Music Information Retrieval
http://www.ifs.tuwien.ac.at/mir

Andreas Rauber
Department of Software Technology and Interactive Systems
Vienna University of Technology
http://www.isis.tuwien.ac.at

Who am I?
- Vienna University of Technology
  http://www.tuwien.ac.at
  - Faculty of Computer Science
    Department of Software Technology and Interactive Systems
    Software and Information Engineering Group
      - Andreas Rauber
        Machine Learning, Neural Networks
        Text Mining, Digital Libraries
        Music Retrieval
        Digital Preservation

Lead-in

Who else is MIR@ifs?
- Thomas Lidy
- Robert Neumayer
- Rudolf Mayer
- Jakob Frank

Other members
- Veronika Zenz
- Peter Hlavac
- Ewald Peisler
- Andreas Scharf
- Andrei Grecu

Former members
- Markus Fröhlich
- Elias Pamplaga
- Stefan Letich
- David Laister
- Doris Baum

– Lead-in
– Chorus
– Verse 1: Music-IR
– Verse 2: Audio Features
– Verse 3: Classification and Benchmarking
– Verse 4: Clustering & Browsing
– Verse 5: Some other applications
– Fade-out

Activities
- Audio Feature Extraction
- Music Classification
- PlaySOM: Organisation of Music Archives
- PocketSOM: Browsing Music on Mobile Devices
- 3D Worlds for Music
- Audio Segmentation
- Chord Detection
- Blind Source Separation
- Text and Music (Lyrics, Bio, ...)

Music IR – Music?

What is „Music“?
- Music, of course!
  Audio: wav, au, mp3, ...
  Symbolic: MIDI, mod, ...
  Scores: Scan, MusicXML
- Text
  - Song lyrics
  - Artis Biographies
  - Websites:
    - Fanpages,
    - Album Reviews,
    - Genre descriptions
- Community data
  - Playlists
  - Market basket
  - Band evolution
- Video/Images
  - Album covers
  - Music videos
Music - Sound

- Sound as acoustic wave
- Characterized by the properties of waves (frequency/wavelength, amplitude)

- Frequency; pitch
  - Humans can hear approx. 20Hz-20kHz
  - speech: 200Hz-8kHz

- Amplitude: Loudness
  - measured as pressure in micropascal \( \mu Pa \)
  - hearing threshold: approx. \( 20 \mu Pa \)
  - logarithmic decibel scale \( L_p = 10 \log \left( \frac{P}{P_0} \right) = 20 \log \left( \frac{L}{L_0} \right) \) dB

Nyquist sampling theorem: Exact reconstruction of a continuous-time baseband signal from its frequency is greater than twice the signal bandwidth.

Samples must be sampled with twice that frequency for reconstruction.

Different file formats for storing sound:
- lossless formats
  - WAV (may hold compressed audio, but usually lossless PCM)
  - FLAC, Shorten, Monkey’s Audio, ATRAC Advanced Lossless, Apple Lossless, WMA Lossless, TTA
- lossy formats
  - MP3
  - ATRAC
  - AAC
  - Ogg Vorbis
  - WMA
  -...

Digital representation of an analog signal where the magnitude of the signal is sampled regularly at uniform intervals, then quantized to a series of symbols

Used in WAV, CD-recordings, ...

Quantization error: choosing discrete value near the analog signal for each sample

Any frequency above or equal to 1/2 sampling frequency is lost

Music - Sound - Loudness

- hearing threshold: approx. 120 20 hearing damage during short-term effect

- measured as pressure in micropascal \( \mu Pa \)

- logarithmic decibel scale \( L_p = 10 \log \left( \frac{P}{P_0} \right) = 20 \log \left( \frac{L}{L_0} \right) \) dB

Different file formats for storing sound:
- WAV (may hold compressed audio, but usually lossless PCM)
- FLAC, Shorten, Monkey’s Audio, ATRAC Advanced Lossless, Apple Lossless, WMA Lossless, TTA
- lossy formats
  - MP3
  - ATRAC
  - AAC
  - Ogg Vorbis
  - WMA
  -...

- WAV (may hold compressed audio, but usually lossless PCM)
- FLAC, Shorten, Monkey’s Audio, ATRAC Advanced Lossless, Apple Lossless, WMA Lossless, TTA
- lossy formats
  - MP3
  - ATRAC
  - AAC
  - Ogg Vorbis
  - WMA
  -...

