as:

$$N_{fcq} = 100 \cdot N_{fft} \cdot \log(N_{fft}) \ , \ N_{fft} = \frac{Qf_s}{f_{\min}} \tag{26}$$

For the Fast RTFI analysis of a 1-second signal, the required number of complex multiplies can be roughly obtained as:

$$N_{fr} = 2 f_s N_2 (1 - 0.5^{N_1}) \tag{27}$$

In the proposed method, the constant-Q factor Q is set as 17, the lowest analysis frequency f_{min} is 26 Hz, the number of octave bands N_1 is 9, and the number of frequency components in one octave band is equal to 120. Accordingly, for constant-Q analysis of a 1-second signal, Fast RTFI implementation needs approximately 240*f_s complex multiplies, constant-Q transform implementation needs approximately 24900*f_s, and fast constant-Q transform implementation needs approximately 2000*f_s. The comparison clearly suggests that Fast RTFI implementation is also much faster than transform-based implementation for a constant-Q time-frequency analysis.

4.4. Description of the Multiple Pitch Estimation Method

4.4.1. System Overview

Figure 2 provides an overview of the new polyphonic pitch estimation method. It can be conceptually partitioned into five different steps. First, a time-frequency processing based on the fast multi-resolution RTFI analysis is performed. Harmonic components are then extracted by transforming the RTFI average energy spectrum into a relative energy spectrum. Similarly, preliminary estimates of the possible multiple pitches are found by a simple peak-picking procedure in a relative pitch energy spectrum, which is obtained from the RTFI average energy spectrum.

Then a confidence measure is employed to remove pitch candidates whose harmonic components are not strongly represented. Finally, the pitches are found by investigating the spectral irregularity of the remaining candidates. These five steps are described in detail in the following subsections.

4.4.2. Detailed Description

1) Time-frequency Processing Based on the RTFI Analysis:

In the first step, the Fast RTFI is used to analyze the input music signal and to produce a time-frequency energy spectrum. The input sample is a monaural music signal frame at a sampling rate of 44.1 kHz. All 1080 filters are used. The centre frequencies are set on a logarithmic scale. The centre frequency difference between two neighbouring filters is equal to 0.1 semitone, and the analyzed frequency range is from 26 Hz up to 13 kHz. Then, the time-frequency energy spectrum of the input frame is used to obtain an RTFI average energy spectrum according to equation (8). This RTFI average energy spectrum is used as the only input vector for later processing. An integer k is used to denote the frequency index on a logarithmic scale, and f_k denotes the corresponding frequency value expressed in Hz in the equation:

$$f_k = 440 \cdot 2^{(k-690)/120} . \tag{28}$$

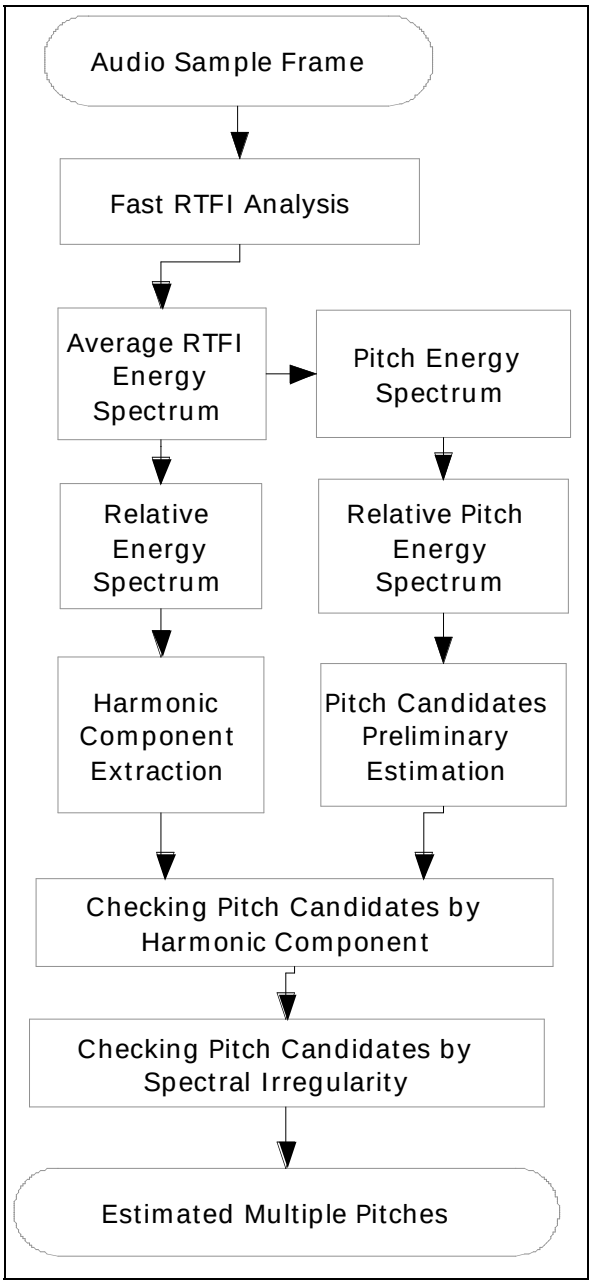

Figure 4 System overview of the computationally efficient multiple pitch estimation method

Equation (28) has been derived from the fundamental frequencies of musical notes on the western music scale. One example for the input RTFI average energy spectrum of a piano note is provided in Figure 3.

