音声音響行動情報処理特論
～音楽情報処理～

Yasunori Ohishi / 大石康智

NTT Communication Science Laboratories
Media Information Laboratory / メディア情報研究部
Media Recognition Group / メディア認識研究G
大石康智 / Yasunori Ohishi

- 2009年3月 名古屋大学大学院情報科学研究科武田研 卒業
  - 学位論文: 多様な歌唱様式を予測・説明する歌声音響信号の分析合成モデルとその応用に関する研究
- 2009年4月 NTT コミュニケーション科学基礎研究所 入社
  - 歌声・話し声の音高(F0)軌跡のモデリングとその応用
  - 音声/音楽に限定しない、あらゆる音(音響イベント)の検出・識別
  - 多言語音声分類・識別
  - 機械学習理論

NTT厚木研究開発センター
Organizational Chart

Board of Corporate Auditors
- Corporate Auditors
- Auditor’s Office

Board of Directors
- Chairman
- President

Corporate Strategy Planning Department
Technology Planning Department
Research and Development Planning Department
Finance and Accounting Department
General Affairs Department
Strategic Business Development Division
Service Innovation Laboratory Group
Information Network Laboratory Group
Science and Core Technology Laboratory Group
Intellectual Property Center

Service Evolution Laboratories
Media Intelligence Laboratories
Software Innovation Center
Secure Platform Laboratories

Network Technology Laboratories
Network Service Systems Laboratories
Access Network Service Systems Laboratories
Energy and Environment Systems Laboratories

Network Innovation Laboratories
Device Innovation Center
Device Technology Laboratories
Communication Science Laboratories
Basic Research Laboratories
Organizational Chart

Service Innovation Laboratory Group
- Service Evolution Laboratories
- Media Intelligence Laboratories
- Software Innovation Center
- Secure Platform Laboratories

Information Network Laboratory Group
- Network Technology Laboratories
- Network Service Systems Laboratories
- Access Network Service Systems Laboratories
- Energy and Environment Systems Laboratories

Science and Core Technology Laboratory Group
- Network Innovation Laboratories
- Device Innovation Center
- Device Technology Laboratories
- Communication Science Laboratories
- Basic Research Laboratories

Research and development leading to the creation of new technologies and products for new broadband and ubiquitous services

Research and development of infrastructure technology to support next-generation network services

Cutting-edge research and development leading to the creation of new far-sighted principles and concepts
NTT Communication Science Labs.

Media Information Laboratory
- Recognition Research Group (厚木)
  - Signal Processing Research Group (京阪奈)
  - Computing Theory Research Group (厚木)

Innovative Communication Laboratory
- Learning and Intelligent Systems Research Group (京阪奈)
- Linguistic Intelligence Research Group (京阪奈)
- Communication Environment Research Group (京阪奈)

Human Information Science Laboratory
- Sensory Resonance Research Group (厚木)
- Sensory Representation Research Group (厚木)
- Sensory and Motor Research Group (厚木)

Moriya Research Laboratory (厚木)
- Speech/audio signal processing and encoding
Recognition Research Group

- Media search
  - Detecting and locating media fragments that are "similar" to a fragment of audio, video or image given as a query, on a huge amount of unlabeled audio, video or image archives

- Speech and audio, music signal modeling
  - Constructing a new sparse representation model of music and a statistical model of speech generating process

- High fidelity color reproduction and analysis
  - Developing a technology for accurate color measurement and reproduction using multi-band images

- Fun and intuitive programming
  - Providing a intuitive programming language “Viscuit”
ロバストメディア探索技術（RMS）

- 登録されたメディアコンテンツ（音・映像）を瞬時に検出・特定する
- キーワードに依存することなく、音・映像そのものに基づいてコンテンツを特定する。これによりコンテンツの関連情報も引き出せる

（適用例）放送局向け 音楽著作権管理
従来、放送で用いられた音楽の著作権管理は繁雑で困難な作業であったため、サンプリング調査を行っていた
本技術により、実際の放送音に基づく曲名・使用時間の全リスト化が自動で行えるようになったため、実用実態の全数把握と、それに基づく権利処理が可能になった