Actually: MPEG-1 Audio Layer 3

Developed by a group around Fraunhofer, Thomson, AT&T Bell Labs, several patent issues pending

Lossy compression, based on psycho-acoustic models
- differential encoding of stereo signal (lossless)
- focus on audible frequencies
- masking effects
- adaptive bit-depth encoding
- quantization and huffman-encoding
Music IR – Music?

Music - Sound - MP3
- ID3-Tags
- Added later-on to allow embedding of meta data
- ID3v1: 30 char per entry, few standard fields
- ID3v2.4: UTF-8 support, tags at beginning of file
- Used by search engines

Music IR – Music?

What is „Music“?
- Music, of course!
- Audio: wav, au, mp3, ...
- Symbolic: MIDI, mod, ...
- Text
  - Song lyrics
  - Artis Biographies
  - Websites:
    - Fanpages,
    - Album Reviews,
    - Genre descriptions
- Community data
  - Playlists
  - Market basket
  - Band evolution
- Video/Images
  - Album covers
  - Music videos

Music IR – Music?

Musical Instrument Digital Interface - MIDI
- Symbolic Music File Format
- Dave Smith, proposed in 1981
- MIDI specification 1.0 in 1983
- Interacting with keyboard produces messages
  - Note-On, Aftertouch, and Note-Off
  - 127 note pitches
- Sequence of control commands

Music IR – Music?

MOD
- Similar to MIDI, but
- stores audio samples together with control instructions
- should sound the same on every player
- a.k.a. tracker modules (first ever module creating program was Soundtracker, created by Karsten Obarski 1987)

Music IR – Music?

MOD
- Some examples (from http://modarchive.org)
  - Classical: Dark Castle (Part 1)
  - Classical: Canon in D
  - Classical: Beethoven: Für Elise
  - Guitar: Sweet Lorraine
  - Latin: Heart and Soul
  - Techno: UKBlur
  - Disco: Rob Hubbard

Music IR – Music?
Music IR – Music?

What is „Music”?  
- Music, of course!  
  - Audio: wav, au, mp3, ...  
  - Symbolic: MIDI, mod, ...  
- Scores: Scan, MusicXML

Text  
- Song lyrics  
- Artist Biographies  
- Websites:  
  Fanpages,  
  Album Reviews,  
  Genre descriptions

Community data  
- Playlists  
- Market basket  
- Band evolution

Video/Images  
- Album covers  
- Music videos

Music IR – Music?

Scores  
- Also referred to as „Sheet Music”  
- Hand-written or printed form of musical notation  
  - Handwritten scores  
  - Printed scores  
  - Typeset scores  
  - MusicXML

- Different IR tasks  
  - Scan & Optical Music Recognition (OMR)  
  - Score following  
  - Melodic retrieval

Music IR – Music?

Handwritten scores  
- Different styles of notation  

- Ancient greek:  
  stone at Delphi containing the second of the  
  two hymns to Apollo

- Indian notation  
- China  
  Quin notation

Music IR – Music?

Handwritten / printed scores  
- Different styles of notation  
  - Neumes  
  - Staff

- Complex annotations  
- Scanning scores  
  e.g. Musitek SmartScore:  
  http://www.musitek.com/

- Bach SheetmusicDemo:  
  http://www.samplesmith.com

Music IR – Music?

Music Typesetting / Scorewriter  
- Software used to automate the task of writing and engraving sheet music, aso word processor for text

- Input: UTF-8, no graphical interface  
- Some support Scan+OMR  
- Output: PS/PDF, graphics, MIDI, MusicXML

- Popular programs:  
  - GNU LilyPond Software: http://lylypond.org/  
  - GUIDO Music Notation: http://www.salien.org/GUIDO/  
  - Finale: http://www.finalemusic.com/  
  - Sibelius: http://www.sibelius.com/  
  - Comprehensive list: http://en.wikipedia.org/wiki/Scorewriter

Music IR – Music?

GNU LilyPond Software  
- http://lylypond.org/  
- Input: UTF-8, no graphical interface  
- Some graphical editors produce LilyPond output (e.g. Rosegarden, NoteEdit, Canorus)

- Output: compiled to PDF, SVG, MIDI, ...