2) Extraction of Harmonic Components: In the second step, the input RTFI average energy spectrum is first transformed into the relative energy spectrum (RES) according to the following expression:

$$RES(f_k) = ARTFI(f_k) - \frac{1}{N_1 + 1} \sum_{i=k-N_1/2}^{k+N_1/2} ARTFI(f_i) \tag{29}$$

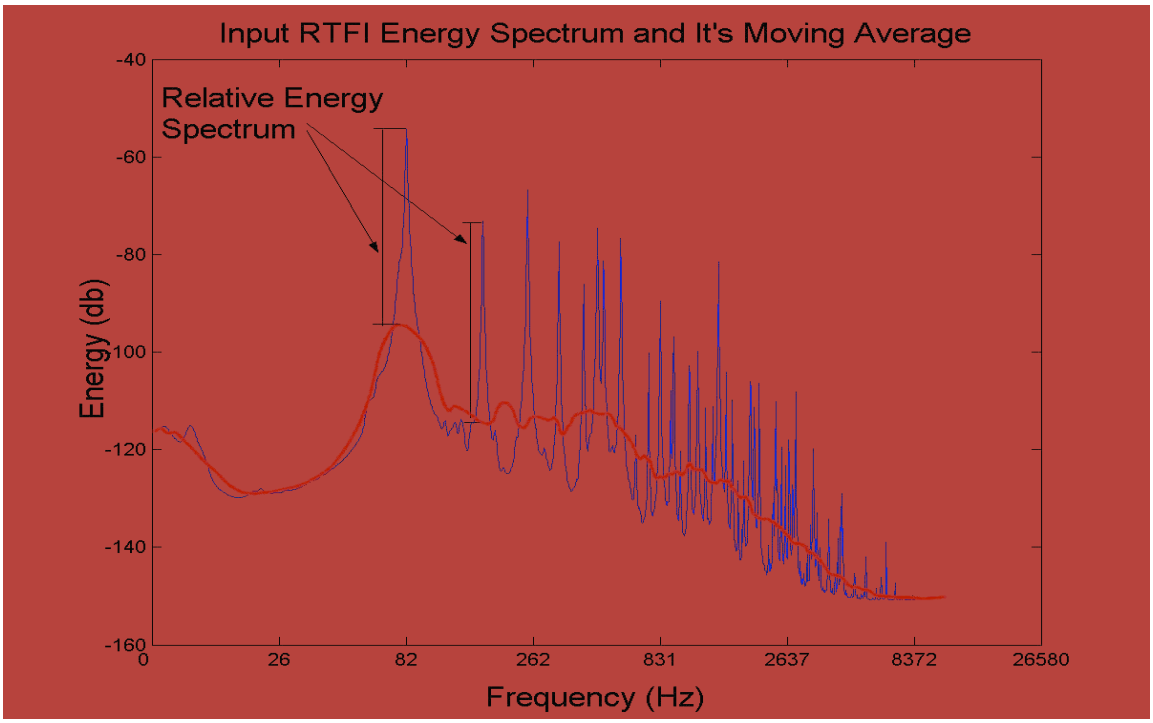

Figure 5 The input RTFI energy spectrum and the relative energy spectrum of a piano polyphonic note consisting of two concurrent notes with fundamental frequencies 82 Hz and 466 Hz

ARTFI denotes the input RTFI average energy spectrum, k = 1, 2, 3, … is the frequency index on the logarithmic scale, the second term in the right hand part of the equation denotes the moving average of ARTFI , and N_1 is the length of the window for calculating the moving average.

As shown in Figure 5, the red line represents the moving average of the RTFI energy spectrum. The relative energy spectrum RES(f_k) is a measure of the energy spectrum for the k^{th} frequency bin, relative to the energy spectrum over a frequency range near the k^{th} frequency bin.

If there is a peak in the relative energy spectrum at the k^{th} frequency index and the value RES(f_k) is larger than a threshold A_1, it is likely that there is a harmonic component at the frequency index k. The corresponding value RES(f_k) is assumed to be a measure of confidence in the existence of the harmonic component.

3*) Preliminary Estimations of Pitch Candidates:* In the third step, based on the harmonic grouping principle, the input RTFI average energy spectrum is first transformed into the pitch energy spectrum (PES) and the relative pitch energy spectrum (RPES) as follows:

$$PES(f_k) = \frac{1}{L}\sum_{i=1}^{L} ARTFI(i \cdot f_k), \text{ k=1,2,3,…} \tag{30}$$

$$RPES(f_k) = PES(f_k) - \frac{1}{N_2 + 1}\sum_{i=k-N_2/2}^{k+N_2/2} PES(f_i), \text{k=1,2,3,} \tag{31}$$

where N_2 is the length of the window for calculating the moving average, and L is a parameter that denotes how many low harmonic components are together considered as important evidence for determining the existence of a possible pitch. Similar techniques have been proposed for pitch estimations by some researchers. In [10], the authors propose a polyphonic pitch estimation approach by summing harmonic amplitudes. There are two main differences between the

