

Music-to-Score Temporal Alignment by Discriminative Graphical Models

Cyril Joder

TELECOM ParisTech

2011/09/29

Context: Automatic Indexing of Multimedia Document

- Huge databases of available multimedia documents
- **Meta-data** are needed for accessing and browsing these databases
 - tags (keywords), links, thumbnails, summaries, . . .
- Have to be created **automatically**

Special Case of Musical Contents

- Possible useful meta-data for music:
 - Scale, chord progressions
 - Meter (rhythm)
 - Main melody, pitches. . .

- Many of these pieces of information can be easily derived from the score
- One can take advantage of score databases

Special Case of Musical Contents

- Possible useful meta-data for music:
 - Scale, chord progressions
 - Meter (rhythm)
 - Main melody, pitches. . .

Autumn Leaves

Words & Music by Jacques Prevert, Johnny Mercer, Joseph Kosma

The fol - ling leaves drift by my win - dow, the au - tumn
 leaves of red and gold; I see your
 lips, the sum - mer kis - ses, the sun - burned
 hands I used to hold. Since you
 went a - way the days grow long, and soon I'll
 hear old win - ter's song, but I
 miss you most of all, my dar - ling, when
 au - tumn leaves start to fall.

Chord progressions: Cm⁷, F⁷, Bbmaj⁷, Ebmaj⁷, Am⁷(b9), D⁷, Gm⁶, Cm⁷, F⁷, Bbmaj⁷, Ebmaj⁷, Am⁷(b9), D⁷, Gm⁷, Cm⁷, F⁷, Bbmaj⁷, Ebmaj⁷, Am⁷(b9), D⁷, Gm⁷, Cm⁷, Fm⁷, Bb⁷, Ebmaj⁷, D⁷, Gm⁶.

- Many of these pieces of information can be easily derived from the score
- One can take advantage of score databases

Copyright reserved by MusicFly
 www.musicfly.com - Sponsored by Waldford Foundation - Music engraving by LilyPond

Special Case of Musical Contents

- Possible useful meta-data for music:
 - Scale, chord progressions
 - Meter (rhythm)
 - Main melody, pitches. . .

Autumn Leaves

Words & Music by Jacques Prevert, Johnny Mercer, Joseph Kosma

The fol - ling leaves drift by my win - dow, the au - tumn
 leaves of red and gold; I see your
 lips, the sum - mer kis - ses, the sun - burned
 hands I used to hold. Since you
 went a - way the days grow long, and soon I'll
 hear old win - ter's song, but I
 miss you most of all, my dar - ling, when
 au - tumn leaves start to fall.

- Many of these pieces of information can be easily derived from the score
- One can take advantage of score databases
- Needs **music-to-score alignment**.

Copyright reserved by MusicFly
 www.musicfly.com - Sponsored by Walidiana Foundation - Music engraving by Lilypond

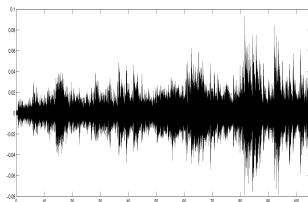
Music-to-Score Alignment

200

SONATE.
Op.22 No.2
(Sonata quasi una Fantasia)
für Violin Solo (Violoncello optional)

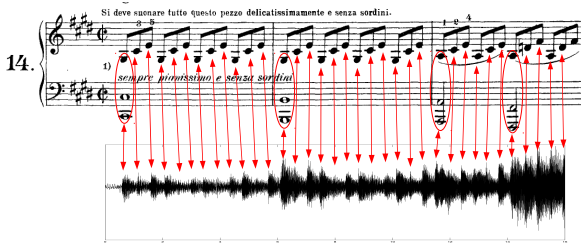
Adagio sostenuto.
Si non cessi (non più mosso) (Andaltesissimo e non cessi)

© P. Schötenberg



- **Data:** score and audio which match (same piece)

Music-to-Score Alignment



- **Data:** score and audio which match (same piece)
- **Goal:** find the correspondance between the positions in the score and the positions in the audio

Possible Applications

- Use of score for music indexing [Garbers,2008]
- Score-based browsing of a recording [Fremerey,2007]
- Music education (error spotting) [Montecchio,2008]
- Score retrieval from audio query [Hu,2003]
- Score-informed source separation [Hennequin,2011]

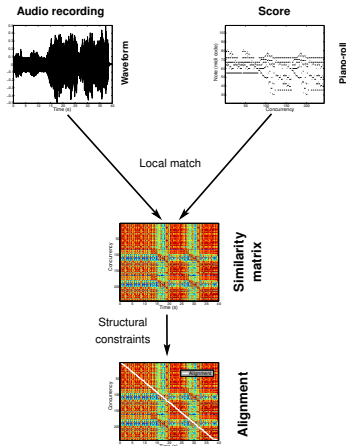
With real-time constraint:

- Computer accompaniment [Dannenberg,1984], [Raphael,2001], [Cont,2010]
- Automatic page turning [Arzt,2008]

Overview of an Alignment System

Two stages:

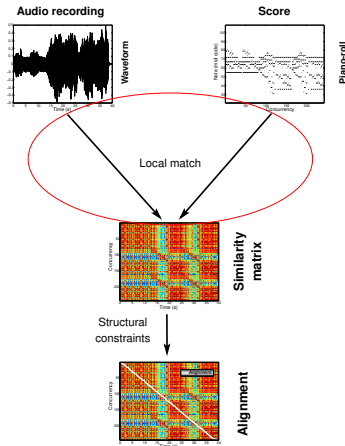
- Similarity matrix calculation: local matching measure
- Alignment: incorporation of structural constraints (transitions, durations)



Overview of an Alignment System

Similarity matrix calculation

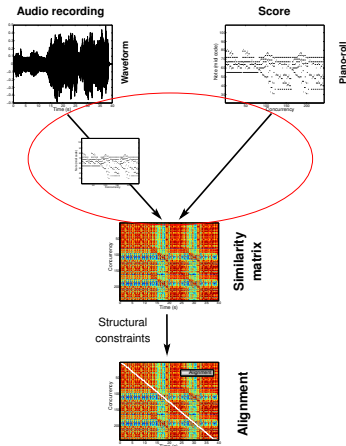
- Pitch extraction [Arifi, 2004]
→ error-prone
- Learning a generative model [Raphael, 1999] → intractable for polyphony
- Template-based [Orio, 2001]



Overview of an Alignment System

Similarity matrix calculation

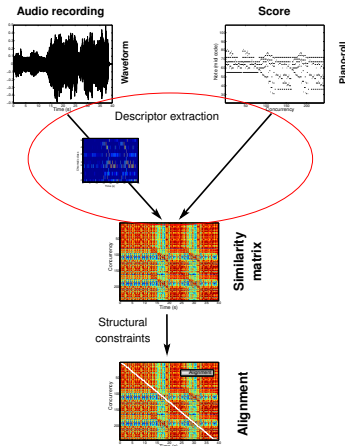
- Pitch extraction [Arifi, 2004]
→ error-prone
- Learning a generative model [Raphael, 1999] → intractable for polyphony
- Template-based [Orio, 2001]