- Notes are entered in note, pitch and length format

- Used by several projects (Mutopia, Musipedia)
LilyPond example
%% Theme to "Fire Breathers", a homebrew NES game perpetually
%% under development. Composed by Urpo Lankinen.
%% Note: The composer has made the source code available
%% to Wikipedia under the GFDL license. Other versions outside
%% Wikipedia are typically under CC BY-SA license.
%% This file uses Finnish note names (for example, where
%% Americans use "F#" and "Bb", Finns use "Fis" and "Bb").
%% Dutch note names are used by default.
\include "suomi.ly"
\version "2.6.0"

LilyPond example
%% The header block defines the titles and texts.
\header {
  title = "Theme to "Fire Breathers!"
  instrument = "For the 2A03 or SID"
  composer = "Urpo Lankinen"
  enteredby = "Urpo Lankinen"
  updatedby = "Jan Nieuwenhuizen"
  date = "June 2005"
}

LilyPond example
Melody = \relative c'' {\clef treble \time 3/4 \key a \minor \partial 4 \p a4 | e'4.( d8 [ c]) r8 | d4.( c8 [ h]) r8 | a2. | e2 a4 | e'4.( d8 [ c]) r8 | d4.( e8 [ f]) r8 | e2. | r2 e4 | f4.( e8 [ d]) r8 | d4.( c8 [ h]) r8 | a2. ~ a2 r4 | \bar | "|
}

LilyPond example
%% This is the second voice.
SecondVoice = \relative c {\clef bass \time 3/4 \key a \minor \partial 4 \p r4 | e2. | d2. | a2. | e2 a4 | e'2. | d2 f4 | e2. | r2. f2. | d2. | a2. | e2 a4 | e'2. | d2 h4 | a2. ~ a2 r4 | \bar | "|
}

LilyPond example
Theme to "Fire Breathers"
 Urpo Lankinen

GUIDO
- [http://www.sailer.org/GUIDO/]
- Computer music notation system
- Named after Guido of Arezzo (991/992 – after 1033)
- Designed by Holger Hoos, (TU Darmstadt, now Vancouver, Canada)
- Open format, capable of storing musical, structural, and notational information
Music IR – Music?

GUIDO

MusicXML

MusicIR – Music?

MusicXML

What is „Music“?

MusicIR – Music?

Text: Song lyrics

MusicIR – Music?
Music IR – Music?

**Text:** other

- Plenty of other types of information available
- Artist biographies
- Album reviews
- Fan websites
- Genre description sites
- Instrument description sites

---

Music IR – Music?

- There is more to music than sound and text
- Which genre is this album?

---

Music IR – Music?

- There is more to music than sound and text
- Which genre is this album?

---

Music IR – Music?

- There is more to music than sound and text
- Which genre is this album?

---

Music IR – Music?

- There is more to music than sound and text
- Which genre is this album?
There is more to music than sound and text
Which genre is this album?

Image / Video
• Album covers
• Music videos
• Carefully designed to convey a specific information
Style, Image, Character
• Hardly exploited so far
• Indications, that humans are able to deduce music
genre from album covers.
(Sally Jo Cunningham: „What People Do When They
Look for Music: Implications for Design of a Music

What is Music IR? - Other tasks
• Genre classification
• Mood classification
• Artist identification
• Artist similarity
• Cover song detection
• Rhythm and beat detection
• Score following
• Chord detection
• Audio segmentation
• Instrument detection
• Automatic source separation
• Onset detection
• Optical music recognition
• Melody transcription
• Symbolic music similarity

What is Music IR?
• Searching for Music, of course!
  – Searching for music on the Web
  – Query by Humming
  – Similarity Retrieval
  – Identity detecting (fingerprinting)

• Plenty of other tasks!
**Music IR – Music?**

**Music IR material**
- **Papers**
- **Conferences**
  - ISMIR: International Conference on Music Information Retrieval
  - DAFx: Conference on Digital Audio Effects
  - ICMC: International Computer Music Conference
  - other Multimedia, Information Retrieval, and Digital Library Conferences
- **Journals**
  - ICMJ: International Computer Music Journal
  - JNMR: Journal on New Music Research

---

**Chorus**

- **Lead-in**
- **Verse 1: Music-IR**
- **Chorus – Questions?**
- **Verse 2: Audio Features**
- **Verse 3: Classification and Benchmarking**
- **Verse 4: Clustering & Browsing**
- **Verse 5: Some other applications**
- **Fade-out**

---

**Audio Features**

- A number of features can be calculated
  - MPEG7-Standard Features
  - Marsyas System
  - Rhythm Patterns
  - Rhythm Histograms
  - Statistical Spectrum Descriptors
- Capture different characteristics of sound
- Have different dimensionality
- Perform differently on different task

---

**MPEG7 Features**

- Low-level Descriptors
  - spectral, parametric, and temporal features of a signal
- High-level Description Tools:
  - specific to a set of applications
- general sound recognition and indexing
- instrumental timbre
- spoken content
- audio signature description scheme
- melodic description tools to facilitate query-by-humming