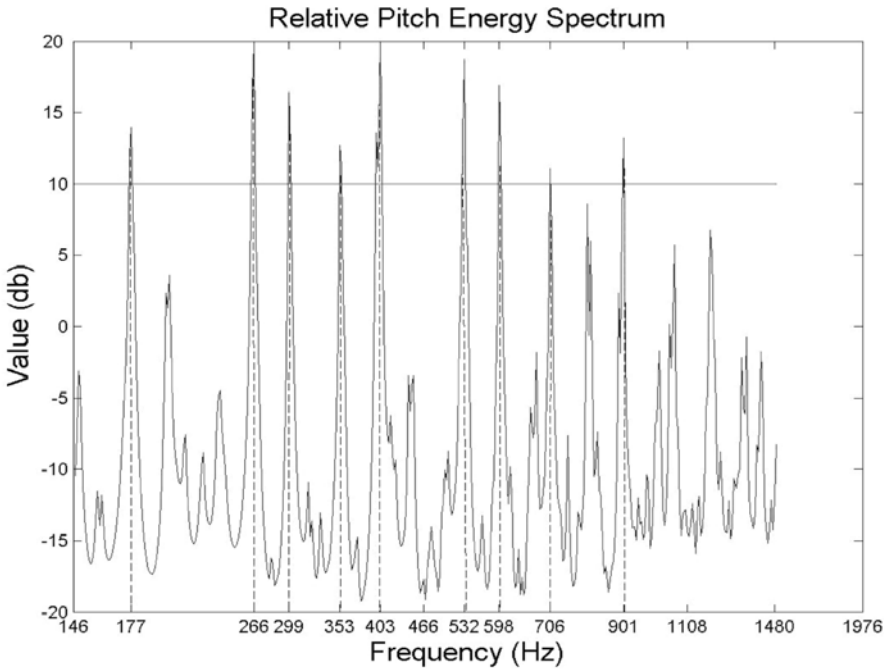

Figure 6. Relative Pitch Energy Spectrum of a violin example consisting of four concurrent notes with the fundamental frequencies 266Hz, 299Hz, 353Hz and 403Hz.

method described in this paper and the approach introduced in reference [10]. First, the reference approach is based on the STFT spectrum, whereas the proposed method employs an RTFI constant-Q spectrum. Secondly, the reference approach directly sums harmonic amplitudes and does not use a decibel scale, whereas the new method produces a pitch energy spectrum by summing the harmonic energies on a decibel scale. Our experiments demonstrate that directly summing the harmonic energies yields lower estimation performances.

The appropriate values of L and N_2 need to be set by performing experiments on a training database. In the following test experiments, L and N_2 have been respectively fixed to 4 and 50.

In practical implementations, instead of using equation (31), the pitch energy spectrum on a logarithmic scale can easily be approximated by the following expression (here L is less than 10):

$$PES(f_k) = \frac{1}{L}\sum_{i=1}^{L} ARTFI(f_{k+A[i]}) \tag{32}$$

$$A[10] = [0,120,190,240,279,310,337,360,380,399]$$

There are two assumptions made when determining a preliminary estimate of the possible pitches from the relative pitch energy spectrum. If there is a pitch with fundamental frequency $f_{k,}$ in the input signal, there should be a peak centred around the frequency f_k in the relative pitch energy spectrum, and the peak value should exceed a threshold A_2. Both assumptions are consistent with real music examples when a suitable threshold A_2 is selected.

Figure 6 illustrates the relative pitch energy spectrum of a violin example, which consists of four concurrent notes with fundamental frequencies of 266 Hz, 299 Hz, 353 Hz and 403 Hz respectively. As shown, there are 9 pitch candidates that can be preliminarily estimated when selecting the threshold A_2= 10dB. The fundamental frequencies of the 9 pitch candidates are 177 Hz, 266 Hz, 299 Hz, 353 Hz, 403 Hz, 532 Hz, 598 Hz, 796 Hz and 901 Hz. Such preliminary estimation includes 4 correct pitch candidates and 5 incorrect ones. The incorrect pitch estimations

usually share many harmonic components with the true pitches. In this example for instance, the false pitch of 177Hz is positioned at a frequency that is nearly half that of the true pitch of 353 Hz.

4*) Removal of Extra Pitches by Checking Harmonic Components*: By means of a large number of experiments it has been observed that the lowest harmonic components of the music notes are relatively strong and can be reliably extracted by applying the second step of the developed method. Only the low-pitch notes may have very faint first harmonic components that cannot be reliably extracted. Based on these observations, some assumptions concerning the extracted harmonic components can be made for determination of whether an extracted pitch is correct. For example, if there is a pitch with a fundamental frequency higher than 82 Hz, either the lowest three harmonic components or the lowest three odd harmonic components of this pitch should all be present in the extracted harmonic components. If there is a pitch with a fundamental frequency lower than 82 Hz, four of the lowest six harmonic components should be present in the extracted harmonic components.

In two typical cases, the extra estimated pitches can be removed based on the above assumptions. In the first case, the extra pitch estimation is caused by a noise peak in the preliminary pitch estimation. In the second case, the harmonic components of an extra estimated pitch are partly overlapped by the harmonic components of the true pitches. In such a case, the non-overlapped harmonic components become important clues to check the existence of the extra estimated pitch. If a polyphonic set of notes contains two concurrent music notes C5 and G5, for example, the fundamental frequency ratio of the two notes is nearly 2/3. Then, it is probable that there is an extra pitch estimation on the C4 note, because its even harmonics are overlapped by the odd harmonics of C5, and the C4 note's third, sixth, ninth, … harmonic components are nearly overlapped by the G5 note's odd harmonics. However, the C4's first, fifth, and seventh harmonic components are not overlapped, so the extra C4 estimation can be easily identified by checking the existence of the first harmonic component based on the above assumption.

5*) Determining the Existence of the Pitch Candidate by the Spectral Irregularity:* By means of the previous steps, the extra incorrect estimations centred around the pitches whose note intervals are 12, 19, or 24 semitones higher than the identified true pitches. In such a case, the fundamental frequencies of the extra estimated pitches are placed 2, 3 or 4 times the frequency of a true pitch, and the harmonic components of each extra pitch are completely overlapped by the true pitch. For example, consider when two of the estimated pitch candidates are the notes with fundamental frequencies F_0 and $3F_0$. Here the difficulty is to determine if the note with the fundamental frequency $3F_0$ is an incorrect extra estimation caused by the overlapped frequency components of the lower frequency music note. This is the most difficult case in the polyphonic pitch estimation problem. However, such a problem can be solved by investigating spectral irregularity.

