

FACULTAD DE INGENIERÍA DE LA
UNIVERSIDAD DE LA REPÚBLICA

PROCESAMIENTO DIGITAL DE SEÑALES DE
AUDIO

CURSO 2012

Beat Tracking

Autores:

Gonzalo GUTIÉRREZ

Matías TAILANIÁN



11 de diciembre de 2012

Introducción

Al escuchar música una reacción inconsciente muy común es mover el pie golpeando el piso a tiempo con el **beat**. La tarea computacional que intenta replicar ese comportamiento es conocida como **beat tracking**.

La métrica de la señal se puede pensar como una estructura de pulsos percibidos a diferentes escalas temporales en una pieza musical. Se consideran 3 niveles métricos básicos: el *tatum*, el *tactus* o *beat* y el *compás*. El *tatum* es el valor en tiempo más pequeño que puede encontrarse en una pieza musical, es la unidad atómica de la pieza. En general los otros valores de duración presentes en la pieza son múltiplos de *tatum*. El *tactus* o *beat* está más relacionado con el aspecto semántico, y está directamente vinculado con el **tempo** de la pieza. Por último el *compás* está vinculado con la tasa de cambios armónicos o la duración de un patrón rítmico.

El problema del seguimiento de *beat* o *beat tracking* es un problema todavía abierto, donde se sigue intentando con gran actividad lograr mejorar los resultados del estado del arte. Tiene un atractivo muy importante en sí mismo pensando en aplicaciones como el acompañamiento automático, asistencia a la hora del editado, estudios musicológicos, efectos de música adaptativos, o seguimiento del ritmo de una batería tocando en vivo, pero además es una parte imprescindible de algunos sistemas más complejos para aplicaciones como etiquetado de música, reconocimiento automático del género, análisis de similitud de música y para lograr la *transcripción automática de música*.

El presente trabajo presenta un algoritmo de seguimiento de *beat* de una pieza musical y se basa en el trabajo de “João Lobato Oliveira, Fabien Gouyon, Luis Gustavo Martins, Luis Paulo Reis”, titulado “IBT: A real time tempo and beat tracking system, presentado en la 11^a *International Society for Music Information Retrieval Conference, ISMIR*, en 2010” ([1]). Dicho trabajo se basa a su vez en el sistema **BeatRoot**, presentado en ([2]): Dixon S., “Automatic extraction of tempo and beat from expressive performances.”, *Journal of New Music Research*, 2001. De [2] se toma la idea de varios **agentes** compitiendo y llevando varias hipótesis de tempo y fase paralelamente, se agrega robustez ante entradas ruidosas y se implementa en tiempo real. Aún realizando todas las operaciones de forma causal y en tiempo real, se logra obtener resultados comparables con el estado del arte y se convierte además en el primer software *open source* de seguimiento de *beat* en tiempo real.

Algoritmo

Como se puede ver en la figura 1 el algoritmo consta de 3 fases fundamentales:

- **Audio Feature Extraction:** En esta etapa se transforma la señal de audio en 1 secuencia continua que caracteriza la información más relevante para el análisis rítmico. En esta etapa se basarán las siguientes partes del algoritmo.
- **Pre-Tracking:** Al finalizar el Pre-Tracking se tendrá un conjunto de hipótesis iniciales con respecto a posibles períodos y fases de los beats. Consta de 3 subetapas donde se estimarán las características que definen a un agente:
 - Período
 - Fase
 - Puntaje
- **Beat-tracking:** En esta etapa se propagan las hipótesis y se crean, matan y puntúan agentes.

A su vez se presenta un sistema de evaluación que decide en función del puntaje de cada agente, cuál es el más adecuado para la pieza musical: el **Agent Referee**.

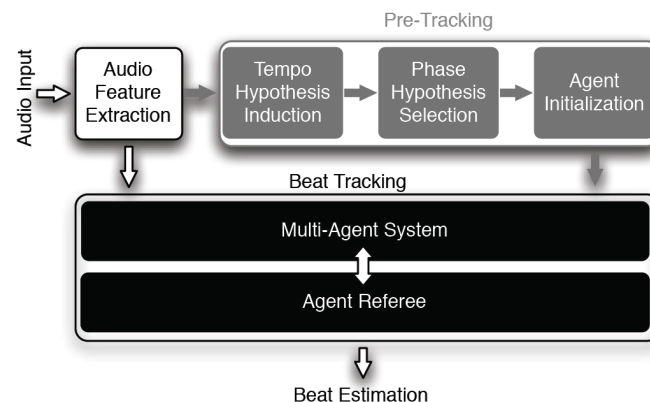


Figura 1: Diagrama de bloques

Audio Feature Extraction

Pre-Tracking

- **Período**

$$A(\tau) = \sum_{n=0}^m SF(n)SF(n + \tau)$$

SF es el flujo espectral suavizado para el *frame* n

$$\begin{cases} P_i = \arg \max_i \{A(\tau)\}, & i = 1, \dots, N \\ A(\tau) > \delta \frac{rms(A(\tau))}{M} \end{cases}$$

δ es un umbral determinado empíricamente y M es un rango de tiempos definido entre [50,250] BPM.