- Details:
  - ISO/IEC JTC1/SC29/WG11N8828; editor: José M. Martinez
  - Palma de Mallorca, Oct. 2004, MPEG-7 Overview (version 10)
Audio Features

MPEG7 Features

- Audio Framework
- Echonest Temporal
- logarithmic
- Band Scales
- Spectral

Marsyas System

- Music Analysis, Retrieval and Synthesis for Audio Signals
- Developed by George Tzanetakis (Univ. of Victoria, CA)
- Implements a range of functions and feature extractors
- Details:
  - http://marsyas.sness.net/
  - http://sourceforge.net/projects/marsyas

Audio Features

Rhythm Patterns

- Amplitude-modulated frequency bands
- First version in 2001 later expanded by psycho-acoustic transformations
- High-dimensional vector (1,440 dimensions)
- Captures regular patterns of activities in the various frequency bands
- Similar to 3d graphic equalizer
  - http://www.tic.tuwien.ac.at/~andi/somejp/prototype2.html

Audio Features

RP - Phase 1

Step 1: FFT

- Fast Fourier Transform
- Window size of 256 samples which corresponds to about 23ms at 11kHz
- Hanning window 50% overlap
- Power spectrum

Step 2: Bark Scale

- Frequencies are bundled into 24 critical-bands (Bark scale)
- Reflect characteristics of the human auditory system, in particular of the cochlea in the inner ear
- Below 500Hz the critical-bands are about 100Hz wide.
- Above 500Hz the width increases rapidly with the frequency
Step 3: Spectral Masking

- Occlusion of a quiet sound by a louder sound when both sounds are present simultaneously and have similar frequencies
  - Simultaneous masking: two sounds active simultaneously
  - Post-masking: a sound closely following it (100-200 ms)
  - Pre-masking: a sound preceding it (usually neglected, only measured during about 20ms)
- Spreading function defining the influence of the j-th critical band on the r-th

Step 4: dB & Phon Transformation

- Transform into decibel
- Relationship between sound pressure level in decibel and hearing sensation is not linear.
- Perceived loudness depends on frequency of the tone
- Equal loudness contours for 3, 20, 40, 60, 80, 100 phon

Step 5: Sone Transformation

- Perceived loudness measured in phon does not increase linearly
- Transformation into Sone
- Up to 40 phon slow increase in perceived loudness, then drastic increase
- Higher sensibility for certain loudness differences

Step R1: Amplitude modulation per critical band

- Loudness of a critical-band usually rises and falls several times.
- Periodical pattern, aka rhythm
- Fourier transform
- 6-second sequences, time quanta of 12ms
- > modulation frequencies in the range from 0 to 43Hz
- A modulation frequency of 43Hz corresponds to almost 2600bpm
- 60 bins per frequency band

Step R2: Fluctuation Strength Model

- Amplitude modulation of the loudness has different effects on our sensation depending on frequency.
- Fluctuation strength around 4Hz
- Roughness at 15-150Hz
- Above 150Hz the sensation of hearing three separately audible tones increases
Step R3: Gradient Filter, Gaussian Smoothing

- Amplitude modulations that occur at several frequency bands with the same frequency are perceived as beat
- Gradient filter to emphasize distinctive beats
- Gaussian smoothing to blur slightly
- Performed for individual 6-second segments
- May be used individually, or median for a whole song

Rhythm Patterns


Statistical Spectrum Descriptors (SSD)

- Start from RP-process at end of stage 1
- SSD: 24*7=168-dimensional vector

Rhythm Histograms (RH)

- Starts from RP-process at end of stage 2:
- RH: 60 dimensions
- Captures rhythmic events

Summary

- A set of different features can be extracted from audio
  - MPEG-7
  - Marsyas System
  - Rhythm Patterns (RP)
  - Rhythm Histograms (RH)
  - Statistical Spectrum Descriptors
- many further are possible

Chorus

- Lead-in
- Verse 1: Music-IR
- Verse 2: Audio Features
- Chorus – Questions?
- Verse 3: Benchmarking: Retrieval and Classification
- Verse 4: Clustering & Browsing
- Verse 5: Some other applications
- Fade-out
We have features computed from audio
- We can use them now for
  - Similarity-based retrieval
  - Classification
  - Clustering

Problem: Benchmark evaluation:
- How do we compare the performance of different feature sets or different algorithms?
- What is the ground truth?