The spectral value difference between two neighbouring harmonic components is small and random in most cases. But when a music note with the fundamental frequency F_0 is mixed with another note with the higher integer ratio fundamental frequency nF_0, then the corresponding spectral value of every n^{th} harmonic component will become clearly larger than the neighbouring harmonic components.

Figure 5 illustrates the RTFI energy spectrum of the first 30 harmonic components of two piano music samples. The top image presents the analysis results for a piano sample that contains only one music note with a fundamental frequency of 147 Hz. The bottom image shows the result of analysis for a piano sample that has two concurrent music notes with a fundamental frequency of 147 Hz and 440Hz (≈3*147Hz). It is clear that, in comparison to the top image, the 3rd, 6th, 9th,etc, harmonic components are reinforced and their spectral values are significantly larger than the neighbouring harmonic components.

If there are two estimated pitch candidates that have fundamental frequencies of F_0 and F'_0 ($F'_0 \approx nF_0$), and a frequency ratio that is approximately an integer n, then the proposed method employs the following two steps to determine if the higher pitch with the fundamental F'_0 occurs. First, the energy spectrum of the first $10n$ corresponding harmonic components with the fundamental frequency F_0 is calculated by an RTFI analysis with uniform resolution. The RTFI average energy spectrum of the harmonic components can be expressed as $ARTFI_H(k)$, k=1, 2, 3, …, (10n), where k denotes the harmonic component index.

The second step is composed of the following operations. The Spectral Irregularity (SI) is calculated using the expression:

$$SI(n) = \sum_{i=1}^{9} (ARTFI_H(i \cdot n) - (\frac{ARTFI_H(i \cdot n - 1) + ARTFI_H(i \cdot n + 1)}{2})) \quad (33)$$

According to our observations, if two of the estimated pitch candidates have the fundamental frequencies, F_0 and F'_0 for which ($F'_0 \approx nF_0$) and if the higher pitch does not occur, then SI(n) is usually small. On the other hand, if the higher pitch does occur, then the overlapped harmonic components are often strengthened so that SI(n) results in a larger value. When SI(n) is smaller than a given threshold, the overlapped higher pitch candidate is removed. The threshold is determined by experiments on a training database. In practical examples, most incorrect extra estimates caused by the overlapping of harmonic components are placed at a low integer multiple of the frequency of the true pitch. Consequently, the new method proposed in this paper only consider cases for which the fundamental frequency ratio of two pitch candidates is equal to 2, 3 or 4.

4.4.3. Novelty of the Proposed Method

In this subsection, the novelty and promising features of the proposed method is outlined. In the time-frequency processing part, the Fast RTFI constant-Q time-frequency analysis is first employed for polyphonic pitch tracking. As explained in Section 4.3, it is much more computationally efficient than other implementations.

In the post-process phase, the developed method first estimates pitch candidates by peak-picking from the relative pitch energy spectrum. Since the sounds with integer fundamental frequency ratio can produce very similar peak patterns in a pitch energy spectrum, usually an extra incorrect estimation has an integer ratio to the fundamental frequencies of an identified pitch. This problem mainly arises from the coinciding frequency partials between Western polyphonic music notes.

The state-of-the-art method solves the problem by employing iterative estimation and cancelation schema [10]. The basic idea is to first find a predominant pitch, and estimate the spectrum of the predominant pitch. Then the estimated spectrum is cancelled from the mixture and produces residual signals before the next estimation. The estimation and cancellation is repeated iteratively on the residual signal. It may also involve the process of estimating the polyphonic number of the analyzed sound.

So as to solve the problem of coinciding frequency partials, the basic idea of the new proposed method is completely different from the state-of-art approach introduced above. The proposed method provides a much simpler solution to the problem and does not require to implement an iterative procedure or to estimate the polyphonic number. In the new method, the preliminary estimation finds all possible pitch candidates. Then some pitch candidates are removed if their harmonic components are not enough represented in the energy spectrum. Finally, if fundamental frequencies between any two pitch candidates have an integer ratio, the spectral irregularity is calculated to remove the pitch candidate, which is considered to be an error estimation caused by coinciding frequency partials from a lower pitch.

