Automatic Synchronisation Based On Beat Tracking

Ivan Bruno, Paolo Nesi
DSI, Department of Systems and Informatics, University of Florence
Via S. Marta 3, 50139 Florence, Italy
Fax: +39-055-4796363, Tel: +39-055-4796523
ivanb@dsi.unifi.it, nesi@dsi.unifi.it
See for the research group: http://www.dsi.unifi.it/~nesi/

Abstract:

Nowadays, computers are massively used to design, stock or play all kinds of music. Their performances are constantly improving, while next challenge is to allow interactions with human musicians. These interactions could consist in playing a background music following a musician's live performance or just in displaying the score corresponding with the part of the music being played by the musician. These issues are affected with many synchronization problems. In order to have a part of these problems solved, the main idea was to develop a algorithms and programs, which could recognize the beats of a polyphonic music piece so as to determine the positions of its measures. The obtained results have been validated by using several types of music.

Key-Words: beat tracking, pitch recognition, audio processing, automatic music recognition.

1 Introduction

Computers are massively used to design, stock or play all kinds of music. One of the last interesting features consists in playing a background music following a musician's live performance or just in displaying the score corresponding with the part of the music being played by the musician. These issues are affected by synchronisation problems. In order to solve a part of them the recognition of beats of a polyphonic music piece is mandatory.

The proposed system is expected to process complete music stocked in wave files (off-line process) or to listen to on-line music, identifying the measures in real time (on-line process). The developed global program can realise an off-line process as well as an on-line process. *The on-line architecture has been developed but not reported in this paper.* Several application fields can be taken into consideration for this system. Whenever used as an off-line process, it allows listening to a recorded music piece, while an application displays the score with a cursor indicating which measure is currently under execution [7]. For musicians who are used to create music by mixing MIDI tracks and “wave” instruments records, the recognition of the measures in the "wave" samples allows to insert them quite easily in a MIDI-formatted music piece. Whenever used as an on-line process, this system could listen to the live performance of a musician and display the score, following the interpretation.

It could also control the speed of the background music being played by the computer (a feature which has turned out to very useful for music tuition, for example). The “Department of Systems and Informatics” at the University of Florence (Italy) worked on the development of a software called WEDELMUSIC, [5]. This application offers a storage system for all kinds of data related to a music part (records, audio, video, scores, author's biography, images, animations, etc.) and some options like playing the music while the score is displayed on the screen. In order to check the performance of the automatic synchronisation, it has been tested with many music pieces of the WEDELMUSIC database and the results were compared with the synchronisation carried out on the same pieces by a set of experts. This paper describes the system that has been realized. After a description of the previous works linked to the subject, the general features of the system are presented. Then, the main modules of the system are described in separate sections. The experimental results produced by the adoption of the system to synchronize some classical polyphonic music pieces are reported in the last section.

2 Previous Works

The main issue in this project is the detection of the music beats. This step, also called "beat-tracking", is necessary to correctly identify the measures (which are, basically, groups of beats).

In the literature of the beat tracking, various methods can be found and classified on the basis of the level of music knowledge they require. Methods based on no music knowledge aim at tracking the beats without any knowledge of the notes or any

theoretical music feature. They just consider the music stream as an audio signal and work in the frequency-domain, such as in [3]. A group of resonators are applied to the signal divided into six frequency bands and the most energetic output indicates the current tempo. The value of the tempo position of beats allows setting by considering a regular interval between each beat. A real-time use of these methods requires a very powerful processing resource. Methods based on low-level of music knowledge just consider the elementary audio events called onsets. They transform the audio signal in a sequence of onsets and then use various algorithms to identify a periodic structure in this sequence. The recognition of the onsets can be performed either in the time- or frequency-domain. The latter are more efficient and allow the recognition of more onsets. The temporal approaches are easier and lighter and detect only a part of the onsets. In this case, the system in charge of beat patterns' recognition must be able to make extrapolations in order to identify incomplete patterns (with missing beats also called "ghost beats"). Most of the recent beat-tracking systems are based on onsets detection: Dixon [1], Roberts and Greenhough [2]. Methods based on higher levels of music knowledge, Goto and Muroaka [4], were developed in order to consider music elements such as chords or information given by the drum in modern music.

