Automatic Extraction Of Tempo and Beat From Expressive Performances

by Simon Dixon

presented by
Naru Sundar : CSCI575
Feb 3, 2005

Introduction

- Untrained listeners can still "tap to the beat"
- **Task: design a program to discover the underlying beat without any a priori musical knowledge**
- Technique involves using onset times of notes and some notion of musical salience

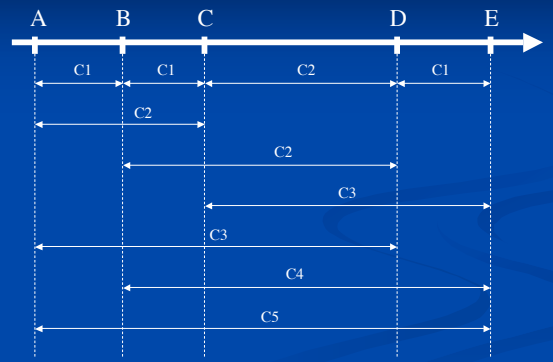
Algorithm

- Initially, the input data is converted into onset information
- First phase consists of creating tempo hypotheses based on note onsets
- Hypotheses are ranked based on relevance
- Second phase consists of creating agents base on these hypotheses to track beats
- Highest scoring agent wins

Generating Onsets

- Technique based on paper by Schloss ('85)
  1. Pass through high pass filter and smooth to produce amplitude envelope
  2. Compute slope of envelope
  3. Peak-picking algorithm finds local maxima

Tempo Induction



Tempo Induction

- Cluster pairs of onset events by interval time, based on comparison to average cluster interval time
- Combine clusters with similar interval averages
- Cluster score increased by size of clusters within integer multiples of the interval time

Tempo Induction

- Cluster score adjustment uses non-linear function to smooth score contribution from related clusters
- Each remaining cluster represents a tempo hypothesis
- Does *not* represent metrical information, beat detection phase will compute metrical information

Beat Detection

- Each tempo hypothesis is assigned an *agent*
- Each event in the input is processed and the agent scores are updated based on relevance and note salience
- Results of the highest scoring agent is returned

Beat Detection

- Each agent maintains a history of the events that it considers to be "on beat"
- Count intervals from last predicted time to current event (in units of agent's interval time)
- If current event falls within tolerance window of predicted event
  - Add event if inner tolerance requirement is met
  - Create new agent without the current event otherwise (I.e. keep existing note history and assume current event is not "on beat")

Note Salience

- Agent score is a function of the error between predicted beat time and actual event time, as well as a notion called "note salience"
- Note salience is a function of
  - Duration
  - Pitch
  - Amplitude
- Two types of functions tested: linear and multiplicative
  - S_add(d, p,v) = c1 * d + c2 * p[p_min, p_max] + c3 * v
  - S_mul(d,p,v) = d * (c4 - p[p_min, p_max]) * log(v)

Note Salience

- Constants were empirically detected
- Duration was the significant factor
- Dynamics and pitch used mainly to distinguish between notes of similar duration
- Experiments were tried both using salience function as well as without salience

Results

- Algorithm was tested on data ranging from pop songs to classical songs to jazz songs
- Data had varying levels of difficulty in terms of beat detection (empirically)
  - Pop songs were easiest to track
  - Classical songs varied
  - Jazz songs had complicated syncopation rhythmns that made them difficult to track even for people

Results

- Each song was broken up into ten second segments and the algorithm was applied to each
- Algorithm correctly identified beat for the most part within the first 2 or 3 segments
- Algorithm had high success rates
  - Worst score was 92.4% on the jazz song
- Short segment lengths produced worse scores on more complicated material (79.2% for 5s segments of the jazz song)

Note Salience Relevance

- Without using salience component, algorithm discovered 75% of correct beat times
- When using some salience component algorithm discovered as high as 91% of correct beat times

Future Research

- Transition to a realtime algorithm
- Current algorithm is too slow and not causal enough (requires analysis of large segments in order to succeed)
- Better notions of salience would help with more complicated input types