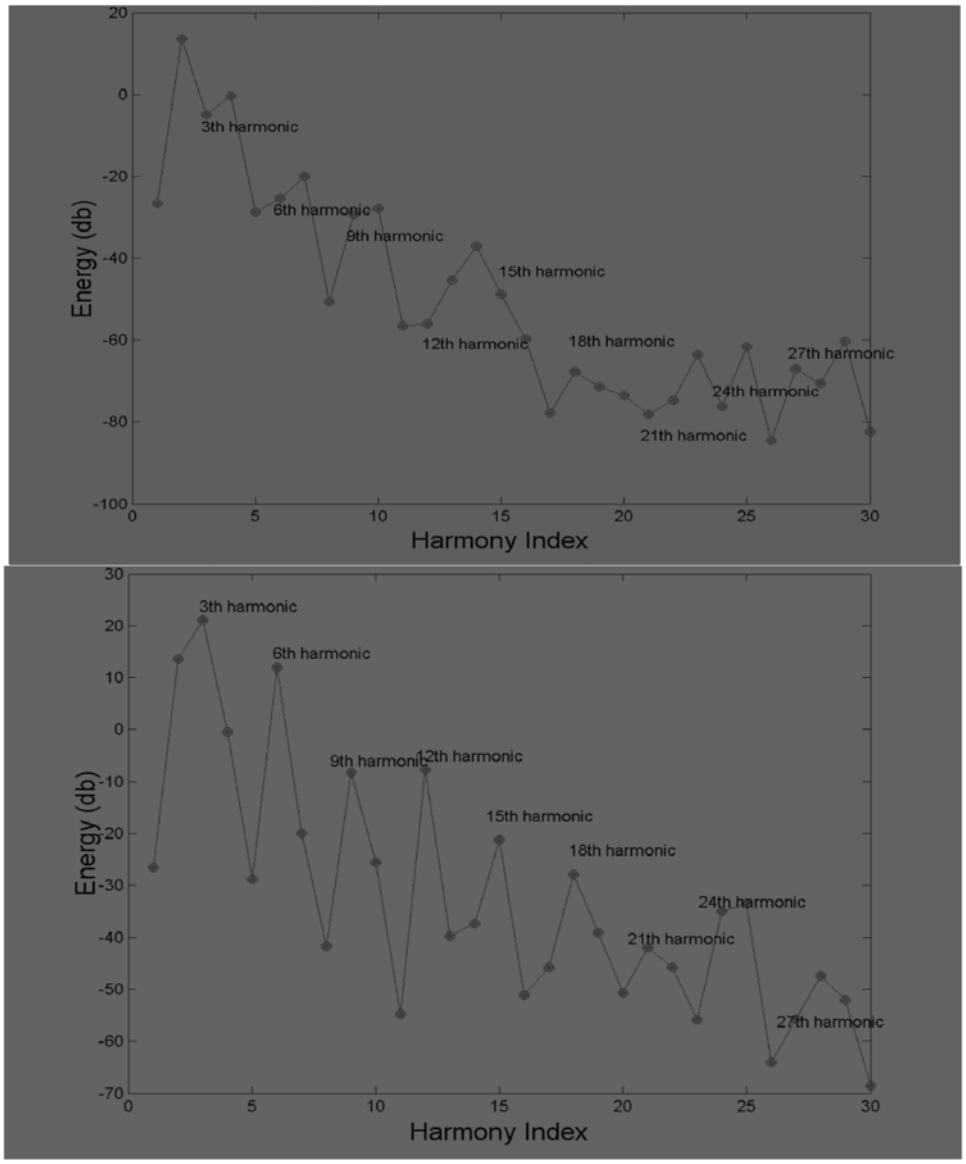

Figure 7 Harmonic component energy spectrum of a piano sample including a single note with fundamental frequency at 147 Hz (top), and a piano sample including two concurrent notes with fundamental frequencies at 147 Hz and 440 Hz (bottom).

By employing these new techniques, the proposed method is more computationally efficient, but presenting comparable performance with the other state-of-art methods.

4.5. Experiments and Results

4.5.1. Performance Evaluation Criteria

Three criteria were used to evaluate the performance of the polyphonic pitch estimation methods; “Precision”, “Recall”, and “F-measure”. Given a reference fundamental frequency, if there is an estimation that is equal to or presents an error of no more than 3% deviation from the reference fundamental frequency, it is considered to be a correct detection. Otherwise, it is considered as a false negative (FN). Any estimation that deviates by more than 3% from all

reference fundamental frequencies is considered to be a false positive (FP). Precision, Recall, and F-measure can be defined according to the following expressions:

$$P = N_{CD} /(N_{CD} + N_{FP}) \tag{34}$$

$$R = N_{CD} /(N_{CD} + N_{FN}) \tag{35}$$

$$F - measure = 2PR/(P + R) \tag{36}$$

where N_{CD}, N_{FP}, and N_{FN} denote the total number of correct detections, false positives and false negatives, and P and R denote the values of precision and recall, respectively. In addition, the Overall Accuracy is also used for the performance comparison with other state-of-art methods.

4.5.2. Setting the Method Parameters

The real performance of an estimation method may be overestimated when parameters have been optimally selected to fit the test data. So as to prevent such occurrence, separate training and testing datasets have been constructed. The training dataset was used to tune the parameters of the described method. The thresholds A_1, A_2 and the thresholds of spectral irregularity are the most sensitive parameters. The different parameter values were selected by a heuristic method. Values

