Automatic MUSIC TRANSCRIPTION SUPPORTING DIFFERENT INSTRUMENTS

Automatic Music Transcription

Supporting Different Instruments

I. Bruno, P. Nesi

Dep. of Systems and Informatics University of Florence

Via S. Marta, 3 50139 Florence, Italy

+39 055 4796425, +39 055 4796365, +39 055 4796523

ivanb@dsi.unifi.it, nesi@dsi.unifi.it

(version 6.0, March 2005)

ABSTRACT

Automatic music recognition from an audio performance is a key problem and a challenge for coding music in western music notation in the digital world. This problem has been addressed in several manners and obtaining suitable results when a single specific instrument and monophonic music are processed. The development of a system for the automatic music transcription that is able to cope with different music instruments is the aim of this paper. Experimental results have shown that for monophonic pieces the recognition is quite viable and the process can be parameterised to realize a music independent recognition tool and process.

Keywords: **beat tracking, pitch recognition, audio processing, automatic music recognition.**

1.   Introduction

The automated transcription of music is the process that allows the extraction of musical information by a recorded audio signal and representing it by means of music notation in terms of notes, pitches, duration, attack and decay time. This operation is not an easy task, since it requires an analysis considering both physical and psycho-acoustical points of view. The transcription process has to consider the relationships between the sound as physic phenomena and the sound perception of the human ear. The human ear can perceive musical tones even in presence of noise, simultaneous tones and expressive tonal deviations like the vibrato. Various auditory models ([1], [2]) which try to physiologically reproduce the human auditory system have been developed in the past. They were introduced to cope with the speech recognition process ([15], [21]), but their use was extended also in audio signal processing. Even if they must be considered as only an approximation of physical reality, they appear to be a suitable system for identifying those aspects of the audio signal that are relevant for automatic audio analysis and recognition. Furthermore, with these models of auditory processing, perceptual properties can be re-discovered starting not from the sound pressure wave, but from a more internal representation which is intended to represent the true information available at the acoustic nerve of the human auditory system. In this paper, an algorithm and a model for automatic transcription of music are described. They are inserted in the study and development of a system capable of coping with different musical instruments. In order to satisfy this requirement, several parameters have been introduced to describe the musical instrument features and tune the system in order to manage it. The solution proposed is based on the auditory model of M. Slaney and Roy Patterson ([12]) and neural networks banks for the recognition of sound features for polyphonic music. The paper is focussed mainly on the recognition of monophonic music. The paper is organised as follows. In Section 2, an overview of automatic music transcriptions methods is presented. In Section 3, the main architecture of the proposed system is shown. In Section 4, the parameters for setting up the system and single modules are described. In Section 5, experimental results are shown and discussed. Results were obtained converting some recorded audio pieces related to monophonic music score and performed by different musical instruments. Conclusions are reported in Section 6.

2. Overview

In the last years, several automated transcribers for music and audio processing algorithms were developed and focussed to the identification of the attack (*onset detection algorithm*) and pitch of notes (*pitch detection*). These algorithms [16] are classified into two main categories: (i) time and (ii) frequency domain methods.

The time domain methods are used in a real-time context; that is, when the main requirement is the velocity of note recognition. They provide good results when they are employed in the transcription of monophonic music (when notes are played one by one). In this category, the pitch detection algorithms are based on the periodicity theory of the pitch perception: evaluating the periodicity with which the characteristic waveform of the signal is repeated allows pitch recognition of the acoustic signal. In the works of Moorer [3], the periodicity detection is conducted by means of the *zero-crossing* method (number of times in which the signal crosses a zero-threshold reference in a time unit). In [20] and [30], the pitch detection algorithm is performed by using the *auto-correlation* function ([4]): this function is a likelihood index between an audio signal and its translated version. The periodicity of the audio signal implies the periodicity of the corresponding auto-correlation function and vice versa. In this way, it is possible to estimate the period of the audio signal by evaluating the distance between the two consecutive maxima of the autocorrelation function. The *Envelope Periodicity* method is based on the pitch periodicity that can be derived from the periodicity of the envelope by setting marks where the signal exceeds the envelope or in correspondence of zero crossing. The original method, developed in [33], has been extended in [34] by considering both negative and positive amplitudes of the envelope. A *Time Parallel Processing* approach was defined for the fundamental frequency detection [36]: the signal is processed to create a number of impulse trains that retain the periodicity of the original signal and discard features that are irrelevant to the pitch detection method. Simple estimators are used to detect the period of these impulse trains. All the estimates are logically combined to infer the period of the signal waveform.

The frequency domain methods provide better results than those based on time domain, but they are not suggested to cope with real time requirements. They allow extending the recognition to the polyphonic music (that is, two or more notes are played at the same time). They are based on theoretical and experimental results ([5], [6], [7]), which is proof that the human brain decomposes the audio signal in spectral components and tries to detect a common reference to all harmonics inside the signal in order to establish the pitch. For instance, if a signal is constituted by the 600 Hz, 800 Hz and 1000 Hz harmonics, the brain is oriented to fix the common reference at 200 Hz, in other words it selects the missed fundamental harmonic, since all the harmonics are multiple of 200 Hz. Many pitch detection algorithms were designed on this theory ([8]). The *Spectrum autocorrelation* is derived from the observation that a periodic non sinusoidal signal has a periodic magnitude spectrum, whose period is the fundamental frequency and it could be extracted by using the autocorrelation function ([18]). In the *Harmonic Matching Methods*, the aim is to extract the period from a set of spectral maximum of the magnitude spectrum of signal. The detected peaks in the spectrum are compared to the predicted harmonics for each of possible candidate note frequencies by means of a particular fitness measure ([19]). Another frequency method is based on the *Wavelet transform*. This transform allows a multi-resolution and multi-scale analysis that has been shown to be well suited for music processing because of its similarity to human ear behaviour. In contrast to the short-time Fourier transform, which uses a single analysis window, the wavelet transform uses short window at high frequencies and long window for low frequencies. The main approach is to filter the signal using wavelet with derivative properties. The output signals will have maxima where zero crossing happens on the input signal. The distance between two consecutive maxima is used to estimate the fundamental frequency [31]. The constant Q frequency analysis was proposed in [32] to define an algorithm for periodicity analysis that calculates independent fundamental frequencies estimated at separate frequency bands (*Bandwise Envelope Periodicity*).