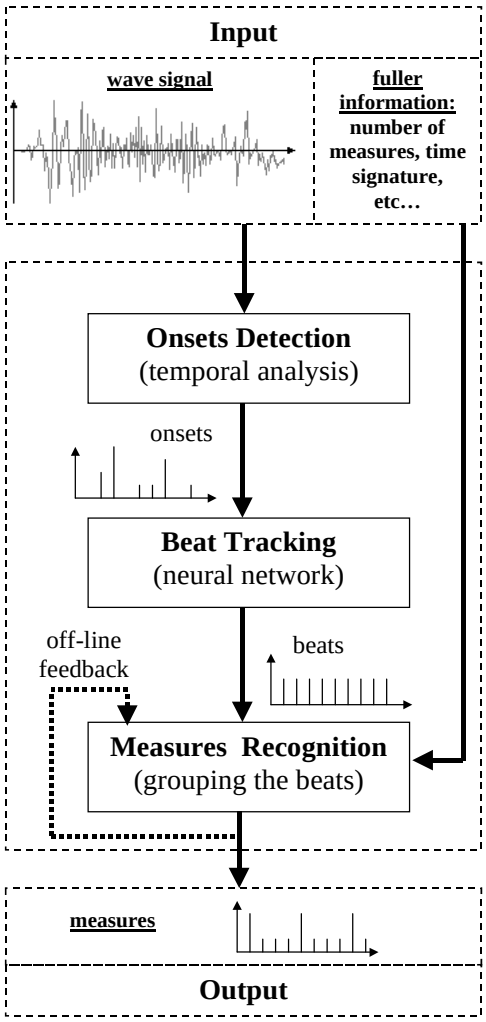

Fig.1 – General Architecture

3 General Architecture

The general architecture of the proposed system is based on low-level music knowledge and was developed to work off-line. It only considers the relevant audio events called *onsets* and a recorded audio. The Figure 1 shows the system architecture which presents 3 main modules:

Onsets Detection (OD) - The detection of the onsets is based on a time-domain analysis in order to obtain a short processing time.
Beat Tracking (BT) - It based on a self-organising neural network such as [2]. The network takes in input a partial detection of the onsets produced by the previous module.
Measure Recognition (MR) – It consists in defining the measures' position. This module takes advantage of the knowledge of the total number of measures, the number of beats in a measure etc.

The most important input is a wave file containing the music to process. Some other information is used in order to allow the measure recognition: time signature, music duration, and the number of measures. The wave file is in PCM format, with a 16 bits sample size, and a 44100 Hz sample rate. The onsets detection module processes the wave file and produces the list of the onsets. The beat tracking module processes onsets and produces the list of beats. These are processed by the measure recognition module to produce the list of the detected measures. A feedback loop has been added in order to improve the performance and find a number of measures closer to the expected one (see Figure 1).

4 Onsets Detection

The onset identifies a relevant sound variation 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. The onsets are expected to identify the important moments of a melody and, for some of them, the music's beats. The adopted method has been derived

from [2]. It consists in a time-domain analysis of the signal with the following steps:

- Filtering the audio file with a first order high-pass.
- Smoothing it to produce an amplitude envelope.
- Using a peak-picking algorithm to find the local maxima. Local peaks are rejected if there is a greater peak within a given time interval (fixed as a parameter) or if it is below a given threshold (limit value).

For estimation of the envelope, the points were calculated every 10 ms as the average of a 20 ms window centred on the estimation point. For the peak-picking, local peaks were rejected if there was a greater peak within 50 ms or if their amplitudes were below 10% of the average amplitude. All these parameters have to be adapted to the type of music considered. For the need of the project, it was decided to skip the first step (high pass filter) in order to work completely in the time-domain and to increase the performance. The two important steps for this method are the determination of the envelope and the extraction of the peaks. Figure 2 shows the results of these steps.