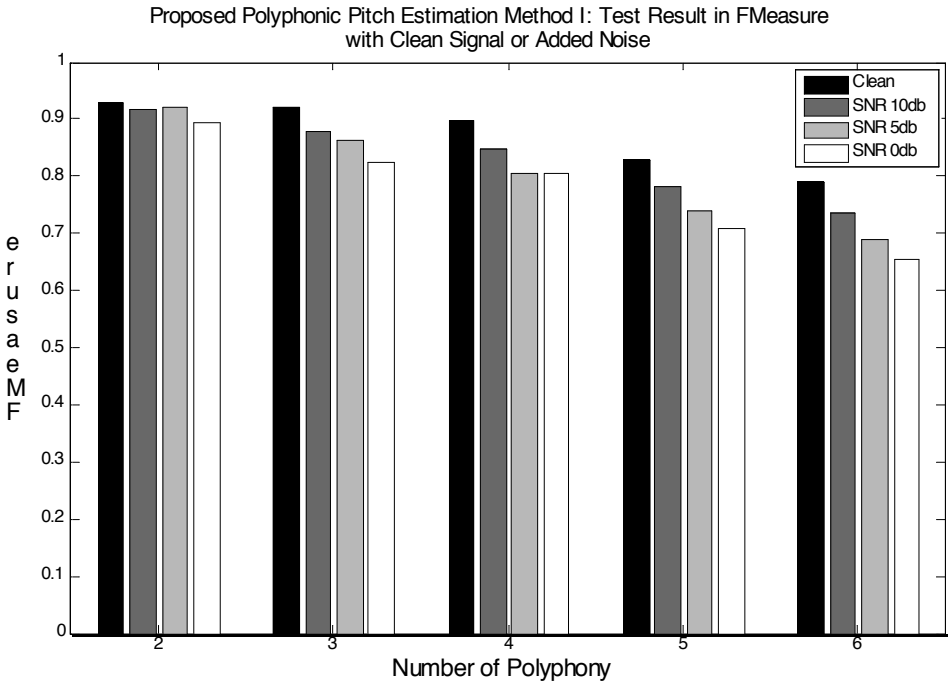

Figure 8 F-Measure of test results of the proposed method with a clean signal or various levels of added noise.

that yielded the best average F-Measure on the training dataset were selected, and parameters were fixed when the method was evaluated on the test dataset.

It is quite difficult to record a large number of polyphonic samples from different musical instruments and label their polyphony content. A preferred method is to produce the polyphonic samples by mixing real recorded monophonic samples of different music instruments.

In these experiments, two different monophonic sample sets were used to create the training and test dataset. The monophonic sample set I consisted of a total of 755 monophonic samples from 19 different instruments, such as piano, guitar, winds, strings, and brass, etc. Every monophonic

Table 2:F-Measures of Proposed Method with/without Applying Relative Spectrums		
Polyphony Number	Using Relative Spectrum	Not Using Relative Spectrum
2	93%	89%
3	92%	87%
4	90%	85%
5	83%	81%
6	78%	78%

s
ample was normalized into equal mean-square level and fades within a one second duration time. The high number of polyphonic samples was generated by randomly mixing these different monophonic samples.

To obtain fairer evaluation results of practical cases, the monophonic sample set II was used to generate the test dataset in the same way in which the training dataset had been produced from the monophonic sample set I. Compared to set I, the monophonic samples in Set II, for the same type of instrumentation as samples in Set I, were played by different performers and instruments from different instrument manufacturers. Set II included 23 different instrument types, a total of 690 monophonic samples in the five octave pitch range of 48 Hz to 1500 Hz. All the monophonic samples in Set I and Set II were selected from the RWC instrument sound database. The test dataset was then used for performance evaluation of the described method.

4.5.3. Performance and Robustness

The method was tested on the test dataset and achieved F-measures of 93%, 92%, 90%, 83%, and 78% respectively on polyphonic mixtures ranging from two to six simultaneous sounds. In order to test the robustness, pink noise was added into the polyphonic mixtures with different Signal-to-Noise ratios. The pink noise was generated in the frequency range of 50 Hz to 10K Hz. The Signal-to-Noise refers to the ratio between the clean input signal power and the added pink noise power.

Figure 8 shows the F-measure of the new method with different levels of added pink noise, where a value of 1 for the F-measure indicates optimal performance. In general, the method is robust, even in cases of severe noise levels. The tested samples were classified into five different sample subsets according to the polyphony number of the mixed polyphonic samples. For example, in Figure 6, the F-measure corresponding to the polyphony number 2 denotes the F-measure value estimated on the sample subset, in which every polyphonic sample consists of a two-note mixture. In this test experiment, 100 test examples were randomly selected from the test dataset for every polyphony sample subset.

4.5.4. Comparison Experiments with/without Applying Relative Spectrums

In the described method, the relative spectrums (relative energy spectrum and relative pitch energy spectrum) have been used. A comparison experiment has been made to evaluate how the application of relative spectrums improves the method's performance. As in the previous experiment, 100 test examples were randomly selected from the test dataset for every polyphony sample subset. The test results of the method with or without applying the relative spectrum are

reported in Table 2. The results demonstrate that the application of the relative spectrum improves the method's performance.

4.5.5. Trade-off between Recall and Precision

Figure 7 shows the estimation performance (F-measure, Recall, Precision) of this method with two different parameter sets. Comparing the left image (with a smaller parameter value) to the right image (with the larger parameter value) in Figure 7, the corresponding Precision shown in the right image increases at the price of a lower Recall. In general, the total estimation performance, F-measure, is reduced by increasing the polyphony number of the estimated polyphonic sample. In Figure 7, it can also be observed that the estimation of Recall is greatly reduced in the cases where the polyphony number is increased. On the other hand, the Precision is gradually changed.