Recently, solutions based on statistical and neural network approaches have been proposed. In [22] and [35], a Bayesian model describes each component ‘note’ at a given time in terms of a fundamental frequency, partials (‘harmonics’), and amplitude. This basic model is modified for greater realism to include non-white residuals, time-varying amplitudes and partials ‘detuned’ from the natural linear relationship. A pitch detector based on *hidden Markov model* (HMM) was proposed in [23] and [24]. Marolt proposed a system focussed to the pitch recognition of notes played polyphonically by a piano by means of 76 adaptive time-delay neural networks (TDNN) simulating the piano notes [9] and an transcription system based on a connectionist architecture that employs networks of adaptive oscillators for partial tracking and feed forward neural networks for associating partial groups with notes [25]. In [26], an intelligent neural networks (INN) was trained to recognise the pitch both guitar and humming. Finally, Pertusa in [17] deals with the monotimbrical polyphonic version using a TDNN fed only with the spectrogram of notes.

3. General Architecture

The general architecture of the proposed music transcriber is modular (see Figure. 1). It is based on the Patterson-Meddis auditory model and includes a partial tracking module and a neural network bank. Since, the main goal has been to realise a generic transcriber in capable of managing different musical instruments with both monophonic and polyphonic, three different working modalities were developed:

- **Monophonic Transcription mode**: it allows the transcription of monophonic music performed by a generic music instrument.
- **Polyphonic Transcription mode**: it allows the transcription polyphonic music performed by a generic musical instrument. If the neural network bank was previously trained on notes simultaneously played by the used instrument, it is capable of recognising chords of notes.
- **Training mode**: it allows the automatic creation of features and patterns set to use in training new neural networks with new music instruments..

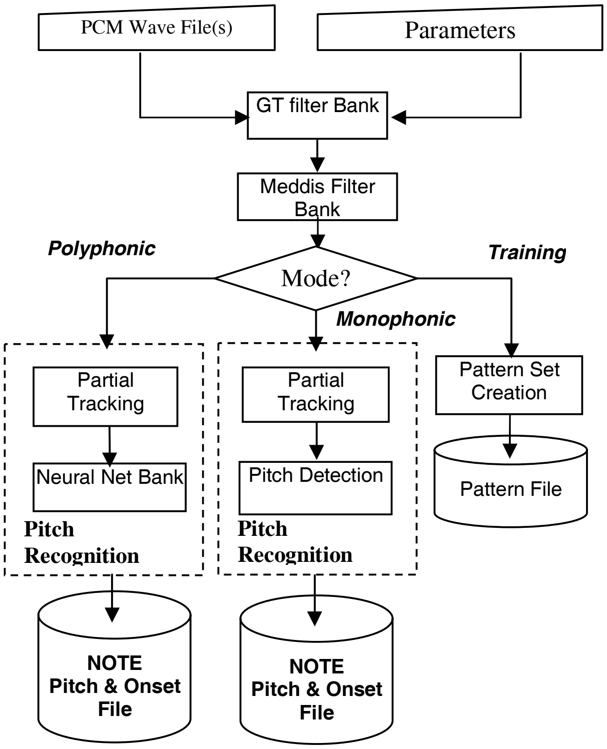


Fig.1 - General Architecture of the music transcriber.

The output provided by the monophonic and polyphonic mode is a list of notes. For each of them, the note-on and note-off time instant, the pitch and the volume are provided. This description is translated into a MIDI file.

The other modules presented in the figure are:

- **Gammatone (GT) Filter Banks** – A set of filters implement the Patterson auditory model, whose aim is to model the movement of the basilar membrane of the internal human ear. It is made of filters band-pass bank, called gammatone ([12]).
- **Meddis Filter Bank** – A set of filters to implement the Meddis model ([10]). They simulate the behaviour of hair cells of the human ear. This module converts the output of the GT filter banks into a probabilistic representation of firing activity in the auditory nerve.
- **Partial Tracking** – The aim of this module is the detection of the entire harmonic components of each note to be recognised.
- **Pitch Detection** – The module performs the harmonic components analysis in order to establish the pitch of note.
- **Neural Network Bank** - Used when it needs to recognise pitches both of singles notes and of chords of notes. Each neural network has one output that is activated every time a note or a chord is recognised.