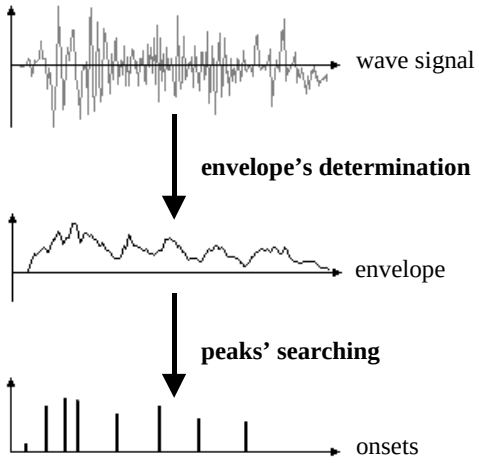

Fig. 2 – Onset Detection steps

5 Beat tracking

The Beat Tracking module has been realized by using and expanding the Neural Network, NN, solution proposed by Roberts and Greenhough [2]. It is based on a foot-tapper algorithm that tries to reproduce the human mechanisms observed during beat tracking experiments. The NN is a self-organizing and it is divided in two layers: input field and pulse field (P-Field). The input of the Input Field module is the set of the onsets detected in the Onset Detection module described above. The single onsets are sent to the module one by one. Between each onset the NN is updated from the last state producing all the intermediate states. This operation is performed every Trefresh seconds. According to [2], the input field contains a set of neurons (S-Cells) which are directly connected to all the cells of the P-Field. In the same input field there is also a G-cell that is entitled to keep trace of the status from one onset processing to the next. The P-Field is in charge of the beat pattern recognition. Each neuron of this layer corresponds to a possible interpretation of the beats. The number of the neurons of this layer is not imposed at the beginning of the process. During the process, every time that a new beat is detected, a new neuron is created to analyze it. During the process some other Neurons can be destroyed. The dynamic management of the neurons is the most important feature of the Self-organizing NN adopted. Each Neuron in the P-Field is characterized by 3 features:

- **P:** the period defining the time in seconds between the beats and its rhythm interpretation;
- **Φ:** a coefficient based on period P, which permits a simpler detection of the expected beats;
- **c:** the confidence level indicating the interpretation reliability.

This description is a simplification of the whole process and algorithm that presents several parameters. In our research, we have analyzed the behavior of the algorithm in order to better identify the meaning of the parameters discussed in [2] and to add more parameters to make the process more controllable and precise in the recognition and tracking. The most important parameters are listed and commented:

Tempo - this value corresponds to the duration in seconds of the beats. The estimations which are closer to α present a higher confidence. The typical value is of 0.6s, which corresponds to the typical duration of a whole note. It can be changed according to the music genres.

C range (thresholds) – They are limit values that identify the range in which the confidence value can be accepted. Values outside the range lead to destroy the neuron. Typical values are of 0.1 and 0.9. Increasing the lower threshold leads to increase the neurons life and thus the processing time.

Tl - latch time in seconds during which the confidence of an estimation is increased. It is the slop factor that controls the evolution of the confidence. The typical value is of 0.002s.

Φ_0 - the phase threshold (from 0 to 1). A tolerance value in estimating the tempo duration. A too high value can be too restrictive since only extremely

precise estimations will be considered good. In the experiments a value of 0.86 has been taken.
Number of S-cells in the input field: this parameter expresses the number of onsets that can be considered at each time instant. The typical value is of 20. **Number of G-cells** in the input field: this parameter is related to the process of renovation of old onsets leaving space to newer. Typical value is of 22. If these values are too small, the input field is not capable of to take into account a significant number of onsets to estimate the beat tracking.
Time to win - the time within which the NN has to produce an output neuron with a confidence value over the maximum C threshold. With a smaller value a lot of best estimations appear for a short time, while for long values only few of them are produced. A compromise is needed according to the music type, the value adopted in our experiments has been of 1.6s.
Trefresh has been introduced above. Increasing this value decreases the duration of the recognition process. Its value has to be lower than that of above Tl. Typical value, in our cases, has been 0.001s.