Overview of an Alignment System

Similarity matrix calculation

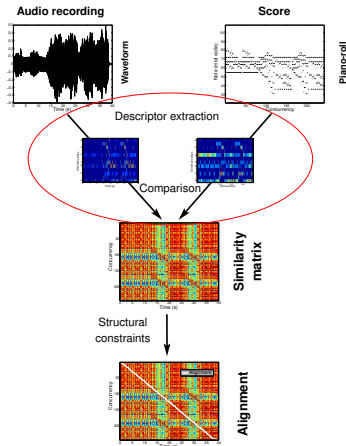
- Pitch extraction [Arifi, 2004]
→ error-prone
- Learning a generative model [Raphael, 1999] → intractable for polyphony
- Template-based [Orio, 2001]



Overview of an Alignment System

Similarity matrix calculation

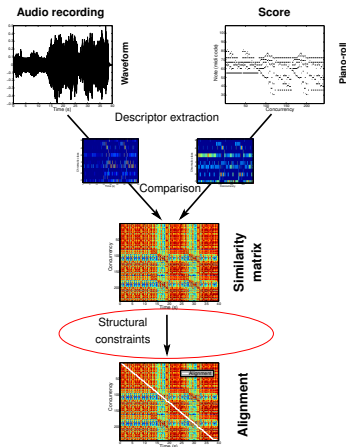
- Pitch extraction [Arifi, 2004]
→ error-prone
- Learning a generative model [Raphael, 1999] → intractable for polyphony
- **Template-based** [Orio, 2001]



Overview of an Alignment System

Alignment

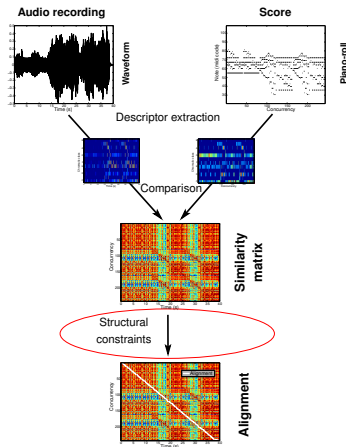
- Sequence alignment (DTW)
 - [Dannenberg,2003], [Dixon,2005], [Müller,2006]
- + simple and easy to implement
- difficult to control (implicit model)
- Statistical model (HMM)
 - [Orio,2001], [Grubb,1997], [Raphael,2006]
- + intuitive and flexible modeling, allows for parameter learning
- can be complex



Overview of an Alignment System

Alignment

- Sequence alignment (DTW)
 - [Dannenberg,2003], [Dixon,2005], [Müller,2006]
- + simple and easy to implement
- difficult to control (implicit model)
- **Statistical model** (HMM)
 - [Orio,2001], [Grubb,1997], [Raphael,2006]
- + intuitive and flexible modeling, allows for parameter learning
- can be complex



Guidelines for our Audio-to-Score Alignment System

Constraints

- Polyphonic music
- Any instrument
- No real-time constraint

Design choices

- Template-based matching measure
- Alignment by statistical model

Outline

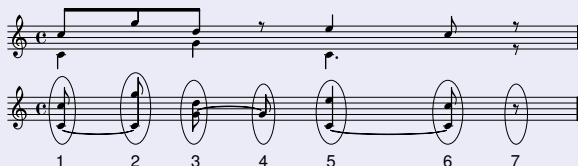
- 1 Music-to-Score Alignment: Introduction
- 2 Alignment by Statistical Model
- 3 Conditional Random Fields for Alignment
- 4 Modeling of Time
- 5 Optimization of the Concurrency Templates
- 6 Conclusion and Perspectives

Outline

- 1 Music-to-Score Alignment: Introduction
- 2 Alignment by Statistical Model**
 - Definitions
 - A First Simple System
- 3 Conditional Random Fields for Alignment
- 4 Modeling of Time
- 5 Optimization of the Concurrency Templates
- 6 Conclusion and Perspectives

Problem Definition

Score Segmentation into concurrencies [Raphael,2006]



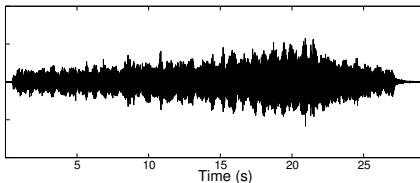
Statistical Model

- At each time n , random variable X_n representing the concurrency
- **Goal:** finding the most probable concurrency sequence

Audio Parameterization: Pitch Content

Representations used for alignment

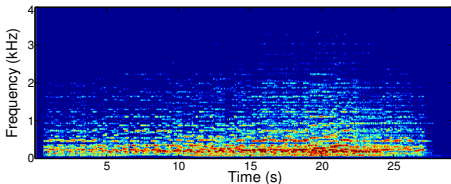
- Spectrogram: power spectrum in linear frequency scale (STFT) [Orio,2001]
- Semigram: power spectrum in logarithmic scale (semitones) [Montecchio,2009]
- Chromagram: “strength” of the 12 chromatic classes (wrapping of semigram on one octave) [Müller,2005]



Audio Parameterization: Pitch Content

Representations used for alignment

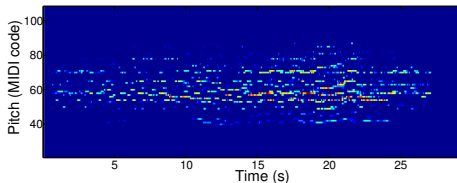
- **Spectrogram**: power spectrum in linear frequency scale (STFT) [Orio,2001]
- Semigram: power spectrum in logarithmic scale (semitones) [Montecchio,2009]
- Chromagram: “strength” of the 12 chromatic classes (wrapping of semigram on one octave) [Müller,2005]



Audio Parameterization: Pitch Content

Representations used for alignment

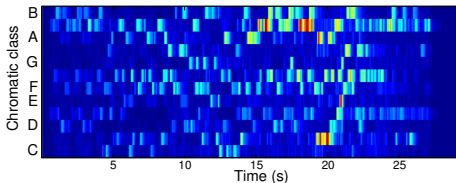
- Spectrogram: power spectrum in linear frequency scale (STFT) [Orio,2001]
- **Semigram**: power spectrum in logarithmic scale (semitones) [Montecchio,2009]
- Chromagram: “strength” of the 12 chromatic classes (wrapping of semigram on one octave) [Müller,2005]



Audio Parameterization: Pitch Content

Representations used for alignment

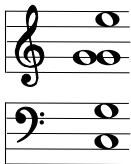
- Spectrogram: power spectrum in linear frequency scale (STFT) [Orio,2001]
- Semigram: power spectrum in logarithmic scale (semitones) [Montecchio,2009]
- **Chromagram**: “strength” of the 12 chromatic classes (wrapping of semigram on one octave) [Müller,2005]



Similarity Matrix Calculation

Concurrency:
symbolic representation

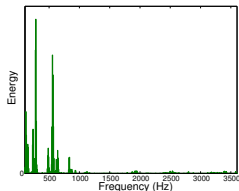
x



Matching
←→
Measure ?