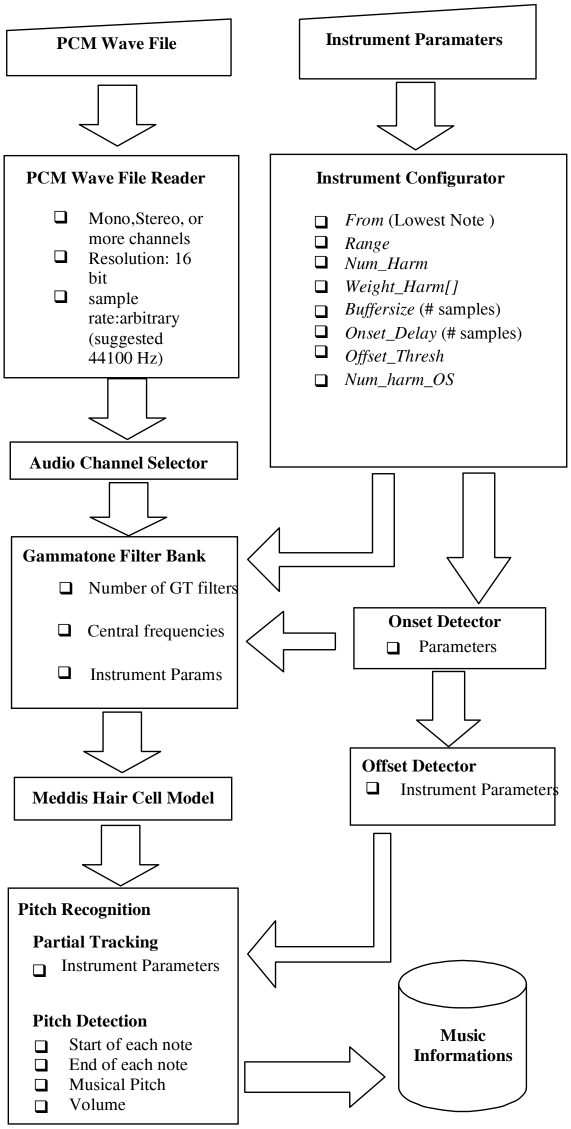


Fig. 2 - Architecture of the monophonic transcriber.

- **Pattern Set Creation** – It performs the automatic patterns generation that are used in the neural networks training.

4. Monophonic Transcription Mode

This mode executes the transcription of a monophonic music performed by a generic musical instrument. The architecture for the monophonic transcription is depicted in the Figure 2. The input to the system is constituted by:

- An audio file in ".wav" format, with arbitrary sampling frequency (the 44100 Hz sampling frequency provides better results), 16-bit resolution and any number of audio channels (the audio could be mono, stereo or with more than two channels).
- A list of parameters related to the music instrument. This information is used to set up the pitch recognition process. Different values have been defined to cope with the guitar, the piano and the violin.

The Audio Channel Selector module extracts samples related to a specific channel from the audio file. Alternatively, when the audio file has more than one channel, this module creates a new set of audio samples, where each sample is calculated as the average value of samples of each channel related to the same moment of time.

The Gammatone Filter Bank is a band-pass filters bank, the dimension of the filters bank is strictly connected to the features of selected musical instrument involved in the transcription. These features are collected and stored in the Instrument Configuration module. The number of filters (auditory channels) is related to the range (extension) of notes that the instrument can play. Each filter is centred on the fundamental frequency of a specific note. The first filter is normally centred on the lower note and the others are centred on increasing frequencies. The distance between frequencies is a semitone and follows the rule of the equable temperament scale. The Gammatone Filter Bank receives samples corresponding to a note attack coming from the Onset Detector module ([11], [13]). The distances between two consecutive samples define the portion of signal to be filtered.

The Meddis Hair Cell Model performs a second filtering on signals and the ith Meddis channel elaborates the signal centred on the f_i frequency. The Partial Tracking and the Pitch Detection algorithm perform the Pitch Recognition. For each note, the pitch and the volume are provided. The note recognition is completed with the note on instant and its duration. These parts of data are provided respectively by the Onset Detector and the Offset Detector. Finally, the detected values and features are listed into a text file, which could be easily translated into MIDI code.

4.1 Instrument Configuration

The audio signal-filtering modules and the Note Recognition Module are strictly linked to the features of the music instrument involved in the audio processing. A music instrument is characterised by: the range of notes it can play, the timbre and the wave envelope. To define the range of the specific instrument, the FROM and RANGE parameters were introduced. They allow fixing respectively the pitch of the lower note in terms of standard MIDI code and the number of notes that the specific musical instrument can play.

Each family of instruments has its own wave envelope that is described by four main phases (see Figure 3): attack transient (*A*), decay transient (*D*), sustain state or steady-state (*S*) and release transient (*R*). In the attack and release transient the sound is rich of enharmonic frequencies, whereas in the sustain state, the sound maintains the

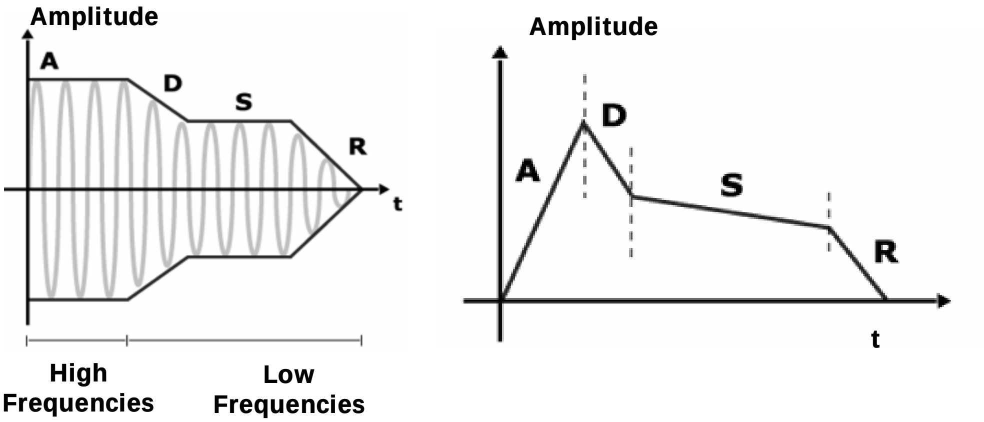


Fig. 3 - Sound envelope: attack (A), decay (D), sustain (S) and release (R).