- **Fase**

Para cada P_i estimado se generan varias hipótesis para la fase: ϕ_i^j . Se supone fase y períodos constantes en cada ventana de análisis.

Se utiliza un *template* de tren de pulsos para ver cuál ajusta mejor.

- **Agentes**

Para cada pareja (P_i, ϕ_i) se computa la suma de errores entre el template de tren de pulsos y los máximos del flujo espectral

Beat-Tracking

Idea

Supervisar flujo de entrada y mantener balance entre inercia y rapidez de la respuesta

Niveles de tolerancia: $T_{in} \in [T_{in}^l, T_{in}^r]$ y $T_{out} \in [T_{out}^l, T_{in}^l] \cup [T_{in}^r, T_{out}^r]$

- $T_{in} \in [T_{in}^l, T_{in}^r]$

$$\begin{cases} P_i = P_i + 0,25 * error \\ \phi_i = \phi_i + 0,25 * error \end{cases}$$

- $T_{out} \in [T_{out}^l, T_{in}^l] \cup [T_{in}^r, T_{out}^r]$

El agente mantiene su período y fase y además crea 3 “hijos” variando dichos parámetros

Agent REferee

Evalúa la distancia entre la predicción del *beat* (b_p) y el máximo local (m) de SF .

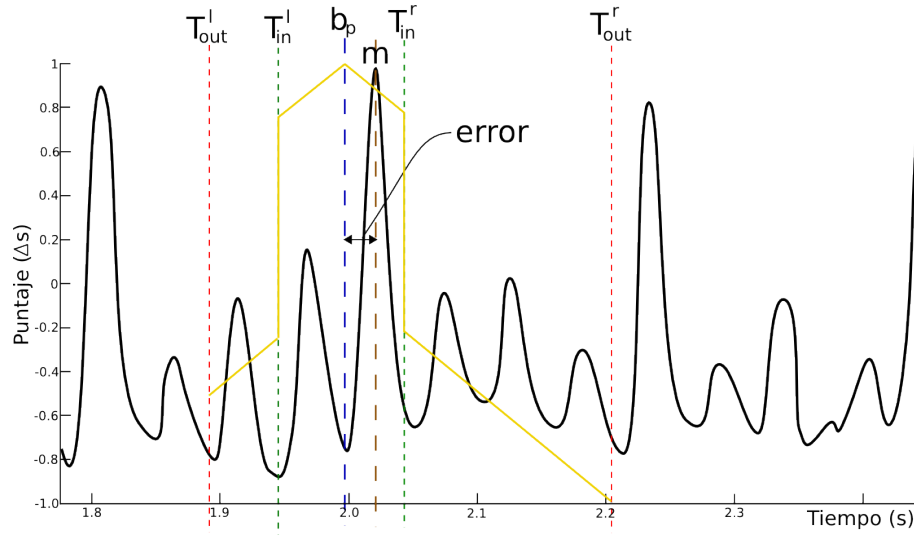


Figura 2: Niveles de tolerancia

P_m es el máximo período permitido

$$\begin{cases} \Delta s = \left(1 - \frac{|error|}{T_{out}^r}\right) \frac{P_i}{P_m} SF(m), & \exists m \in T_{in} \\ \Delta s = -\left(\frac{|error|}{T_{out}^r}\right) \frac{P_i}{P_m} SF(m), & \exists m \in T_{out} \end{cases}$$

Referencias

Bibliografía

- [1] João Lobato Oliveira, Fabien Gouyon, Luis Gustavo Martins, Luis Paulo Reis, IBT: A real time tempo and beat tracking system, In *11th International Society for Music Information Retrieval Conference, ISMIR*, 2010.
- [2] S. Dixon. Automatic extraction of tempo and beat from expressive performances. In *Journal of New Music Research*, 30(1):39–58, 2001.
- [3] S. Dixon. Onset detection revisited. In *in Proceedings of the 9th International Conference on Digital Audio Effects*, pages 133–13, Montreal, Canada, 2006.
- [4] F. Gouyon, P. Herrera, and P. Cano. Pulse-dependent analyses of percussive music. In *AES 22nd International Conference on Virtual, Synthetic and Entertainment Audio*, 2002.