Audio Observation:
time-frequency representation

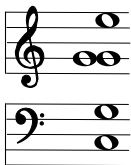
y



Similarity Matrix Calculation

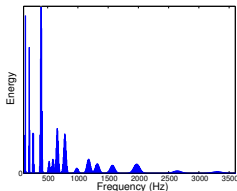
Concurrency:
symbolic representation

x



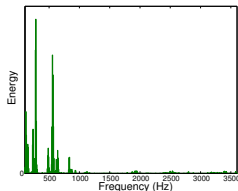
Template:
audio domain

u_x



Audio Observation:
time-frequency representation

y

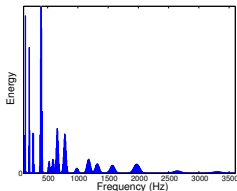


Similarity Matrix Calculation

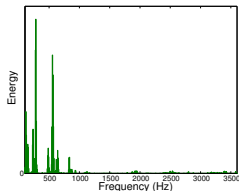
Concurrency:
symbolic representation

 x


Template:
audio domain

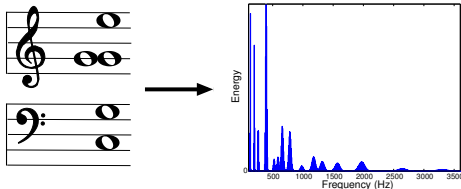
 u_x


Audio Observation:
time-frequency representation

 y

 $D(y, u_x)$

Template Construction

- Mapping from symbolic to audio domain
- Generally set by *ad hoc* rules
- Depends on the audio representation

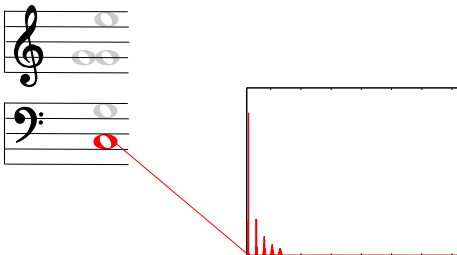


Template Construction: a Unified Framework



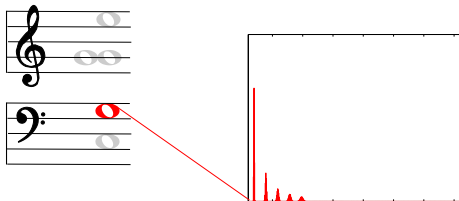
Template Construction: a Unified Framework

- Templates for isolated notes



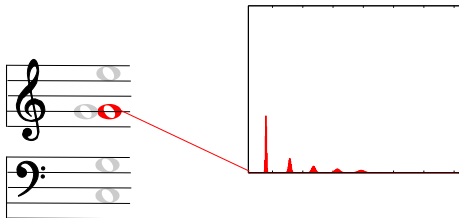
Template Construction: a Unified Framework

- Templates for isolated notes



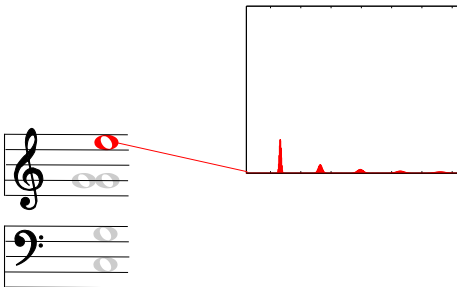
Template Construction: a Unified Framework

- Templates for isolated notes



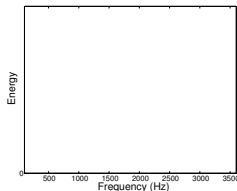
Template Construction: a Unified Framework

- Templates for isolated notes



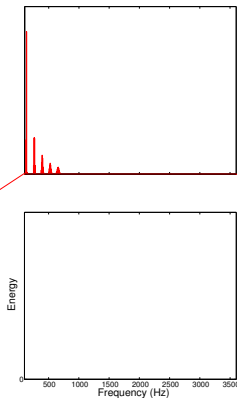
Template Construction: a Unified Framework

- Templates for isolated notes
- Superposition of one-note templates



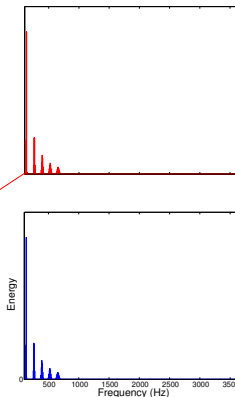
Template Construction: a Unified Framework

- Templates for isolated notes
- Superposition of one-note templates



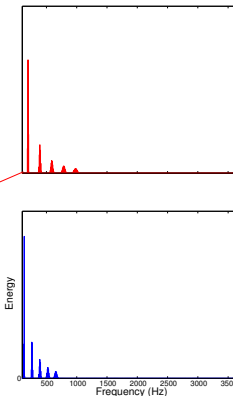
Template Construction: a Unified Framework

- Templates for isolated notes
- Superposition of one-note templates



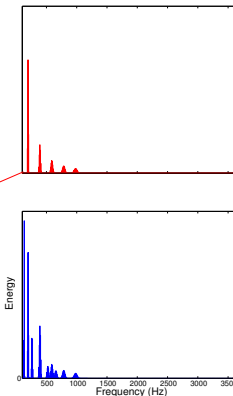
Template Construction: a Unified Framework

- Templates for isolated notes
- Superposition of one-note templates



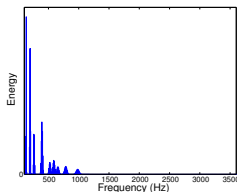
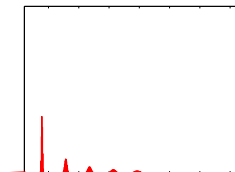
Template Construction: a Unified Framework

- Templates for isolated notes
- Superposition of one-note templates



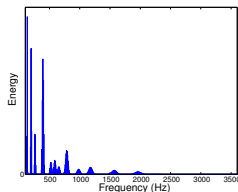
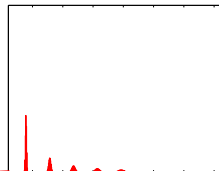
Template Construction: a Unified Framework

- Templates for isolated notes
- Superposition of one-note templates



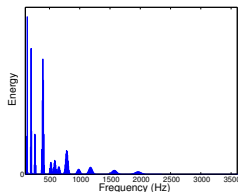
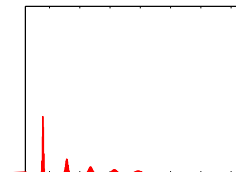
Template Construction: a Unified Framework