テレビ・ラジオの放送

BGM
効果音
挿入歌

放送音と常時照合

流通楽曲のデータベース（随時更新）

検出！

使用音楽リスト

「コンテンツの関連情報」＝曲名など
ロバストメディア探索の原理と特長

- 音や映像から抽出した特徴データどうしを極めて高速に照合することにより、コンテンツの存在の有無や、（あるとすれば）その対応する時間区間どうしを正確に特定する

- 特徴抽出と照合における工夫により、編集などによる信号の様々な変化にも影響を受けることなく、超高速の照合・検出が可能

音楽・映像の断片データ

特徴抽出

音楽

映像

数百万タイトル以上のメディアデータに対応

結果

タイトルや使用区間などの関連情報

発見！

特徴データ

1

2

3

4

RMSエンジン
スマホで探索

身の回りの音・映像で情報検索
Multimodal Music Processing

- Edited by Meinard Müller, Masataka Goto, Markus Schedl
Multimodal Music Processing

- Edited by Meinard Müller, Masataka Goto, Markus Schedl
- Contents
  - Linking Sheet Music and Audio
  - *Lyrics-to-Audio Alignment* and its Application
  - Fusion of Multimodal Information in Music Content Analysis
  - A Cross-Version Approach for *Harmonic Analysis* of Music Recordings
  - Score-Informed *Source Separation* for Music Signals
  - Music Information Retrieval Meets *Music Education*
  - Human Computer *Music Performance*
  - User-Aware Music Retrieval
  - **Audio Content-Based Music Retrieval**
  - Data-Driven Sound Track *Generation*
  - Music Information Retrieval: An Inspirational Guide to Transfer from Related Disciplines
Introduction

- Large collections containing millions of digital music documents are accessible from anywhere around the world.
  - Online music stores
  - On-demand streaming music services

Such a tremendous amount of readily available music requires retrieval strategies that allow users to explore large music collections in a convenient and enjoyable way.
Most audio search engines

- Metadata and textual annotations of the actual audio content
  - Descriptions of the artist, title, or other release information

![Google search results for Beethoven](attachment:image.png)

Traditional retrieval using textual metadata (e.g., artist, title) and a web search engine

Typical query terms may be a title such as “Act naturally” when searching the song by The Beatles or a composer’s name such as “Beethoven”.

Drawback 😞
User needs to have a relatively clear idea of what he or she is looking for.
General and expressive annotations

- So called *tags* of the actual musical content
  - Descriptions of the musical style, genre of a recording, the mood, the musical key, or the tempo
  - Form the basis for music recommendation systems that makes the audio content accessible even when users are not looking for a specific song or artist but for music that exhibits certain musical properties

**Drawbacks**

- The generation of such annotations is typically a labor intensive and time-consuming process
- Musical expert knowledge is required for creating reliable, consistent, and musically meaningful annotations
Crowd (or social) tagging

- Voting and filtering strategies based on large social networks of users for “cleaning” the tags
  - Tags assigned by many users are considered more reliable than tags assigned by only a few users

Last.fm *tag cloud* for “Beethoven”

Font size reflects the frequency of the individual tags

Drawback 😞

- Relies on a large crowd of users for creating reliable annotations
- While mainstream pop/rock music typically covered by such annotations, less popular genres are often scarcely tagged (“Long-tail” problem).
Audio content-based retrieval strategies

- Great potential as they do not rely on any manually created metadata but are exclusively based on the audio content and cover the entire audio material
  - One possible approach is to employ automated procedures for tagging music, such as automatic genre recognition, mood recognition

Drawback 😞
- Requirement of large corpora of tagged music examples as training material
- The quality of the tags generated by state-of-the-art procedures does not reach the quality of human generated tags
Audio content-based retrieval strategies

- Given an audio recording or a fragment of it, the task is to automatically retrieve documents from a given music collection containing parts or aspects that are similar to it.

The notion of similarity used to compare different audio recordings (or fragments) is of crucial importance.