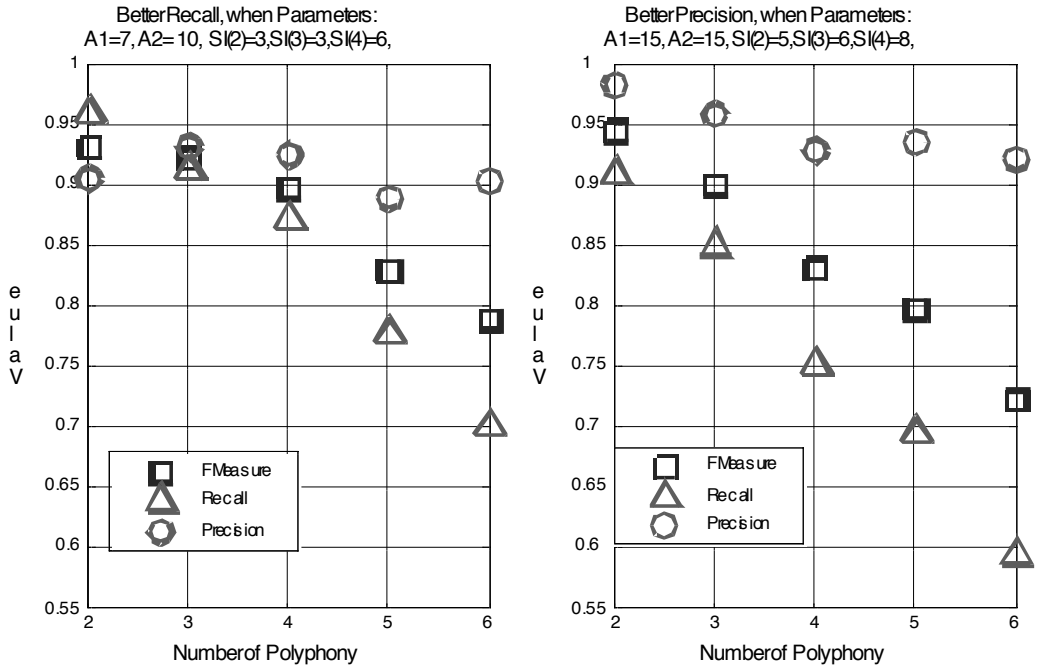

Figure 9 F-Measure, Recall and Precision results for the proposed method with different parameters.

4.5.6. MIREX 2007 Evaluation

In order to compare our technique with other state-of-art approaches, the new method was submitted to the multiple fundamental frequency frame level estimation task of MIREX 2007. In this evaluation task, there were 28 test files, each of which had a 30-second duration. These files consisted of 20 real recordings, 8 synthesized from RWC samples. The summary results of the first 10 methods in the rank are reported in Table 3. In the evaluation, our method (labelled as team 'ZR') was ranked third in the 16 submitted approaches. However the difference of results between our method and the best method (team 'RK') was really minor, whereas our method was approximately 13 times faster than the best method (team 'RK'). The algorithm has been implemented as Matlab M-files and MEX-files. The execution time on a 2 GHz Pentium processor is about one third of the time duration of a monaural audio recording.

Table 3:Results of Multiple Fundamental Frequency Frame Level Estimation Task of MIREX 2007		
Team ID	Accuracy	Running Time (sec)
ZR	58.2%	271
RK	60.5%	3540
CY	58.9%	132300
PI1	58.0%	364
EV2	54.3%	2233
CC1	51.0%	2513
SR	48.4%	41160
EV1	46.6%	2366

4.6. Conclusion about the multiple pitch estimation method

A computationally efficient and robust method has been developed to estimate polyphonic pitches of real polyphonic music. Compared to the state-of-art approach, the proposed method is much faster, but presents comparable performance. The method achieves F-measures of 93%, 92%, 90%, 83%, and 78% respectively on polyphonic mixtures ranging from two to six simultaneous sounds. Approximately 48% of all errors are octave errors, and about 15% of all errors are due to confusion between notes with the fundamental frequency ratio of 1/3.

The method is based on a preliminary estimate of pitch candidates by simple peak-picking in the relative pitch energy spectrum. Then, some pitch candidates are removed if their existence contradicts some general assumptions concerning the spectral harmonic characteristics of notes on the western musical scale. Possible remaining ambiguities (such as an integer ratio between fundamental frequencies) can efficiently be solved by investigating the spectral irregularity. Other than the assumption of spectral characteristics generally associated with western music, no other assumptions are made concerning the structure of the music or the specific instrument.

In future extension of this work, a priori knowledge of the content of the analyzed music may be used to further improve the performance of this technique. For example, if it is known that the analyzed music is a solo piano music, then spectral characteristics and thresholds could be specifically set using the same class of solo piano music. Moreover, the method can also be improved by using temporal features not exploited yet. The harmonic components from the same instrument sound source often present similar temporal features, such as a common onset time, amplitude modulation and frequency modulation. The harmonic relative frequency components with similar temporal features should have a higher probability of representing the same note than the harmonic relative frequency components with different temporal features.