- Templates for isolated notes
- Superposition of one-note templates



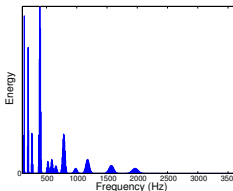
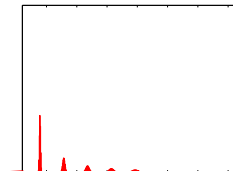
Template Construction: a Unified Framework

- Templates for isolated notes
- Superposition of one-note templates



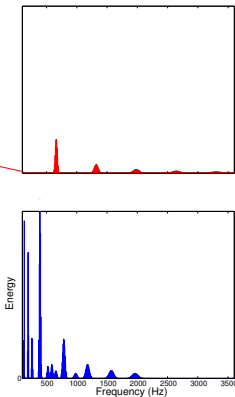
Template Construction: a Unified Framework

- Templates for isolated notes
- Superposition of one-note templates



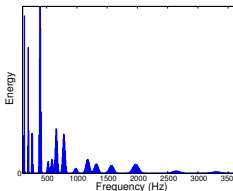
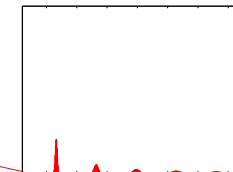
Template Construction: a Unified Framework

- Templates for isolated notes
- Superposition of one-note templates



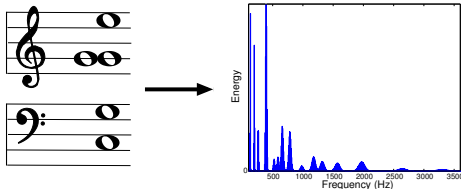
Template Construction: a Unified Framework

- Templates for isolated notes
- Superposition of one-note templates



Template Construction: a Unified Framework

- Templates for isolated notes
- Superposition of one-note templates



Template Construction: a Unified Framework

- Templates for isolated notes
- Superposition of one-note templates
- Advantage: only a few templates to set



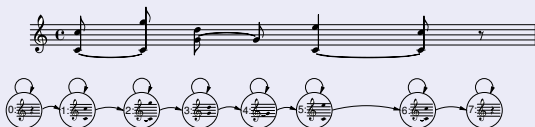
First Alignment System

Similarity matrix calculation (chosen after [1])

- Chromagram representation
- Kullback-Leibler divergence

Alignment strategy

- Structural constraint: no concurrency skipping
- Hidden Markov Model:



[1] C. Joder, S. Essid & G. Richard: *A comparative study of tonal acoustic features for a symbolic level music-to-score alignment*. ICASSP 2010

Database

Two corpora

- MAPS [Emiya,2010]: 49 classical piano pieces ($\approx 4h15$)
 - Ground-truth: aligned MIDI files
 - Scores: tempo modified to be constant
- RWC-pop [Goto,2002]: 90 pop songs ($\approx 6h$)
 - Ground-truth: aligned MIDI files
 - Scores: random tempo changes (piecewise constant)

Learning and Test Databases

- Learning: 50 pieces (20 from MAPS & 30 from RWC)
- Test: remaining 99 pieces

Results

Evaluation Measure

- **Alignment rate**: proportion of onsets recognized inside a tolerance window of θ around ground truth.





Results

Evaluation Measure

- **Alignment rate**: proportion of onsets recognized inside a tolerance window of θ around ground truth.

Results

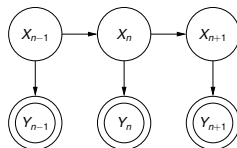
Alignment Rates for $\theta = 300$ ms:

| | | |
|---|---|---|
|  | MAPS corpus | RWC corpus |
|  | 87.8% | 72.4% |
| |  |  |

- Globally follows the important changes
- Poor fine-level alignment when numerous notes overlap
 - Noisy observations (drums, reverberation. . .)

Limitation of the Current Approach

- **Need:** more robust similarity matrix
- **Idea:** use neighboring observations
- **However:** conditional independence of the observations in HMM



- Requires a **more flexible statistical framework**

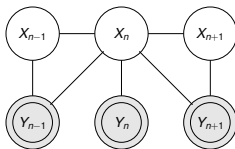
Outline

- 1 Music-to-Score Alignment: Introduction
- 2 Alignment by Statistical Model
- 3 Conditional Random Fields for Alignment**
 - Definition
 - Exploiting Neighboring Observations
 - Fusion of Several Descriptors
 - Experiments
- 4 Modeling of Time
- 5 Optimization of the Concurrency Templates

Definition

Conditional Random Fields

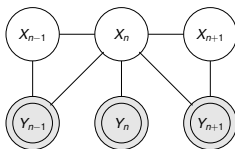
Discriminative undirected graphical model



- Conditioned on the observations:
 - no independence hypothesis
 - “local match” can depend on any observations
- No need for proper conditional probabilities
 - flexible penalty functions
 - weights of different features can be adjusted
- Allows for discriminative learning
- Same decoding complexity as HMM (Viterbi algorithm)

Definition

Conditional Random Fields



Probability of a **label** sequence $\mathbf{X}_{1:N}$, given the observation sequence $\mathbf{Y}_{1:N}$:

$$P(\mathbf{X}_{1:N}|\mathbf{Y}_{1:N}) = \frac{1}{Z} \phi(X_1, \mathbf{Y}_{1:N}) \prod_{n=2}^N \psi(X_n, X_{n-1}) \phi(X_n, \mathbf{Y}_{1:N})$$

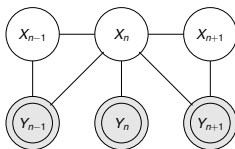
ϕ : observation function \rightarrow local match

ψ : transition function \rightarrow structural constraints

Z : normalization factor

Definition

Conditional Random Fields



Probability of a **label** sequence $\mathbf{X}_{1:N}$, given the observation sequence $\mathbf{Y}_{1:N}$:

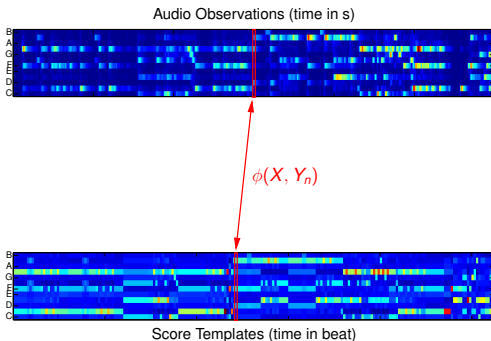
$$P(\mathbf{X}_{1:N} | \mathbf{Y}_{1:N}) = \frac{1}{Z} \phi(X_1, \mathbf{Y}_{1:N}) \prod_{n=2}^N \psi(X_n, X_{n-1}) \phi(X_n, \mathbf{Y}_{1:N})$$