Retrieval systems do not require any textual descriptions!
Specificity and granularity

- Two aspects for characterizing retrieval systems
- Specificity
  - Degree of similarity between the query and the database documents

High-specific retrieval systems return exact copies of the query (they identify the query or occurrences of the query within database documents)

Low-specific retrieval systems return vague matches that are similar with respect to some musical properties
Specificity
high
low
Granularity
fragment
document

**Audio Identification**
- Plagiarism detection
- Copyright monitoring
- Audio fingerprinting

**Audio Matching**
- Variation / motif discovery
- Musical quotations discovery

**Version Identification**
- Cover song detection

**Remix / Remaster retrieval**

**Category-based Retrieval**
- Music / speech segmentation
- Year / epoch discovery
- Key / mode discovery
- Loudness-based retrieval
- Tag / metadata inference
- Mood classification
- Genre / style similarity
- Recommendation
- Instrument-based retrieval

**Recommendation**
- Instrument-based retrieval
- Version Identification
- Audio Matching
- Audio Identification
- Remix / Remaster retrieval
Specificity and granularity

- **Granularity (temporal scope)**
- **Fragment-level retrieval:**
  - Query consists of a short fragment of an audio recording, and the goal is to retrieval all musically related fragment
  - A few seconds of audio content, a motif, a theme, or a musical part of a recording
- **Document-level retrieval:**
  - Query reflects characteristics of an entire document and is compared with entire documents of the database
  - Notion of similarity typically is rather coarse and the used features capture global statistics of an entire recording
**Specificity**

- high
- low

**Granularity**

- document
- fragment

**Audio Identification**
- Plagiarism detection
- Copyright monitoring
- Audio fingerprinting

**Version Identification**
- Cover song detection

**Audio Matching**
- Variation / motif discovery
- Musical quotations discovery

**Category-based Retrieval**
- Music / speech segmentation
- Year / epoch discovery
- Key / mode discovery
- Loudness-based retrieval
- Tag / metadata inference
- Mood classification
- Genre / style similarity
- Recommendation
- Instrument-based retrieval

**Remix / Remaster retrieval**

**Recommendation**
- Instrument-based retrieval

**Instrument**
Specificity and granularity

- Four different groups of retrieval scenarios corresponding to the four clouds
  - Audio identification
  - Audio matching
  - Version identification
  - Category-based retrieval

- Not strictly separated but blend into each other
- Intuitive overview of the various retrieval paradigms while illustrating their subtle but crucial differences
Audio identification (audio fingerprinting)

- High-specific fragment-level retrieval task
- Given a small audio fragment as query, the task consists in identifying the particular audio recording
- Widely used in commercial systems such as Shazam
- Query is exposed to signal distortions on the transmission channel
  - Noise, MP3 compression artifacts, uniform temporal distortions, or interferences of multiple signals

Recent identification algorithms exhibit a high degree of robustness
Audio identification (audio fingerprinting)

- Notion of similarity is very close to the identity
- Distinguish between a piece of music and a specific performance of this piece
  - There exist a large number of different recordings of the same piece of music performed by different musicians

Query fragment from a Bernstein recording of Beethoven’s Symphony No.5

Query fragment from a live performance of “Act naturally” by The Beatles

Karajan recording of Beethoven’s Symphony No.5

Original studio recording of this song

Not designed to deal with strong non-linear temporal distortions or with other musically motivated variations that affect the tempo or the instrumentation etc.
Audio matching

- Lower specificity level and fragment level
  - Allows semantically motivated variations as they occur in different performances and arrangements of a piece of music

Query fragment from a Bernstein recording of Beethoven’s Symphony No.5

Karajan recording of Beethoven’s Symphony No.5

Query fragment from a live performance of “Act naturally” by The Beetles

Original studio recording of this song

- Include non-linear global and local differences in tempo, articulation, and phrasing as well as differences in executing note groups such as grace notes, trills, or arpeggios
- Deal with considerable dynamical and spectral variations, which result from differences in instrumentation and loudness
Version identification

- Document-level retrieval at a similar specificity level as audio matching
  - Identify different versions of the same piece of music within a database
  - Not only deals with changes in instrumentation, tempo, and tonality, but also with more extreme variations concerning the musical structure, key, or melody as occurring in remixes and cover songs