6 Measure Recognition

This module processes the beats found by the beat tracking module in order to define the position of the measures in the music. A measure is considered as a group of consecutive beats. The number of the beats grouped may vary during the process, it depends of the relevance of the beat rate found by the foot-tapper and on the information given by the user: time signature, music duration, number of measures and average duration of a full note. It is assumed that the beat tracking module may give non totally correct results. The found beat can sometimes be two or three times faster or slower than the real one and some phase problems may appear. The measure recognition module has been designed to compensate the faults. It includes the following steps:

- Identification of the constant beat parts
- Estimation of the real beat
- Grouping the beats

Before the beginning of the process, the variable $T_{average}$ has to be identified and corresponds with the average duration of a full note.

It is defined as follow:

$$T_{average} = \frac{\text{music's duration}}{\text{time signature} * \text{number of measures}} \quad [6.1]$$

Identification of the constant beat parts – The list of the beats is defined as (b_0, b_1,…,b_{n-1}), where b_i is the time when $(i+1)^{th}$ beat occurred. The constant beat parts are the lists of beats P_k defined as:

$$P_k = \left(b_{i_k} \quad b_{i_k+1} \quad \ldots \quad b_{i_k+n_k-1}\right)$$

The P_k list contains n_k beats, and starts with the $(i_k+1)^{th}$ beat. The list satisfies two conditions:

1) for $j \in [1, n_k - 1], \left(b_{i_k+j+1} - b_{i_k+j}\right) \approx \left(b_{i_k+1} - b_{i_k}\right)$
2) for $j \in [1, n_k - 1], \left(b_{i_k+j+1} - b_{i_k+j}\right) \approx \left(b_{i_k+j} - b_{i_k+j-1}\right)$

where the approximation noted "≈" is defined as following:

$$T_1 \approx T_2 \Leftrightarrow \min\left(\frac{T_1}{T_2}, \frac{T_2}{T_1}\right) > \Phi_0 \quad [6.2]$$

Where: Φ_0 has been defined above.

Estimation of the real beat – All the constant beat parts that have been identified are processed in the sequential order. For each of them:

- T_b, the average duration of a beat is calculated.
- T_f, the duration of the full note is defined as $c*T_b$ with $c \in N^* \cup \{1/n \mid n \in N\}$
- c is initialised to 1 and the increased or decreased until $T_{fmin} < T_f < T_{fmax}$.

The values T_{fmin} and T_{fmax} are calculated thanks to the variable $T_{average}$ as following:

$$T_{fmin} = T_{average} - \text{average\_window}$$
$$T_{fmin} = T_{average} - \text{average\_window}$$

The duration of the full note T_f corresponds with the most probable value for the real beat's duration. The tempo would be 60/ T_f beats per minutes.

Grouping the beats – n_b/m is the number of beats to be grouped in a measure and n_f/m the time signature (number of full notes grouped in a measure of score). The constant duration of a given measure allows writing:

$$T_b * \frac{n_b}{m} = T_f * \frac{n_f}{m} \quad [6.3]$$

with $T_f = c*T_b$, equation [6.3] becomes:

$$\frac{n_b}{m} = c * \frac{n_f}{m} \quad [6.4]$$

If $n_b/m \geq 1$, the measures are defined as consecutive periods containing n_b/m beats; else, the period between two beats is divided into subparts with a

duration equal to $(n_f/m)*T_f$. If the last identified measure leaves not enough time to add another measure before the end, then the process stops and the next analysis starts from this point. This mechanism allows balancing the time needed by the foot-tapper to validate a beat estimation. The beat identified for a part P_k, may have appeared a little bit before, that is why the validity of this estimation can be extended to the time remaining from the previous part. When this "additional time" (see Figure. 3) is associated with the start of a part some new beats are created and added at the beginning of the part. These beats exactly extend the distribution of the original beats (the time between two new beats is the average duration for the original beats). The process of the part is applied to all the beats (new and original) as one list of beats representing a constant beat part. At the end of a music piece, some beats could be missed because of the decreasing of note's intensity. In this case, the last recognized measure could finish a few seconds before the real end of the music. To deal with this problem, the systems adds new measures (with the same duration as the last recognised one) until the real end of the piece. If the numbers of measures is too great, only the first measures matching the expected number are kept and the last one's duration is modified so that it is longer and reaches the end of the music.