ϕ : **observation function** \rightarrow local match

$$\phi(X_n, \mathbf{Y}_{1:N}) = \exp \sum_i \mu_i f_i(X_n, \mathbf{Y}_{1:N})$$

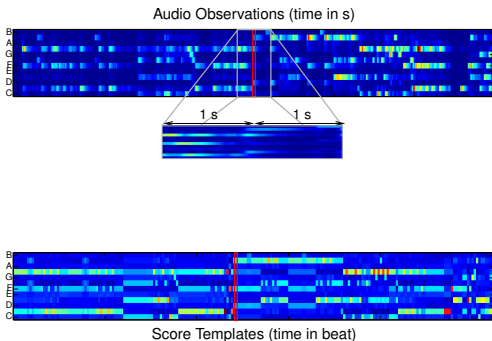
Exploiting Neighboring Observations

Pitch Feature: Neighborhood Integration



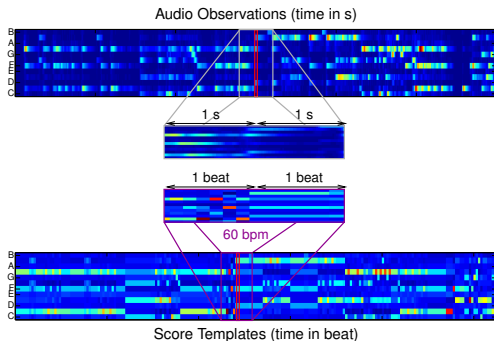
Exploiting Neighboring Observations

Pitch Feature: Neighborhood Integration



Exploiting Neighboring Observations

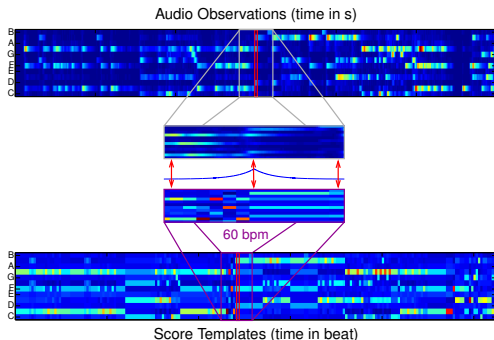
Pitch Feature: Neighborhood Integration



Hypothesis: locally constant tempo T_n (in the label variable X_n)
 → template sequence $u_{n-\nu}, \dots, u_{n+\nu}$

Exploiting Neighboring Observations

Pitch Feature: Neighborhood Integration



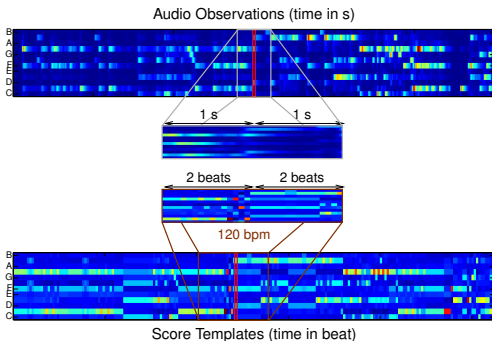
Hypothesis: locally constant tempo T_n (in the label variable X_n)

→ template sequence $u_{n-\nu}, \dots, u_{n+\nu}$

$$\phi(X_n, \mathbf{y}) = \exp \sum_{k=-\nu}^{\nu} -\mu_k D(y_{n+k} \| u_{n+k})$$

Exploiting Neighboring Observations

Pitch Feature: Neighborhood Integration



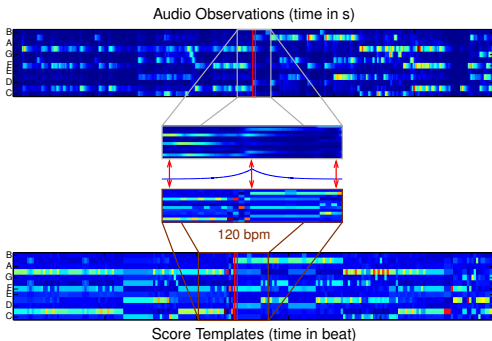
Hypothesis: locally constant tempo T_n (in the label variable X_n)

→ template sequence $u_{n-\nu}, \dots, u_{n+\nu}$

$$\phi(X_n, \mathbf{y}) = \exp \sum_{k=-\nu}^{\nu} -\mu_k D(y_{n+k} \| u_{n+k})$$

Exploiting Neighboring Observations

Pitch Feature: Neighborhood Integration



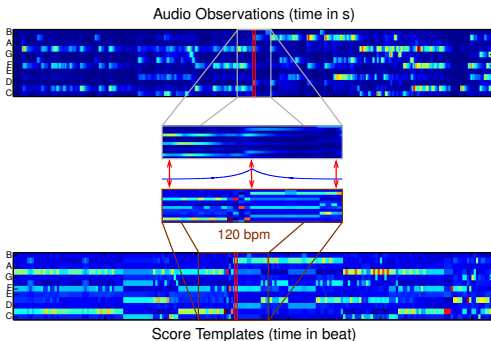
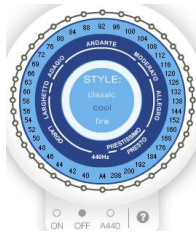
Hypothesis: locally constant tempo T_n (in the label variable X_n)

→ template sequence $u_{n-\nu}, \dots, u_{n+\nu}$

$$\phi(X_n, \mathbf{y}) = \exp \sum_{k=-\nu}^{\nu} -\mu_k D(y_{n+k} \| u_{n+k})$$

Exploiting Neighboring Observations

Pitch Feature: Neighborhood Integration



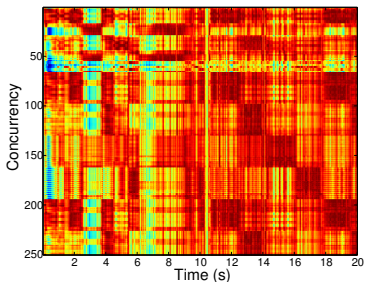
Hypothesis: locally constant tempo T_n (in the label variable X_n)

→ template sequence $u_{n-\nu}, \dots, u_{n+\nu}$

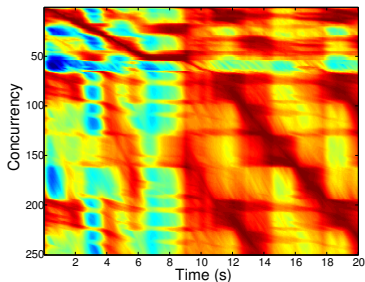
$$\phi(X_n, \mathbf{y}) = \exp \sum_{k=-\nu}^{\nu} -\mu_k D(y_{n+k} \| u_{n+k})$$

Effect on the Similarity Matrix

“Instantaneous” match:



Neighborhood integration:



- “Smoothing” of the similarity matrix
- Enhances paths conforming to score

Using Diverse Sources of Information

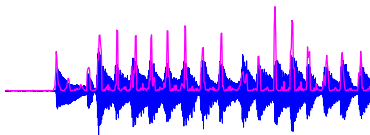
- **Reminder:** observation function can be decomposed into several **features**

$$\phi(\mathbf{X}_n, \mathbf{Y}_{1:N}) = \exp \sum_i \mu_i f_i(\mathbf{X}_n, \mathbf{Y}_{1:N})$$

- Neighborhood integration; exploiting pitch information from **different time positions**
- Also possible to exploit **different descriptors**, characterizing different aspect of the signal
 - Onset detection
 - Tempo

Onset Feature

- Based on spectral flux [Alonso,2005]: $\mathbf{s}_{1:N}$

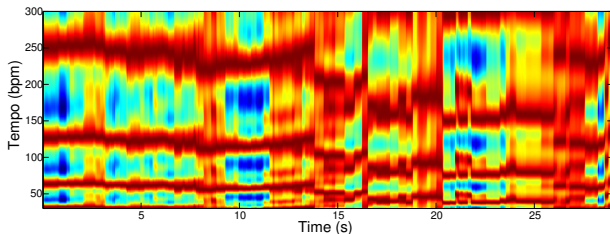


- Characterize the **phase** (attack or sustain) of concurrency

$$f_a(X_n, \mathbf{Y}_{1:N}) = \begin{cases} s_n & \text{if attack} \\ 0 & \text{if sustain} \end{cases}$$

Tempo Feature

- Cyclic tempogram [Grosche,2010]: $\mathbf{g}_{1:N}(t)$



- Characterize the tempo T_n

$$f_t(X_n, \mathbf{Y}_{1:N}) = g_n(T_n)$$

Markovian CRF (MCRF): Alignment Results

- Three types of features:
 - pitch (integrated)
 - onset
 - tempo
- Alignment Rates ($\theta = 300$ ms):

| | MAPS corpus | RWC corpus |
|----------|-------------|------------|
| Baseline | 87.8% | 72.4% |
| MCRF | 94.9% | 87.9% |



- Significant improvement
- Still far from perfect
- Need to exploit other kinds of information on the music

Markovian CRF (MCRF): Alignment Results

- Three types of features:
 - pitch (integrated)
 - onset
 - tempo
- Alignment Rates ($\theta = 300$ ms):

| | MAPS corpus | RWC corpus |
|----------|-------------|------------|
| Baseline | 87.8% | 72.4% |
| MCRF | 94.9% | 87.9% |



- Significant improvement
- Still far from perfect
- Need to exploit other kinds of information on the music
- **Temporal structure**

Outline

- 1 Music-to-Score Alignment: Introduction
- 2 Alignment by Statistical Model
- 3 Conditional Random Fields for Alignment
- 4 Modeling of Time**
 - Introducing Duration Constraints
 - Modeling Tempo Variations
- 5 Optimization of the Concurrency Templates
- 6 Conclusion and Perspectives

Exploiting the Temporal Structure

- Music is highly structured
- Strong priors/dependencies for concurrency durations



Exploiting the Temporal Structure



- Music is highly structured
- Strong priors/dependencies for concurrency durations
- Incorporate temporal constraints into the model
- State of the art in alignment:
 - Hidden Semi-Markov Models [Orio,2002]
 - Hidden Tempo Models [Raphael,2006]

Exploiting the Temporal Structure



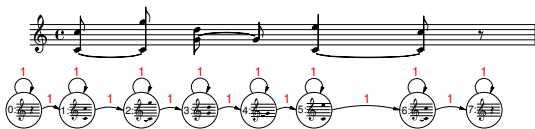
- Music is highly structured
- Strong priors/dependencies for concurrency durations
- Incorporate temporal constraints into the model
- State of the art in alignment:
 - Hidden Semi-Markov Models [Orio,2002]
 - Hidden Tempo Models [Raphael,2006]
- Can be done with CRFs
- Dealt with by the **transition function**

Transition Function

- **Reminder:** probability of a label sequence $\mathbf{X}_{1:N}$, given the observation sequence $\mathbf{Y}_{1:N}$:

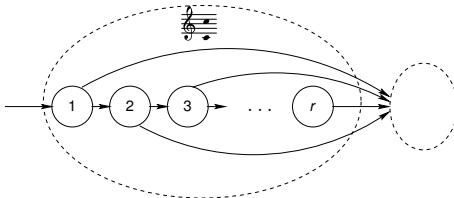
$$P(\mathbf{X}_{1:N} | \mathbf{Y}_{1:N}) = \frac{1}{Z} \phi(\mathbf{X}_1, \mathbf{Y}_{1:N}) \prod_{n=2}^N \psi(X_n, X_{n-1}) \phi(X_n, \mathbf{Y}_{1:N})$$

- $\psi(X_n, X_{n-1})$: potential given to transition between labels
- **MCRF:** no duration constraint \rightarrow uniform transition potentials between concurrencies



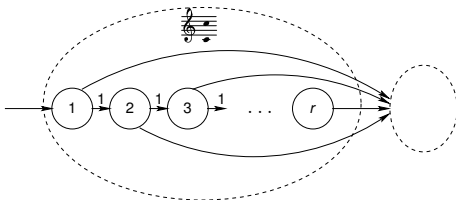
Incorporating Duration Constraints

- Introduction of **occupation variable** D
 - describes the “current duration” of the concurrency



Incorporating Duration Constraints

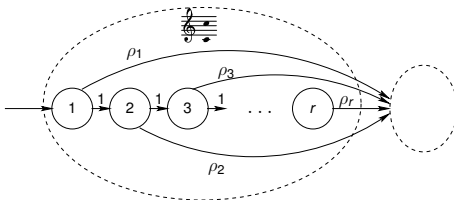
- Introduction of occupation variable D
 - describes the “current duration” of the concurrency



- Transition potentials:
 - Inside concurrency: no penalty

Incorporating Duration Constraints

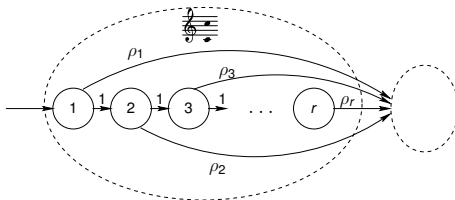
- Introduction of occupation variable D
 - describes the “current duration” of the concurrency



- Transition potentials:
 - Inside concurrency: no penalty
 - Exiting concurrency: ρ_d

Incorporating Duration Constraints

- Introduction of occupation variable D
 - describes the “current duration” of the concurrency



- Transition potentials:
 - Inside concurrency: no penalty
 - Exiting concurrency: ρ_d
- Explicit **duration penalty**

Semi-Markov CRF (SMCRF)