periodicity feature. The wind instruments, for instance, have short attack-decay phases, whereas for strings instruments they are long with several harmonics and enharmonic frequencies. For all instruments, the sustain phase is the part of the envelope that allows detecting the pitch of a note. Thus, ONSET_DELAY and BUFFERSIZE parameters characterise the attack-decay transient. In details, the former indicates the number of samples involved in the transient and belonging to the wave envelope; the latter defines the consecutive number of samples involved in the sustain phase and not yet involved in the decay phase of the envelope. These two parameters are very important during the pitch detection. In fact, the pitch of each note has to be calculated in the interval where the amplitude of audio signal remains unaltered in the sustain phase. In this interval, the audio signal shows its periodicity and this periodicity is exactly used by the human ear to correctly distinguish the note pitch.

The WEIGHT_HARM array and the NUM_HARM allow the consideration of spectral components or harmonic content. More in details, the WEIGHT_HARM is an array of NUM_HARM weights that describes the harmonic content of the signal. NUM_HARM corresponds to the number of first harmonics needed to characterise the note and used in the pitch detection phase. Finally, the OFFSET_THRESH and the NUM_HARM_OS consider the particular behaviour of the envelope during the decay phase of the audio signal. They are used to increase the reliability of the offset detection algorithm. Table 1 shows the list of parameters respectively for the piano, the guitar and the violin relatively to an audio performance sampled at 44100 Hz.

Parameters	Piano	Guitar	Violin
FROM	22	50	55
RANGE	88	47	54
NUM_HARM	3	3	4
WEIGHT_HARM[]	1.0,0.8,0.4	1.0, 0.8,0.4	0.8,1.0,0.6,0.4
BUFFERSIZE	4000	4000	4000
ONSET_DELAY	3000	3000	3000
OFFSET_THRESH	0.08	0.25	0.15
NUM_HARM_OS	3	3	4

Table 1 – Parameters for Piano, Guitar and Violin.

4.2 Onsets Detection

The word onset designates a relevant sound within the music part. Onsets are punctual temporal events that correspond, for example, with the start of a note or a sudden increase of the sound volume. These onsets are expected to emphasise the important moments of a melody and, for some of them, the music's beats. The selected basic method has been derived from [11] and [13]. It consists in a time-domain analysis of the signal with the following steps:

- Signal smoothing to produce an amplitude envelope (envelope's determination).
- Using a peak-picking algorithm to find the local maxima (peaks' searching), local peaks are rejected if there is a greater peak within a given time (fixed as a parameter) or if it is below a given threshold (limit value).

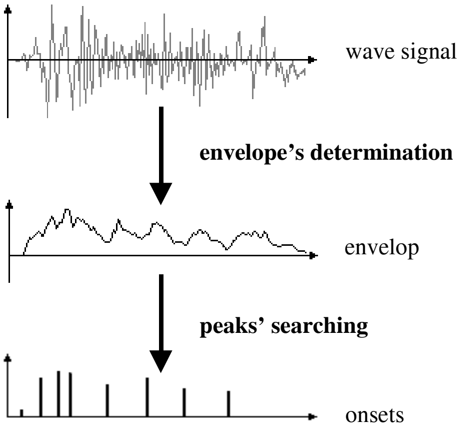


Fig. 4 – Onset Detection steps.

Figure 4 shows these steps of the process. Each point of the envelope function is calculated as the average of the absolute values of the signal within a time window centered on the point. Let be *A* the vector of the *n* samples of the audio signal and *E* the vector containing the *m* values of the envelope functional, *m* must be lower than *n* and the ratio *m/n* defines the translation of the time window.

$$\text{Audio signal} : A = \begin{pmatrix} a_0 \\ a_1 \\ \vdots \\ \vdots \\ a_{n-1} \end{pmatrix} \quad \text{Envelope} : E = \begin{pmatrix} e_0 \\ e_1 \\ \vdots \\ \vdots \\ e_{m-1} \end{pmatrix} \qquad slide = \frac{m}{n}$$

The size of the time window was set to *2*slide*. As shown in Figure 5, each value calculated for the envelope corresponds to a position of the time window on the audio signal.

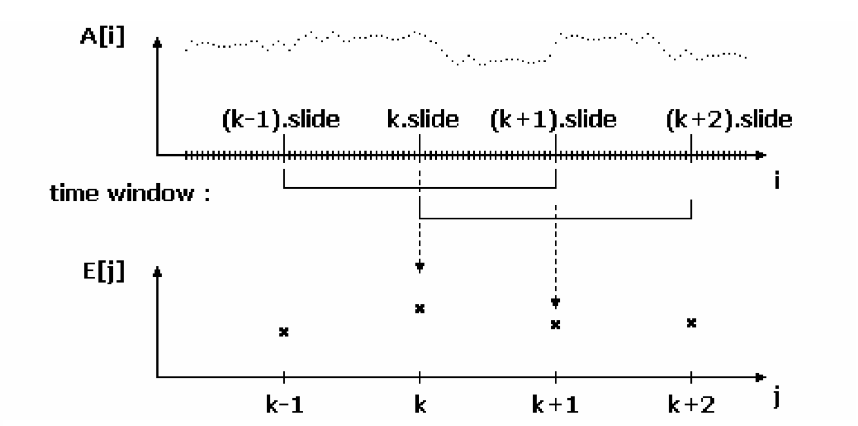


Fig. 5 - Envelope calculation.