5. Prototype Transcription System

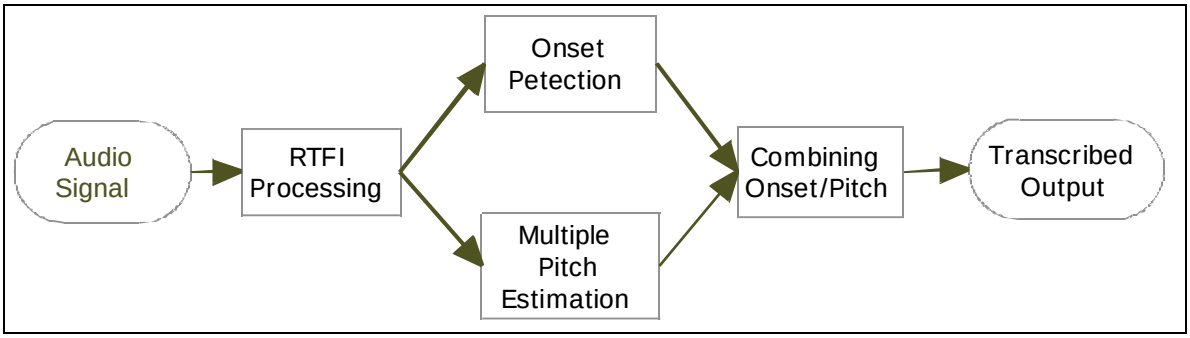

Figure 10 Prototype Transcription System

An automatic music transcription prototype system has been constructed with the combination of the proposed music onset detection algorithms and polyphonic pitch estimation methods. The goal of the transcription system is to detect the music notes occurring, their onset times and note duration times. The sampling rate of input music signal is 44100 Hz. The onset detection algorithms are first used to separate the input real music signal into different segments according to the detected note onsets, and then pitches in each segment are estimated by the developed multiple pitch estimation method. Finally, every estimated pitch in a certain segment must be checked if the pitch begins from a current segment or from the previous segments. For a certain segment N, if a pitch A with fundamental frequency f is estimated; then if the estimated pitches in the previous segment N-1 do not contain the pitch A, the transcription system will consider that this pitch A is a new occurring pitch in the segment N. In another case when the estimated pitches in the previous segment N-1 also contain the pitch A, then this pitch A is considered to be a new occurring pitch only on the condition that the corresponding energy spectrum of the pitch A's first or second harmonic component has been obviously increased at the starting moment of the segment N. The note duration time is directly determined by how long the pitch continues to exit. Figure 10 shows the overview of the automatic music transcription system.

In EASAIER, the transcription system is used to visualize the image. An EASAIER user can view the piano-roll image of a music signal by a music transcription function. Figure 11 shows a piano-roll image of a piano example.

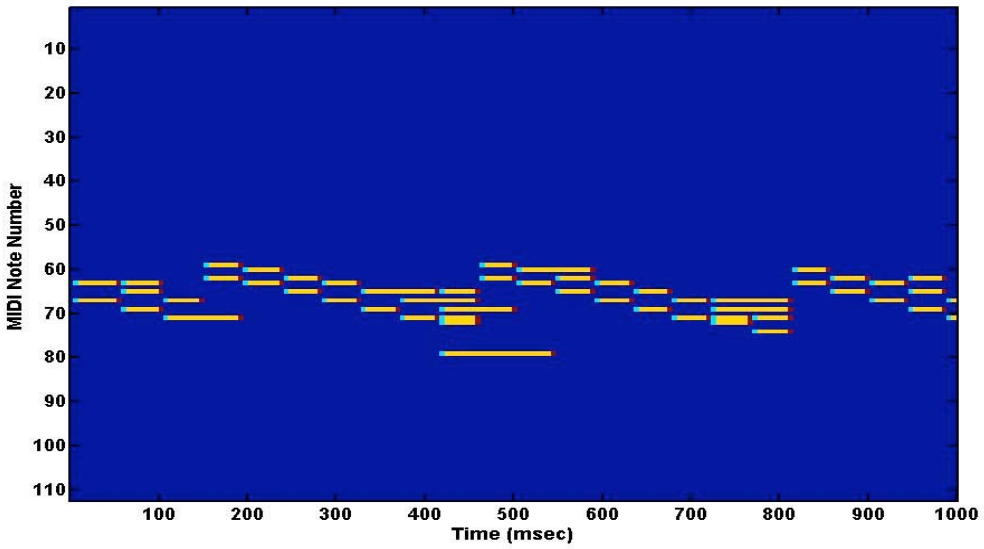

Figure 11 Piano-roll visualization of a piano example

6. References

[1] J.P.Bello, L.Daudet, S.Abadia, C.Duxbury, M.Davies and M.B.Sandler, "A tutorial on onset detection in music signals," in *IEEE Trans. Speech and Audio Signal Processing*, vol. 13, pp. 1035–1047, Sept.2005.

[2] A.Klapuri, "Sound onset detection by applying psychoacoustic knowledge," in *Proc. IEEE International Conf. Acoustics, Speech, and Signal Processing (ICASSP-99)*, pp. 3089–3092, Mar. 1999.

[3] J. P. Bello and M. Sandler, "Phase-based note onset detection for music signals," in *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing (ICASSP-03)*, Hong Kong, 2003, pp.49-52.

[4] J. P. Bello, C. Duxbury, M. Davies, and M. Sandler, "On the use of phase and energy for musical onset detection in the complex domain," in *IEEE Signal Processing Letter*, vol. 11, no. 6, pp. 553-556, Jun. 2004.

[5] N.Collins, "Using a pitch detector as an onset detector," in *Proc. International Conf. On Music Information Retrieval*, London, Sep. 1999.

[6] R.Zhou, Feature Extraction of Musical Content for Automatic Music Transcription, Ph.D. dissertation, Swiss Federal Institute of Technology, Lausanne, Oct, 2006. [Online] Available: http://library.epfl.ch/en/theses/?nr=3638.

[7] R.Zhou and M.Mattavelli, "A new time-frequency representation for music signal analysis" in *Proc. International Conf. on Information Sciences, Signal Processing and its Applications*, Sharijah, United Arab Emirates, Feb. 2007.

[8] R.Zhou and J.D.Reiss, "Music onset detection combining energy-based and pitch-based approaches," in MIREX 2007 audio onset detection contest: http://www.music-ir.org/mirex2007/abs/OD_zhou.pdf

[9] W.L, Y.Shiu and C.J.Kuo, "*Musical onset detection with linear prediction and joint feature* " MIREX 2007 audio onset detection contest: http://www.music-ir.org/mirex2007/abs/OD_lee.pdf

[10] A.Klaupri, " Multiple fundamental frequency estimation by summing harmonic amplitudes," In *Proc. International Conference on Music Information Retrieval (ISMIR-06)*, Victoria, Canada, pp. 216-221, Oct, 2006.