Concurrency Duration Constraint

- Gaussian penalty
- Mean: length ℓ indicated in the score

$$\rho_d = e^{-\gamma_1 |d - \ell|^2}$$

Semi-Markov CRF (SMCRF)

Concurrency Duration Constraint

- Gaussian penalty
- Mean: length ℓ indicated in the score

$$\rho_d = e^{-\gamma_1 |d - \ell|^2}$$

Model Limitation

- Duration constraint is **absolute**
- Does not consider tempo variations

Semi-Markov CRF (SMCRF)

Concurrency Duration Constraint

- Gaussian penalty
- Mean: length ℓ indicated in the score

$$\rho_d = e^{-\gamma_1 |d - \ell|^2}$$

Model Limitation

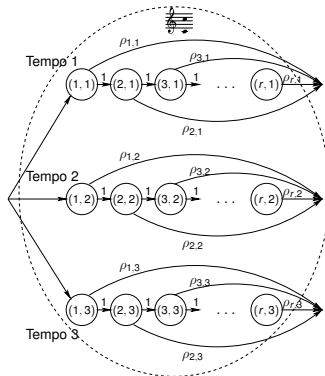
- Duration constraint is absolute
- Does not consider **tempo variations**

Modeling Tempo Variations

Modeling Tempo

- Several tempo possibilities
- Duration penalty depends on tempo hypothesis:

$$\rho_{d,t} = e^{-\gamma_2 \left| \frac{d - \ell(t)}{\ell(t)} \right|^2}$$



Modeling Tempo

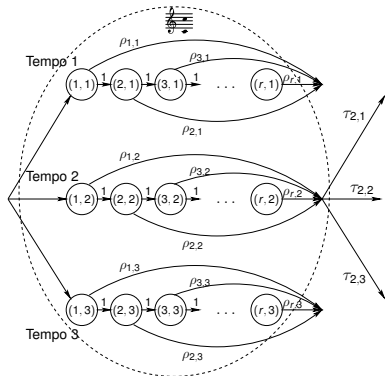
- Several tempo possibilities
- Duration penalty depends on tempo hypothesis:

$$\rho_{d,t} = e^{-\gamma_2 \left| \frac{d - \ell(t)}{\ell(t)} \right|^2}$$

- Tempo variation penalty at concurrency:

$$\tau_{t_1, t_2} = e^{-\gamma_3 \left| \log \frac{t_1}{t_2} \right|^2}$$

- Hidden Tempo CRF (HTCRF) system



Experimental Results

- Alignment Rates ($\theta = 300$ ms):

| | MAPS corpus | RWC corpus |
|----------|-------------|------------|
| Baseline | 87.8% | 72.4% |
| MCRF | 94.9% | 87.9% |
| SMCRF | 97.8% | 93.9% |
| HTCRF | 99.3% | 99.2% |

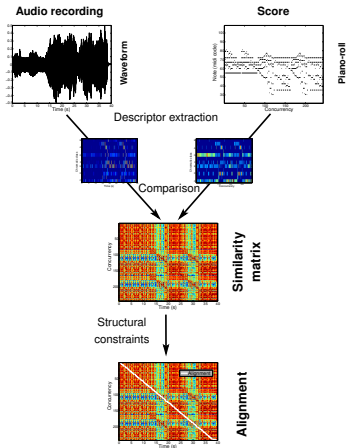


- More complex systems lead to better results
- HTCRF: accurate temporal model → very high precision, even with noisy observation (RWC)

Modeling Tempo Variations

What we have done so far

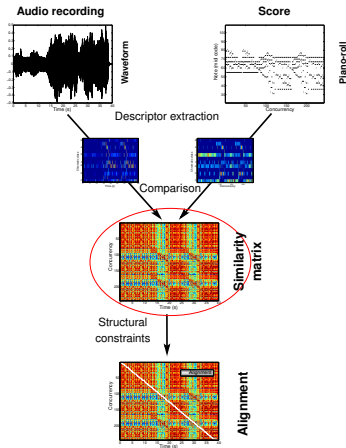
- Enhancement of the similarity matrix
- Exploitation of the temporal structure



Modeling Tempo Variations

What we have done so far

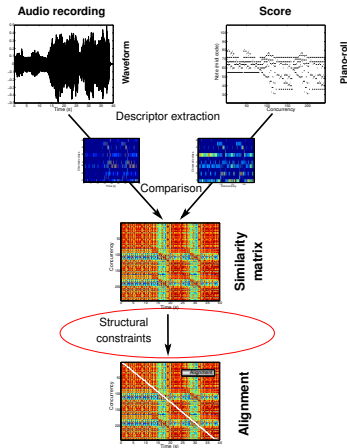
- Enhancement of the similarity matrix
- Exploitation of the temporal structure



Modeling Tempo Variations

What we have done so far

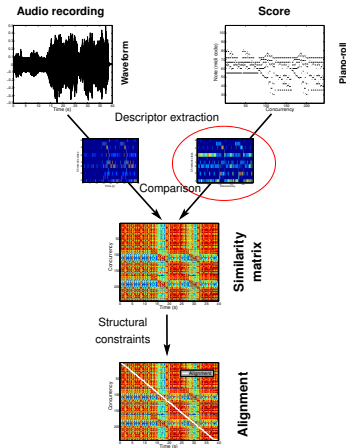
- Enhancement of the similarity matrix
- Exploitation of the temporal structure



Modeling Tempo Variations

What we have done so far

- Enhancement of the similarity matrix
- Exploitation of the temporal structure
- **Template construction?**



Outline

- 1 Music-to-Score Alignment: Introduction
- 2 Alignment by Statistical Model
- 3 Conditional Random Fields for Alignment
- 4 Modeling of Time
- 5 Optimization of the Concurrency Templates**
 - Formalization of the Symbolic to Audio Mapping
 - Learning the Mapping Matrix
 - Alignment Experiments

Template Construction: Reminder

- Templates for isolated notes
- Superposition of one-note templates
- Only few templates must be set



Template Construction: Reminder

- Templates for isolated notes
→ **Set by heuristic**
- Superposition of one-note templates
- Only few templates must be set



Template Construction: Reminder

- Templates for isolated notes
- Superposition of one-note templates
- Only few templates must be set

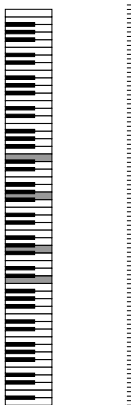


Learn them from data!