For *k∈[1;m-2]*, *E[k]* is given by:

$$E[k] = \text{average}((k-1)slide, (k+1)slide) \qquad [3.1]$$

where *average(i_1,i_2)* corresponds to the average of the absolute values in the time window starting at sample i_1 and ending at sample i_2:

$$\text{average}(i_1, i_2) = \frac{\sum_{i=i_1}^{i_2-1} \left|A[i] - A[i+1]\right|}{i_2 - i_1} \qquad [3.2]$$

If S_k is defined as the following sum (for *k∈[0;m-2]*):

$$S_k = \sum_{i=k*slide}^{(k+1)*slide-1} \left|a_i - a_{i+1}\right| = \sum_{i=0}^{slide-1} \left|a_{k*slide+i} - a_{k*slide+i+1}\right| \qquad [3.3]$$

Then, [3.1] can be written as:

$$E[k] = \begin{cases} \dfrac{S_0}{slide} & , k = 0 \\ \dfrac{S_{k\text{-}1} + S_k}{2 * slide} & , k \in [1; m-2] \\ \dfrac{S_{m-2}}{slide} & , k = m-1 \end{cases} \qquad [3.4]$$

For the definition of the envelope, *slide* is set to 441. This means that for a music recorded at 44100 Hz sample rate, 100 envelope values per second are calculated. In this way, the points were calculated every 10ms as the average of a 20ms window centred on the point. For the peak-picking, it is necessary to calculate the *slope* of the envelope at each point. This is achieved by using a 4 points linear regression method. For each point *j*, the linear regression is processed for points *(k,E[j+k])* with *k=0,1,2,3.* Since, it is of interest only the line slope given by the linear regression, the same abscissa repair for each point is kept.

For *j∈[1;m-4]*, the slope is given by:

$$slope[j] = \frac{< kE[j+k] > - < k >< E[j+k] >}{< k^2 >< k^2 >} \qquad [3.5]$$

where $< F(k) >= \sum_{i=0}^{3} F(i)$.

Once the slope is calculated, the array *slope[]* is processed to find local maxima. This processing algorithm is based on Schloss' "surf board" method [14]. In the experimental tests, local peaks were rejected if there was a greater peak within 50ms (*pickWidth*) or if their amplitudes (*thresholdFactor*) were below 10% of the mean amplitude. The *slide*, *pickWidth* and *thresholdFactor* parameters may have to be adapted to the type of music considered.

4.3 Monophonic Pitch Recognition

The pitch detection algorithm for the monophonic mode is focussed on the fundamental frequency detection and it is performed in the sustain phase. It calculates the weighed sums of the peaks related to the output signals of the auditory channels of the Meddis model. Each channel is centred on the fundamental harmonic frequency of a specific note that has to be recognised. The output of each channel is a quasi-periodical signal whose maximal amplitude is directly proportional to the probability that the specific harmonic is present in the spectrum of input signal. Provided that ***m_out_{ij}*** is the value of the peak related to the jth harmonic frequency referred to the ith gammatone filter centred on the ***CF[i]*** frequency, the pitch of the note is given by:

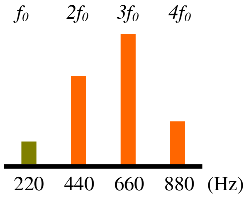
$$pitch = CF[\max_i \sum_j w_j m\_out_{ij}] \qquad [3.6]$$

with $j = 1 \ldots NUM\_HARM$, and where:

- $\boldsymbol{m\_out_{ij}}$ is calculated considering BUFFERSIZE consecutive samples far ONSET_DELAY samples from the note attack instant;
- $\boldsymbol{w_j}$ is the weight associated with the jth harmonic related to the used instrument. Thus, it is the jth component of the WEIGHT_HARM array.

To understand the role of the $\boldsymbol{w_j}$ weights, let us consider the following occurrence. Provided that:

1. the music instrument is unknown;
2. the pitch of the note perceived by the ear is an A2 at 220Hz;
3. the spectrum of the first four harmonic is as depicted below:



A2

If the pitch detection algorithm is designed to consider the harmonic with the maximal amplitude, it would provide the result associated with the third harmonic (660 Hz), but this result would be completely wrong.

On the other hand, knowing the spectrum features of the instrument and the relationships among single harmonic amplitudes (for instance, the third harmonic has an amplitude five times bigger than the fundamental one) allows characterising the sound played by the instrument. In this sense, weights are characteristic of the instrument and in equation (3.6) they allow detecting the pitch by identifying the Meddis auditory channel referred to the fundamental harmonic. Finally thanks to them, it is possible to define a generic pitch detection algorithm, and to cope with different music instruments and different features, simply changing the number of partials to be involved and associated weights.

4.4 Monophonic Offset Detection

In this phase, the note off detection is performed. For each note detected by the onset detection algorithm, a set of BUFFERSIZE samples, following the audio samples already used for the pitch detection of that note, is filtered by the Patterson-Meddis auditory model. The weighed sum of maximal excursions of the output signals coming from the channels of the Meddis model related to the first NUM_HARM_OS harmonics of the specific note is calculated. The used weights are those included in the WEIGTH_HARM array, already used in the pitch detection algorithm. This new value is compared with the value of the greatest excursion registered for that note in its sustain phase. If the value of the weighed sum of the signal excursions is less than a percentage (represented by the parameter OFFSET_THRESH) of the maximal excursion related to that note, a new offset is found. Otherwise, this process is repeated with a new set of BUFFERSIZE audio samples following the currently analysed samples, until a new offset is detected or until the current set of samples is related to the next note. In this last case, the onset value of this note is chosen as the new offset value for the analysing note assuming that there is not any pause between the note and the next one.