MIREX
- Discussion started at ISMIR 2001
  - evaluation frameworks
  - standardized test collections
  - tasks and evaluation metrics
- IMIRSEL project started 2002:
  (International Music Information Retrieval Systems Evaluation Laboratory), Univ. of Illinois, S. Downie
- First Audio Description Contest at ISMIR 2004
- Start of MIREX in 2005
  (Music Information Retrieval Exchange)
- Annual, in connection with ISMIR conferences
- Evaluating many approaches of the MIR domain

MIREX
- Variations on Benchmarking
  - MIR community suffers from
    - lack of benchmark corpora
      - that are representative
      - that may be shared
    - lack of clear task definitions
    - lack of ground truth annotations
  - Some quasi-benchmark corpora
    - GTZAN
    - ISMIR Rhythm
    - ISMIR Genre
    - RWC database
    - MIREX (closed data)

MIREX
- 2004 Audio Description Contest
  - First attempt towards comparative benchmarking of MIR algorithms
  - Five different tasks
    - Genre Classification
    - Artist Identification
    - Tempo Induction
    - Rhythm Classification
    - Melody Extraction
  - Some training/test data made available
  - Automatic evaluation
  - Test for robustness of algorithms

MIREX
- 2004 Audio Description Contest
  - Broader range of tasks
  - Added symbolic MIR tasks
  - M2K framework for development and rapid evaluation in a common setting
  - No training/test data
  - Algorithm is submitted and evaluated on closed test data
MIREX 2005 tasks
- Audio Artist Identification
- Audio Drum Detection
- Audio Genre Classification
- Audio Melody Extraction
- Audio Onset Detection
- Audio Tempo Extraction
- Audio and Symbolic Key Finding
- Symbolic Genre Classification
- Symbolic Melodic Similarity

M2K framework for developing algorithms
Preferred submission form, others also allowed (Matlab, ...)

MIREX 2006 tasks
- Audio Beat Tracking
- Audio Melody Extraction
- Audio Music Similarity and Retrieval
- Audio Cover Song Identification
- Audio Onset Detection
- Audio Tempo Extraction
- QBSH: Query-by-Singing/Humming
- Score Following
- Symbolic Melodic Similarity

Audio Music Similarity and Retrieval task
- Large scale music similarity evaluation
- 5000 music files, 9 genres
- Task:
  - apply feature extraction for audio similarity
  - compute distance matrix between all 5000 songs
  - Submit distance matrix
- Evaluation
  - human listening tests on similarity
  - objective statistics based on meta-data

Music Similarity Retrieval Task
- Similarity retrieval rather than classification
- Evaluated by human judgements:
  - human listening test
- Evalutron 6000:
  - http://www.music-ir.org/evaluation/eva6000
- Test of statistical significance: Friedman test
- No statistical significance between top-5 teams

Music Similarity Retrieval Task: Human Evaluation
- 60 randomly selected queries
- ~ 20 human evaluators
- 7-8 ranked lists per evaluator
- 3 evaluations per ranked list
- 2 evaluation scales:
  - broad scale: very/somewhat/not similar
  - fine scale: between 0 and 10 (10 = best)
- Statistical significance test
MIREX 2006 Human Evaluation

<table>
<thead>
<tr>
<th></th>
<th>Audio</th>
<th>Symbolic</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of evaluators</td>
<td>24</td>
<td>20</td>
</tr>
<tr>
<td>Number of evaluators per query/candidate pair</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>Number of queries per evaluator</td>
<td>7.5</td>
<td>15</td>
</tr>
<tr>
<td>Size of the candidate lists</td>
<td>30</td>
<td>15</td>
</tr>
<tr>
<td>Number of queries</td>
<td>60</td>
<td>17</td>
</tr>
<tr>
<td>Number of evaluations per evaluator</td>
<td>~210</td>
<td>~225</td>
</tr>
</tbody>
</table>

Music Similarity Retrieval Task: Human Evaluation Results

- 6 participating approaches
- Friedman test on fine scale
- no significant differences between first 5 algorithms

MusicSim: Metadata Statistics

- Retrieval of the top 5, 10, 20 & 50 most similar to each file in the database
- Evaluation of the average % match of same
  - Genre
  - Genre after filtering out the query artist
  - Artist
  - Album title

MusicSim: Metadata Statistics

- Results on the top 20 most similar

MusicSim: Metadata Statistics

- Runtime Comparison
Audio Cover Song Identification
- 30 cover songs of a variety of genres
- 11 versions each (i.e., 330 audio files)
- Embedded in 5000 song collection
- Used a reduced data set of 1000 songs
- Task:
  - 30 cover song queries
  - Return the 10 correct cover songs