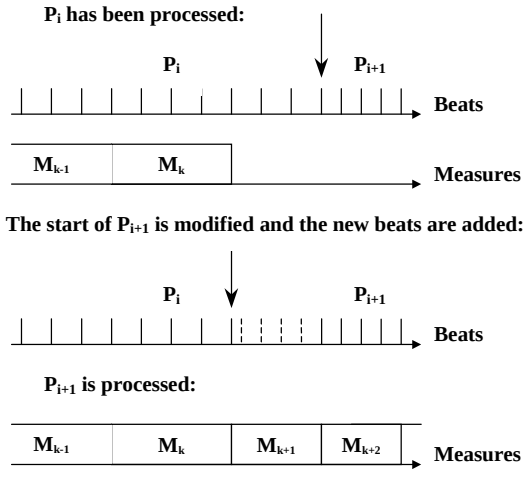

Fig. 3 – Additional time mechanism

7 Results

In order to validate the Measure Recognition process, it was applied to several music pieces. The synchronization of these pieces matching their audio with the measure was manually performed and verified by experts. In this section two examples are described.

"La Primavera" of A.Vivaldi - It is played by a group of string instruments. Some parts of the piece consist in a-solo while others are played by the group. A-solo are characterised by several notes that are not precisely phased on the regular tempo (most of these notes do not correspond with music beats). It is the main difficulty of this example. The parts played by the whole group allow recognising a quite regular rhythmic pattern. This example allows testing the ability of the system to get some information given by useful parts and not to loose this information during the parts that present synchronization problems. The tempo shows some variations from 85 to 95 full notes in a minute. That allows testing the sensitivity to tempo variations.

11th Symphony of J. S. Bach – The particular time signature (3/8) could be a problem for the system. There is only one instrument that may lead to clear detection of the onsets. If the onsets due to the instrument do not allow a good beat tracking, there are not any other instruments to complete the information. The tempo is quite regular during the whole piece. The interpretation recorded for this example adds some small decreases of the tempo at the end of many measures. They are not enough to modify the general tempo, but they could lead the NN to lose correct estimations since some beats are received later than expected.

For the synchronization, all the lists of measures found by the system were automatically adapted to match the expected number (adding or deleting the last measures). However, the global error Eg, calculated before this correction, gives information on the system's performance.

Parameters - The values of all the parameters for tests are given in Table 1. The value for the threshold factor limiting the onsets detection (THRESH_FACT) is set to 20, which allow a good detection for the two pieces.

Errors determination - For the realised tests, different errors between the found measures and the reference value measures defined by the experts were calculated:

- Global error Eg (%)
- Mean Duration Errror, Ed (s)
- Mean Relative Error, Er (%)
- Maximum time error, Emax (s)

For each considered piece of music, the Table 2 indicates the values of errors.

Variable	Parameter	Value	Module
slide	SLIDE	441	OD
thresholdFactor	THRESH_FACTOR	20	OD
peakWidth	PEAK_IDTH	5	OD
alpha	ALPHA	0.6	BT
Chigh	C_HIGH	0.9	BT
Clow	C_LOW	0.1	BT
T_l	LATCH_TIME	0.002 s	BT
Φ_0	PHASE_THRESH	0.86	BT
n_s	S-CELLS_COUNT	22	BT
G_0	GCELL_THRESH	20	BT
-	MAX_PCELLS_COUNT	500	BT
T_{win}	T_WIN	1.6 s	BT
-	ADAPT_PERIOD	1	BT
-	DESTROY_LOW_PCELLS	1	BT
v1, v2, v3, B	V1, V2, V3, B	1, 0.1, 134, 2	BT
$T_{refresh}$	REFRESH_STEP	0.001	BT
$step_{slope}$	AVERAGE_STEP	0.0001	MR
i_{max}	MAX_ITER	500	MR
average_window	*AVERAGE_WIN*	*0.2*	*MR*