5. Experimental Results

In this section, experimental results obtained in monophonic mode are presented. The system has been tested on several phonographic recordings executed by different musical instruments. In particular, a notebook microphone has been used for all recordings – thus being characterised by a strong background noise and low quality – to test the transcription robustness when coping with background noise due to the real condition of work and low cost devices. Finally, the percentage of recognised notes considered in all the tests is related to the starting times and the pitch values of notes.

5.1 Monophonic tests

The monophonic transcription model has been tested on a range of phonographic recordings executed by different musical instruments: guitar, piano and violin. For these instruments, the configuration parameters reported in Table 1 have been used. In detail, for piano and guitar the number of harmonic frequencies involved in the pitch detection phase was fixed to the three first partials, whereas for the violin the fourth harmonic partial was added.

First, the transcription of a set of monophonic excerpts for classical guitar has been carried out. The selected guitar is of average quality and more specifically this means the avoiding of new strings (on general terms, they make the recognition process easier by reducing the appearance of signal discordant components). Furthermore, the recordings allowed a deeper appreciation of the transcription robustness when coping with background noise that is a distinguishing mark of microphone recordings. Table 2 shows the transcription outcomes related to a subset of tests played by the guitar.

Title of the piece	Number of analysed notes	Percentage of recognised notes
Sakura	34	97.4 %
Studio di D. Aguado	42	100.0 %
Der Weichnachtsmann	40	85.0 %
The Muffin Man	29	86.2 %
Red River	38	86.9 %
Chromatic Scale	37	91.8 %
Ballata lombarda	39	97.4 %
Canto trad. siciliano	47	91.5 %
Canto trad. cinese	55	100.0 %
Canto trad. tedesco	57	86.0 %
Scala di Do mag.	15	93.3 %
Arpeggio di Do mag.	5	86.9 %
Studio di F. Carulli	100	93.0 %

Table 2 - Monophonic Transcription of classical guitar

In the tests carried out for the monophonic recognition of the classical guitar, the transcription achieved an average accuracy of about **92%**. Furthermore, the percentage of transcribing mistakes (56.1%) must be put down to octave mistakes, the 36.6 % to semitone mistakes and the rest of it (7.3%) to mistakes of other nature. It is worth noting that 100% of semitone mistakes involved only some notes lower than A3 (with fundamental equal to 220 Hz), namely the instrument’s first six lowest notes (the instrument has a total extension of 47 notes), in a spectral area where the resolution in frequency becomes critical.

In Table 3, a list of outcomes obtained by transcribing a set of monophonic pieces executed by a vertical piano is reported, always using microphone recordings and thus being characterised by a strong background noise.

Also in this second example the system achieved level over **96%**, thus giving evidence of the entire system’s robustness to cope with background noise. In this instance, the percentage of mistakes (75%) is mainly due to semitone mistakes, while the rest of it (21.4%) is due to octave mistakes.

Title of the piece	Number of analysed notes	Percentage of recognised notes
Inno alla gioia	63	96.8 %
Sinfonia dal N.Mondo	66	97.0 %
Minuetto di J.S.Bach	113	91.2 %
Notturno di F.Chopin	73	95.9 %
Per Elisa	80	93.8 %

Romanza in Fa	62	100.0 %
Scala in Do mag.	16	100.0 %
Valzer dal L.dei Cigni	62	90.3 %
Il mattino di E.Grieg	25	100.0 %

Table 3 - Monophonic Transcription of piano

Table 4 shows a set of results concerning the automatic transcription of monophonic violin pieces. The transcription is based on microphone recordings except for the chorale of J. S. Bach played with a synthesised violin. This last piece was first composed via MIDI and then converted in an audio file using a MIDI to WAVE converter.

Title of the piece	**Number of analysed notes**	**Percentage of recognised notes**
Chromatic Scale	48	87.5 %
Corale di J.S.Bach (synthetic)	62	99.4 %

Table 4 - Monophonic Transcription of violin

When it comes to monophonic violin recognition, the 71.4% of mistakes are due to octave mistakes while the rest (28.6%) has to do with semitone mistakes. No other kinds of mistake have been detected and the average accuracy is fairly close to **97%**.

5.2 Comparison to Other Approaches

The lack of a standard set of test examples makes comparison of different transcription systems a difficult task. To produce an effective and precise cross evaluation with other works is a difficult task, for the lack of both an audio test database and assessment rules does not allow an objective consideration of results. Many analysed works do not show tables or statistical evaluation in terms of average accuracy or error rate in pitch detection when the system works in real conditions. The task is further complicated by the fact that systems put very different constraints on the type or style of music they transcribe. However, some comparison can be carried out with some restraint, as the transcribed pieces were recorded with real or synthesized music instruments under different conditions: single notes or melody frames analysis, SNR related to the recording channel, etc. In this section, the performance of solution and system described in this paper is compared with results of [25] and [28], A generic evaluation compared with other proposals.is also reported .