Audio Cover Song Identification
- 8 participants:
  - 4 cover song detection algorithms
  - 4 music similarity algorithms
- Evaluation:
  - Total number of covers identified
  - Mean number of covers identified
  - Mean of maxima (average of best-case performance)
  - Mean reciprocal rank of first correctly identified cover (MRR)

Friedman Test on MRR:
DE is significantly better than others
no significant difference between remaining algorithms

Number of identified covers

MIREX 2005 & 2006 Statistics

<table>
<thead>
<tr>
<th></th>
<th>2005</th>
<th>2006</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Tasks</td>
<td>10</td>
<td>13</td>
</tr>
<tr>
<td>Number of Teams</td>
<td>41</td>
<td>46</td>
</tr>
<tr>
<td>Number of Individuals</td>
<td>82</td>
<td>50</td>
</tr>
<tr>
<td>Number of Countries</td>
<td>19</td>
<td>14</td>
</tr>
<tr>
<td>Number of Runs</td>
<td>72</td>
<td>92</td>
</tr>
</tbody>
</table>

MIREX 2007 Tasks
- Audio Artist Identification
- Audio Classical Composer Identification - Audio Artist Identification subtask
- Audio Genre Classification
- Audio Music Mood Classification
- Audio Music Similarity and Retrieval
- Audio Onset Detection
- Audio Cover Song Identification
- Real-time Audio to Score Alignment (a.k.a., Score Following)
- Query by Singing/Humming
- Multiple Fundamental Frequency Estimation & Tracking
- Symbolic Melodic Similarity
MIREX 2007 Timeline

- 15 August: Submission system open
- 24 August: Audio Similarity (AMS) and Symbolic Similarity (SymSim) submissions CLOSED
- 31 August: ALL OTHER SUBMISSIONS DUE
- 5 September: Evalutron 6000 for AMS and SMS goes live
- 12 September: Evalutron 6000 for AMS and SMS closes
- 17 September: All results data back to community via wiki
- 26 September: 1400-1530 MIREX Plenary Panel at ISMIR 2007
- 26 September: 1530-1630 MIREX Poster Session


MIREX 2007 - Genre Classification

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>GH</td>
<td>71.87%</td>
<td>62.89%</td>
</tr>
<tr>
<td>IM_knn</td>
<td>64.83%</td>
<td>54.87%</td>
</tr>
<tr>
<td>IM_svm</td>
<td>76.56%</td>
<td>68.29%</td>
</tr>
<tr>
<td>TL</td>
<td>75.57%</td>
<td>66.71%</td>
</tr>
<tr>
<td>ME</td>
<td>75.03%</td>
<td>66.60%</td>
</tr>
<tr>
<td>ME_spec</td>
<td>73.57%</td>
<td>73.57%</td>
</tr>
<tr>
<td>GT</td>
<td>74.15%</td>
<td>65.34%</td>
</tr>
</tbody>
</table>

MIREX 2007 - Mood Classification

<table>
<thead>
<tr>
<th>Participant</th>
<th>Avg. Raw Class. Acc. (3fold eval.)</th>
</tr>
</thead>
<tbody>
<tr>
<td>IM_knn</td>
<td>67.17%</td>
</tr>
<tr>
<td>IM_svm</td>
<td>55.83%</td>
</tr>
<tr>
<td>CL</td>
<td>60.50%</td>
</tr>
<tr>
<td>KL_1</td>
<td>48.93%</td>
</tr>
<tr>
<td>KL_2</td>
<td>25.67%</td>
</tr>
<tr>
<td>TL</td>
<td>59.67%</td>
</tr>
<tr>
<td>ME</td>
<td>57.83%</td>
</tr>
<tr>
<td>ME_spec</td>
<td>65.83%</td>
</tr>
<tr>
<td>GT</td>
<td>61.50%</td>
</tr>
</tbody>
</table>

Chorus

- Lead-in
- Verse 1: Music-IR
- Verse 2: Audio Features
- Verse 3: Benchmarking: Retrieval and Classification
- Chorus – Questions?
- Verse 4: Clustering & Browsing
- Verse 5: Some other applications
- Fade-out

Clustering & Browsing

- Need new interfaces to access huge music archives
- SOMeJB: SOM-enhanced Jukebox
- Cluster music (by feature sets)
- Based on Self-Organizing Map (SOM)
- Mapping from input- to output space ("2 dim. map")
- Preservation of Neighbourhood relationships
- Map of music space
- PlaySOM and PocketSOM applications
Clustering & Browsing