Symbolic to Audio Mapping

Pitch Vector Representation

- Vectorial representation of concurrency
- One component per pitch



Pitch Vector Representation

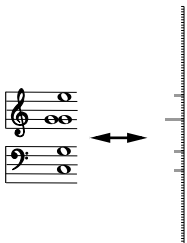
- Vectorial representation of concurrency
- One component per pitch
- Values: number of notes



Symbolic to Audio Mapping

Symbolic-to-Audio Mapping as a Linear Transformation

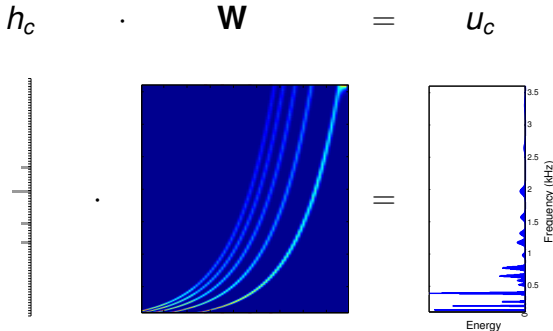
- Concurrency c
- Pitch Vector h_c

 c h_c 

Symbolic to Audio Mapping

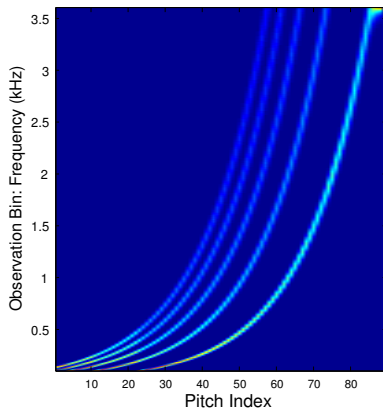
Symbolic-to-Audio Mapping as a Linear Transformation

- Concurrency c
- Pitch Vector h_c
- Mapping Matrix \mathbf{W}
- Template u_c



Mapping Matrix W

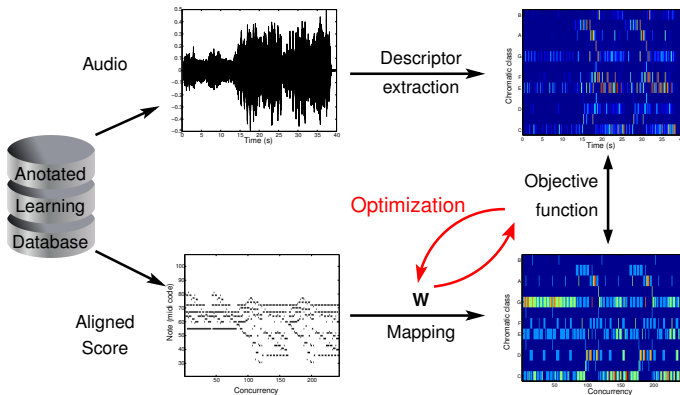
- Contains the one-note templates (in columns)
- Matrix of dimension $I \times J$
 - I : audio representation
 - J : number of pitches
- Example: heuristic matrix for spectrogram



Learning the Mapping Matrix

Learning the Mapping Matrix

- Supervised Learning:



Two Learning Strategies

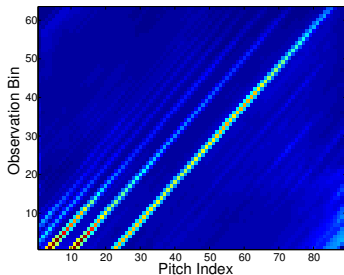
| Method | Minimum Divergence | Maximum Likelihood |
|------------------------------|--------------------|-----------------------|
| Strategy type | best-fit | discriminative |
| Objective function | matching measure | CRF probability |
| Use of structural constrains | no | MCRF (no integration) |
| Optimization problem | convex | non convex |

Learning the Mapping Matrix

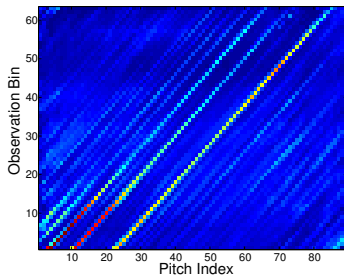
Learned Matrices

Example: Semigram Representation

Minimum Divergence



Maximum Likelihood



- Minimum Divergence: capture the energy distribution of each pitch
- Maximum Likelihood: only learns discriminant information

Experiments

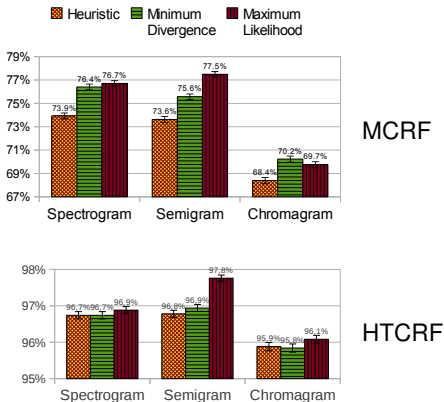
- Application to our alignment models
- No neighborhood integration
- Comparison of learning methods and audio representations

Alignment Experiments

Results

Alignment Rates with $\theta = 100$ ms

- Improved precision
- Influence decrease with accurate temporal model
- Behaviors of learning methods depend on representation
- Winner: semigram with ML learning



Contributions

- Introduction of the CRF framework for audio-to-score alignment
 - allows for flexible features
 - exploits structural constraints
- Optimization of the observation function
 - unified formalization (linear mapping)
 - automatic learning of the mapping matrix
- Miscellaneous adjustments for real-world applications
 - complexity reduction algorithm (hierarchical pruning)
 - musical structure change

Perspectives

- Comprehensive study of the symbolic-to-audio mapping
 - consider neighborhood integration
 - instrument-specific mappings
 - mapping adaptation
 - non-linear mapping
- Considering other observation/transition features
 - superposition of several representations/divergences
 - self-similarity features (change points)
 - multi-modal features (video, motion capture)

The End

Thank you!

Publications

- C. Joder, S. Essid & G. Richard: *A Conditional Random Field Framework for Robust and Scalable Audio-to-Score Matching*. IEEE TASLP, November 2011
- C. Joder, S. Essid & G. Richard: *Optimizing the Mapping from a Symbolic to an Audio Representation for Music-to-Score Alignment*. WASPAA, 2011
- C. Joder, S. Essid & G. Richard: *Hidden Discrete Tempo Model: a Tempo-aware Timing Model for Audio-to-Score Alignment*. ICASSP, 2011
- C. Joder, S. Essid & G. Richard: *A Conditional Random Field Viewpoint of Symbolic Audio-to-Score Matching*. ACM Multimedia, 2010
- C. Joder, S. Essid & G. Richard: *An Improved Hierarchical Approach for Music-to-Symbolic Score Alignment*. ISMIR, 2010
- C. Joder, S. Essid & G. Richard: *A comparative study of tonal acoustic features for a symbolic level music-to-score alignment*. ICASSP, 2010

Other Perspectives

- Refined model structures
 - allow several concurrencies for each score position (reverberation)
 - continuous tempo set
- Other learning or decoding criteria
 - *maximum margin* learning
 - *minimum segmentation error* decoding
- Further complexity reduction
 - particle filtering
- Application to other problems
 - rhythm analysis (beat detection) with HTRF
 - gesture alignment from motion capture. . .