In [25], a solution for pitch recognition of guitar playing has been proposed. The approach used in [25] is based on two intelligent neural networks (INN) indicated by the author as “small” and “big” network. The small network was trained to recognise only one specific pitch of the guitar with an average accuracy of 91.2%, whereas the big one was build by means of a set of small networks, each of them is used as neuron in the hidden layer in order to recognise 49 pitches of guitar playing with an average accuracy of 97.90%. For the humming case, a new big network was trained to recognise only 10 pitches; it provided an average accuracy of 91.3%. The solution described in [28] is very closed to the one proposed in the paper. In fact, it consists of an onset detection functions module associated with a peak picking module, a fundamental frequency estimation and the final note decision module. Two phase vocoders are used in parallel for both onset and pitch detections. Audio waveforms generated from MIDI files were used to test system and were analysed to perform a pitch tracking. For evaluation purposes, test cases were chosen amongst various single voiced scores for piano, violin, clarinet, trumpet, flute and oboe. The performance for the whole set of MIDI test files reaches 90% of correct note labelling.

These results show that the proposed solution seems to be precise with respect to the solution of [25] for a difference of 5% in the recognition rate of the guitar sound; while the proposed solution is better ranked with respect to that proposed in [28]. On the other hand, the solution [25], is not flexible to work with different instruments. In fact, to cope with a new instrument, in [25], a new neural network has to be trained, whereas for the proposed solution it is enough to set the configuration parameters related to the specific instrument. Therefore, the neural network approach trained on each specific music instrument resulted to less flexible. In addition, it should be noted that tests performed during the system evaluation were made by using real instruments and a low cost microphone. Thus, all audio waveforms are affected by the background noise and the noise due to the quality of the microphone. In fact, when the system has been tested with a violin audio waveform generated from a MIDI file by means of a MIDI to WAVE converter, the recognition rated has been of 99.4% of right pitches; this gives evidence that the system is relatively immune to the noise.

In case of approaches not oriented to specific music instruments, the system proposed has been compared following the criteria and considering the methods described in [27]. The results are reported in Table 5 showing also the approaches affected by octave errors.

Method	**Domain**	**Simplicity**	**Noise Robustness**	**Inharmonicity Robustness**	**Spectral peculiarities Robustness**	**Octave Errors**
ZCR ([3])	Time	Very simple				Yes
AutoCorrelation Function ([4], [20], [30])	Time/Freq.	Simple Relatively	Noise immune		Sensitive	Yes
Envelope Periodicity ([33], [34]))	Time	Simple				Yes
Time Parallel Proessing/Rabiner ([36])	Time	Relatively simple				**Yes**
Spectrum AutoCorrelation ([18])	Frequency	Simple				Yes
Harmonic Matching Methods ([19])	Frequency	Quite Complex	Relatively noise immune		Relatively immune	Yes
Wavelet based method ([31])	Frequency (WT)	Quite Complex	Noise immune			
Bandwise/Klapuri ([32])	Frequency	Quite Complex	Relatively immune	Relatively immune	Relatively immune	
Proposed System	Time	Simple	Relatively immune	Relatively immune		Yes

Table 5 - Comparison with other methods

6. Conclusions

The proposed architecture is related to an automatic transcriber for monophonic music, based on the use of an auditory model and a bank of neural networks for notes detection. In this paper, the monophonic transcription mode has been detailed. The transcriber was conceived to be quickly parameterised for the recognition of an arbitrary music instrument, through a first configuration module and a second module, which allows a wider recognition range concerning also other music instruments. Many transcription results concerning the violin, the guitar and the piano have been depicted. More specifically the system turned out to be pretty robust to the background noise, which is a distinguishing mark of microphone recordings. Last, coming to the tests and the comparison with other approaches carried out, the system could provide a fairly encouraging accuracy in the transcription results. Needless to say, the described method for monophonic music could undergo several improvements like the implementation of an onset detection algorithm working in the domain of frequency. The algorithm currently in use (working completely in the domain of time) could succeed in detecting correctly the attack instant of almost all notes played by instruments such as the piano and the guitar. Yet, it had some limitations for the notes executed by the violin, which is due to the particular envelope of the wave shape, having a transitory attack much more gradual than that of other instruments like the guitar and the piano. On the other hand, an onset detection algorithm working in the domain of frequency could more easily detect the attacks of such violin notes.

References

[1] Slaney, M. & Lyon, R. F. (1991). Apple Hearing Demo Reel. Apple Computer, Inc. Technical Report #25, Cupertino CA.M. (http://www.slaney.org/malcolm/pubs.html)

[2] M. Slaney, slaney93efficient.pdf http://www.slaney.org/malcolm/pubs.html

[3] Moorer, J.A, (1995). On the Segmentation and Analysis of Continuous Musical Sound by Digital Computer, Ph.D.Thesis, Dept. of Music, Stanford University, 1975.

[4] Rabiner, L. R.. On the Use of Autocorrelation Analysis for Pitch Detection, IEEE Transactions on Acoustics, Speech and Signal Processing, vol. 25, Feb. 1977, pp. 23-33.

[5] R. Plomp, Aspect of Tone Sensation, Academic Press, London, 1976.

[6] Schouten, J. F. The Perception of Pitch, in Philips Tecnical Review, vol.5, p. 286, 1940.

[7] E. De Boer, “Pitch of Inharmonic Signals”, Nature, vol. 178, p.535, 1956; “On the “residue” and auditory pitch perception”, Handbook of Sensory Psychology, vol. V/3, Springer, Berlin, 1980

[8] X. Serra, G. D. Poli, A. Picialli, S. T. Pope, and C. Roads, “Musical Sound Modelling with Sinusoids Plus Noise”, Ed., Musical Signal Processing, Swets & Zeitlinger Publishers, 1997.