Self-Organizing Map (SOM)

Clustering & Browsing

Self-Organizing Map (SOM)

Clustering & Browsing

Self-Organizing Map (SOM)

Smoothed Data Histograms (SDH)

Clustering and Browsing

- SOM of Music
- Using audio feature vectors (RP, RH, SSD, RH+SSD, ...)
- Create topology-preserving mapping of music
- Music of similar style in neighboring regions of the map
- Different visualizations
  - plain SOM, class coloring, pie charts
  - we'll use smoothed data histograms (SDH): reveals clusters
- More details:
  - [http://www.ifs.tuwien.ac.at/mir/playsom.html](http://www.ifs.tuwien.ac.at/mir/playsom.html)

Clustering & Browsing

PlaySOM

- Organizing Music
- Creating Playlists
PlaySOM - Annotation

PocketSOM-Player
- Application for mobile devices
- Streaming audio
- Remote control
- http://www.ifs.tuwien.ac.at/mir/pocketsom

PocketSOMPlayer

Web-based Browsing
- Web-based interface
- Reduced functionality

Chorus
- Lead-in
- Verse 1: Music-IR
- Verse 2: Audio Features
- Verse 3: Benchmarking: Retrieval and Classification
- Verse 4: Clustering & Browsing
- Chorus – Questions?
- Verse 5: Some other applications
- Fade-out
Other Applications

- Numerous other applications and tasks
- Some selected examples
  - 3D music worlds
  - Text and audio
  - Audio segmentation
  - Chord detection
  - Automatic source separation

3D Music Worlds

- SOM organizes music by sound similarity
- forms baseline for room set-up
  - real-life & virtual
- Coffee shop, tables, each table plays its music
tables in a zone play similar music
- Get your coffee and choose a table where the music is
to your liking (if there's one free there...)

3D Music Worlds

http://ispaces.ec3.at/muscle.php

Other Applications

- Numerous other applications and tasks
- Some selected examples
  - 3D music worlds
  - Text and audio
  - Audio segmentation
  - Chord detection
  - Automatic source separation

Text and Audio

- Music may be organized into genres
- But: genre is not only based on the acoustic aspects
- Some genres defined by music, some by text,
some by both, some by none?
- Examples:
  - classical music
  - christmas songs
  - hip-hop
  - oldies

Text and Audio

- Parallel corpus, indexed by song lyrics and music
- Clustering on a SOM for analysis
  - Lyrics SOM
  - Music SOM
- Analysis of cluster structure on both
class visualization based on
genre labels
Numerous other applications and tasks

- 3D music worlds
- Text and audio
- Audio segmentation
- Chord detection
- Automatic source separation

Piece of Music has inherent structure:
- lead in
- verse
- chorus
- transition
- fade-out

Goal: to detect these structural components

Application
- optimize feature extraction for respective segments
- find representative elements
- use song structure as feature
Audio Segmentation

- 2-stage procedure
  - Phase 1: segmentation
    - extract features from small sample windows
    - compare neighboring windows
    - find neighbors with drastic differences: segments
  - Phase 2: structure analysis
    - analyze pairs of segment
    - identify segments that are more similar to each other (clusters)
    - derive segment structure

Audio Segmentation

Evaluation
- Based on some benchmark data (qmul14, RWC, ...)
- Problem: no uniform ground truth: different papers, using same data, define different segment boundaries
- Countermeasure: designed complex evaluation scheme, 2-level hierarchical segmentation
- Detailed description, ground truth files and code available at: http://www.it.tuwien.ac.at/mir/audiosegmentation.html

Audio Segmentation

Phase 1 - Example
- Chumbawamba: Thubthumping
  - P=1; R=0.5
- Self-similarity matrix
  - Diagonal white Lines: Repetitions
  - Vertical lines indicate true segment boundaries
- Red asterics mark detected segment boundaries

Audio Segmentation

Phase 2: Song structure
- e.g. ABCDBBC *A
- Feature vectors
  - Spectrogramm, MFCC, Rhythm Patterns, CQT
- Segment boundaries from phase 1
- Clustering
  - means-of-frames
  - „voting“
  - dynamic time warping (dtw)
Phase 2: song structure
- e.g. k-means clustering of song segments
- feature vector is mean of all frames in segment
- need to define "correct" number of clusters, i.e. desired segment types