Table 1 – Parameters for tests

	La Primavera	11th Symphony
Expected	83	72
Found	82	72
Eg	1.2 %	0.0 %
Ed	0.366 s	0.274 s
Er	14.6 %	8.567 %
Emax	10.6 s	8.307 s

Table 2 – Errors values

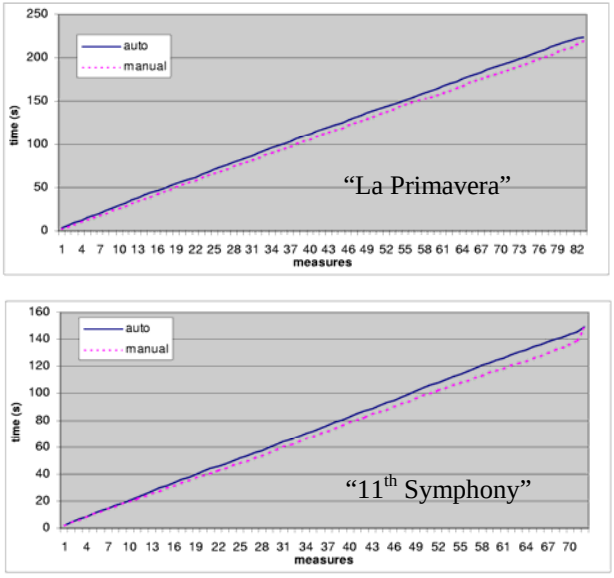

Fig. 4 – Graphical representation of synchronization

The results obtained (see Figure 4) demonstrate that the system proposed for measure estimation is quite precise even for polyphonic music and when the identification of beats is not simple. In all cases that we have used mean errors (Er) lower than 15% have been found with global errors always lower than the 2 %. This means that the systems can be profitably used for synchronising real audio with music score in automatic manner when pieces are smaller that 50 measures.

8 Conclusion

For innovative production of multimedia music products the synchronisation of audio with real music score is mandatory for educational and entertainment aspects. To this end, in this paper a system to estimate the measure in the audio has been presented. It is capable to produce results with a very interesting precision with respect to previous systems for beat tracking detection. It will be adopted as automatic synchronising tool in the WEDELMUSIC synchronisation process and tool [6]. This will shortening the process to produce multimedia music content.

References:

[1] S. Dixon, "Automatic Extraction of Tempo and Beat from Expressive Performances", Austrian Res. Inst. for Artif. Intell., 2001.

[2] S. C. Roberts and M. Greenhough, "Interpreting Rhythmic Structures Using Artificial Neural Networks", University of Wales, College of Cardiff, 1996.

[3] E. Scheirer, "Tempo and beat analysis of acoustic musical signals", Acoustical Society of America, 1998.

[4] Masataka Goto and Yoichi Muraoka, "An Audio-based Real-time Beat Tracking System and Its Applications", School of Science and Engineering, Waseda University, JAPAN.

[5] P. Bellini, J. Barthelemy, I. Bruno, R. Della Santa, P. Nesi, M. Spinu "Multimedia Music Distribution and Sharing among Mediateques, Archives and their Attendees" 2nd WEDELMUSIC Conference (2002),Germany.

[6] P. Bellini, I. Bruno, R. Della Santa, P. Nesi, M. B. Spinu, "Execution and Synchronisation of Music Score Pages and Real Performance Audios" Proc. of the IEEE Int. Conf. on Multimedia and Expo, Switzerland, 2002

[7] P. Bellini and F. Fioravanti and P. Nesi, "Managing Music in Orchestras," IEEE Computer, pp.26-34, September, 1999.