[9] Marolt M, (2001). SONIC: Transcription of Polyphonic Piano Music with Neural Networks. In Proceedings of Workshop on Current Research Directions in Computer Music, November 15-17, 2001.

[10] R. Meddis, Simulation of Mechanical to Neural Transduction in the Auditory Receptor, Journal of the Acoustical Society of America, vol.79, no.3, pp. 702-711, March 1986

[11] Bruno, I., Nesi, P. (2003). Automatic Synchronisation Based on Beat Tracking. In Proceedings of MAXIS 2003 Conference, Leeds, 2003.

[12] R. D. Patterson, K. Robinson, J. Holdsworth, D. McKeown, C.Zhang, andM. H. Allerhand, Complex sounds and auditory images, In Auditory Physiology and Perception, (Eds.) Y Cazals, L. Demany, K.Horner, Pergamon, Oxford, 1992, pp. 429-446.

[13] Dixon, S., Automatic extraction of tempo and beat from expressive performances. J. New Music Research, 30(1), 2001.

[14] Schloss, W. (1985). On the Automatic Transcription of Percussive Music: From Acoustic Signal to High Level Analysis. PhD thesis, Stanford University, CCRMA.

[15] P. Cosi (1993), "On The Use of Auditory Models in Speech Technology", in V. Roberto Ed., "*Lecture Notes in Artificial Intelligence: Intelligent Perception Systems",* Springer Verlag Publisher, *Vol. 745,* 1993.

[16] Klapuri, A. (1998). Automatic transcription of muisc. Master thesis, Tampere University of Technology, Department of Information Technology.

[17] Pertusa, A., Inesta, J. M. (2003). Polyphonic music transcription thtough dynamic networks and spectral pattern identification. In Proceedings of the First IAPR-TC3 Workshop on Artificial Neural Networks in Pattern Recognition, Florence (Italy), September 2003, pp. 19-25.

[18] A. Lahat, R. J. Niederjohn, and D. A. Krubsack. A spectral autocorrelation method for measurement of the fundamental frequency of noise-corrupted speech. IEEE Transactions on Acoustics, Speech and Signal Processing, 35(6):741–750, 1987.

[19] R. C. Maher and J. W. Beauchamp. Fundamental frequency estimation of musical signals using a two-way mismatch procedure. Journal of the Acoustic Society of America, 95:2254–2263, 1993.

[20] Cheveigné, A. and Kawahara, H. (2002). YIN, a fundamental frequency estimator for speech and music. Journal of the Acoustic Society of America, 111(4), 1917–1930.

[21] Clarisse, L. P. et al. (2002). An auditory model based transcriber of singing sequences. In *Proceedings of the International Conference on Music Information Retrieval*, (pp. 116–123).

[22] A. T. Cemgil. Polyphonic pitch identification and bayesian inference. In *Proceedings of the International Computer Music Conference*, Miami, FL, 2004.

[23] Raphael, C. (2002). Automatic transcription of piano music. In *Proc. Int. Symposium on Music Inform. Retriev. (ISMIR)*, Paris.

[24] Alexander Sheh and Daniel P.W. Ellis. Chord segmentation and recognition using EM-trained hidden Markov models. In *4th International Symposium on Music Information Retrieval ISMIR-03*, Baltimore, October 2003.

[25] Marolt M. (2004). Networks of Adaptive Oscillators for Partial Tracking and Transcription of Music Recordings, in Journal of New Music Research, Vol. 33, No. 1.

[26] Minyu Li and Tao Li. Pitch recognition based on intelligent neural network system. 2004 International Conference on Communications, Circuits and Systems, 2004. ICCCAS 2004. Volume: 2 , 27-29 June 2004.

[27] Gómez, E., Klapuri, A., Meudic, B., Melody Description and Extraction in the Context of Music Content Processing. Journal of New Music Research Vol.32(1), 2003.

[28] Brossier, P., Bello, J., Plumbley, M. (2004). Fast Labelling of Notes in Music Signals. Proc. of the 5th International Conference on Music Information Retrieval (ISMIR 2004), Barcelona, October 10-14.

[29] G. Monti and M. Sandler. Monophonic transcription with autocorrelation. In Proceedings of the COST G-6 Conference on Digital Audio Effects (DAFX-00), 2000.

[30] Talkin, D. (1995). Robust algorithm for pitch tracking. In Speech Coding and Synthesis, Kleijn, W. B and Paliwal, K. K editors, Elsevier Science B. V.

[31] Jehan, T. (1997). Musical signal parameter estimation. *CNMAT report*, 1997.

[32] A. Klapuri. Qualitative and quantitative aspects in the design of periodicity estimation algorithms. In *European Signal Processing Conference*, 2000.

[33] Ladislav O. Dolansky. An Instantaneous Pitch-Period Indicator", in The Journal of the Acoustic Society of America, volume 27, n° 1, Jan 1955.

[34] Jonas Engdegard (2000). "Signal Processing for a Real Time Phonetograph", Tal, musik och horsel KTH, Stockholm

[35] S. A. Abdallah and M. D. Plumbley. Polyphonic transcription by non-negative sparse coding of power spectra. In Proceedings of the 5th International Conference on Music Information Retrieval (ISMIR 2004), pp 318-325, Barcelona, Spain, October 10-14, 2004.

[36] Gold, B. & Rabiner, L. (1969). Parallel processing techniques for estimating pitch periods of speech in the time domain. Journal of the Acoustic Society of America, vol. 46, pp. 442-448.