Evaluation results - Phase 2
- Mean $r_f = 0.707 \pm 0.025$
- With minimal user input: 0.717

Segmentation Evaluation Interface

Chord Detection
- Goal: to detect chords present in polyphonic music
- input: polyphonic audio
- output: sequence of chords and timestamps
- 3 levels of information procedure
  - key detection (chord probabilities)
  - beat detection (chord change position)
  - pitch class profile (chords)
- Detailed description & groundtruth files available at:
  http://www.ifs.tuwien.ac.at/mir/chorddetection.html

Other Applications
- Numerous other applications and tasks
- Some selected examples
  - 3D music worlds
  - Text and audio
  - Audio segmentation
  - Chord detection
  - Automatic source separation

Chord Detection
- Algorithm:

Detailed description & groundtruth files available at:
http://www.ifs.tuwien.ac.at/mir/chorddetection.html
Chord Detection

- Evaluation: limited ground truth available (audio + annotation files)
- What precisely to evaluate, how to weight errors

Other Applications

- Numerous other applications and tasks
- Some selected examples
  - 3D music worlds
  - Text and audio
  - Audio segmentation
  - Chord detection
  - Automatic source separation

Source Separation

Principles

- A piece of music consists of mixture of digitized sound waves from several instruments
- Goal: undo mixture
  - calculate signals for the individual instruments
- Goal may be reached at different levels
  - High-Level – sufficient, if sound texture is correct, and notes are correct important for template matching approaches
  - Low-Level – digitized sound wave signal has to correspond as well as possible with original sound wave of instrument used in blind source separation approaches
- Starting point: stereo audio recordings
- Many recording not truly stereo, all kind of mixing artefacts

Template Matching

- Principles:
  - Music has a certain structure
  - Transitions between notes follow a specific timing
  - Tones repeat during a piece of music
  - Each tone lasts only for a short period of time (not really true for e.g. string instruments)
  - Use this structure to detect patterns: templates
- Structure may be represented via repeating templates
- Templates represent wave form of the notes of individual instruments
- Piece of music may be re-synthesized using these templates similar to MOD-files
- Task: iteratively, find templates
  - identify mixture parameters: time, loudness
  - learn wave form of note
Template-Matching - Results
- works well with highly repetitive notes
  - drums
  - Techno music
- works badly with string instruments
- doesn’t work with voice, or rarely occurring tones
- Residual error may be useful (voice)
- Doesn’t really work well in real-life settings
  - depends on initialization
  - templates starting in the middle of a tone
  - instruments that play at the same time are hard to separate

Blind Source Separation
- Instruments have specific:
  - Position in the room
  - Specific time shift between channels
    - results in phase shift in the spectrum
  - Specific loudness shift between channels
    - results in magnitude shift in spectrum
- Frequency spectrum
  - Piccolo
  - Base

Artificially mixed pieces of music
- Advantages
  - Mixture parameters are known
  - Result of separation may be compared with original
- Disadvantages
  - Mixed without echo, instruments do not diffuse in 2D-histogram
  - Even base drums show loudness differences
  - Unrealistic, but nice for testing
Source Separation

Blind source Separation - results
- Works basically pretty well
- Difficult to separate more than 3 instruments in real recordings
- Potential improvements
  - Utilize repetitions (combine with template matching)
  - Utilize properties of harmonic instruments
    - Base frequency and harmonics

Source Separation

Some examples
- Blind Source Separation
  - Original: Georghe Zamfir: Sirba
  - Instruments:
    - Instrument 1: Hapsicord, Contrabass
    - Instrument 2: Panflute
    - Instrument 3: Catch-all

Chorus

- Lead-in
- Verse 1: Music-IR
- Verse 2: Web Search for Music
- Verse 3: Audio Features
- Verse 4: Benchmarking: Retrieval and Classification
- Verse 5: Clustering & Browsing
- Verse 6: Some other applications
- Chorus – Questions?
- Fade-out

Fade-out

You have learned a lot about Music IR
- Different types of music representation
- Different types of musical information
- Features we can compute from audio
- State of the art in retrieval, classification
- Evaluation and benchmarking challenges
- Applications for browsing music collections
- Challenging application scenarios

Fade-out

But
- There is a lot more to learn...
- ...and a lot of open problems to solve!
- Music IR is a very young discipline
- Many surprises, unknown territory waiting to be explored
- I hope this presentation has
  - Given you some interesting and new information
  - Inspired you to pick up challenging research questions in this field

http://www.ifs.tuwien.ac.at/mir
Thank You!