- Document-level similarity measures to globally compare entire documents
Category-based retrieval

- Even less specific document-level retrieval tasks
- Encompasses retrieval of documents whose relationship can be described by cultural or musicological categories
  - Genre, rhythm styles, or mood and emotions
In the following

- Elaborate aspects of specificity and granularity by means of representative state-of-the-art content-based retrieval methods
- Highlight characteristics and differences in requirements when designing and implementing systems for
  - High-specific audio identification
  - Mid-specific audio matching
  - Version identification
- Address efficiency and scalability issues
- Discuss open problems in the field of content-based retrieval and give an outlook on future directions
Audio Identification

- Audio material is compared by means of \textit{audio fingerprints}, which are compact content-based signatures of the recordings.

## Requirements for fingerprints

(1) Robustness against distortions (Capture highly specific characteristics so that a short audio fragment suffices to identify the corresponding recording and distinguish it from millions of other songs)

- Noise
- Artifacts from lossy audio compression
- Pitch shifting, time scaling
- Equalization
- Dynamic compression
Audio Identification

- Requirements for fingerprints
  
  (2) Scalability (Capture the entire digital music catalog, which is further growing every day)

  Million  \[\rightarrow\]  Billion

  (3) Compact and efficiently computable (Minimize storage requirements and transmission delays)

Audio signal \[\rightarrow\]  Fingerprints

Few bytes

- Crucial for the design of large-scale audio identification systems
- Face a trade-off between contradicting principles
Ways to design and compute fingerprints

- Short sequences of frame-based feature vectors
  - Mel-frequency Cepstral Coefficients (MFCCs), Bark-scale spectrogram, or a set of low-level descriptors
  - Vector quantization or thresholding techniques, or temporal statistics are needed for obtaining the required robustness
Short-time Fourier transform (STFT)

Audio signal

Frame length: 25 msec.
Frame shift length: 10 msec.

Discrete Fourier transform

Magnitude spectrogram
Mel-Frequency Cepstral Coefficients (MFCCs)

Magnitude spectrogram

Filter-bank outputs
Logarithmic value
Discrete cosine transform
MFCC
Vector quantization (VQ)

Training data

```
[ ] [ ] [ ]  ...  [ ]
```

e.g. k-means algorithm

Codebook

```
(1) (2) (3) (4) (5)  (256)
```

Query

```
MFCC
```

VQ

```
3 1 4 2 2 4  ...  8
```

Identification

```
Song A 1 1 3 2 2  ...  4
Song B 3 5 4 4 4  ...  2
Song C 2 2 1 5 3  ...  1
```

Database
Ways to design and compute fingerprints

- Short sequences of frame-based feature vectors
  - Mel-frequency Cepstral Coefficients (MFCCs), Bark-scale spectrogram, or a set of low-level descriptors
  - Vector quantization or thresholding techniques, or temporal statistics are needed for obtaining the required robustness

- A sparse set of characteristic points
  - Spectral peaks or characteristics wavelet coefficients

Now describe the peak-based fingerprints suggested by Wang, which are now commercially used in the Shazam music identification service (www.shazam.com)
Shazam audio identification system

- Provides a smartphone application that allows users to record a short audio fragment of an unknown song using the built-in microphone.
- Derives the audio fingerprints which are sent to a server that performs the database look-up.
- Returns the retrieval result and presents to the user together with additional information about the identified song.
Demonstration

- Just give it a try!
Peak-picking

- Compute a spectrogram using a STFT
- Apply a peak-picking strategy that extracts local maxima in the magnitude spectrogram: time-frequency points that are locally predominant
Basic retrieval concept of the Shazam

- Spectrograms for an example database document (30 seconds) and a query fragment (10 seconds)
  - The extracted peaks are superimposed to the spectrograms
Basic retrieval concept of the Shazam

- Reduce the complex spectrogram to a “constellation map”, a low-dimensional sparse representation of the original signal by means of a small set of time-frequency points.
- Peaks are highly characteristics, reproducible, and robust against many, even significant distortions of the signal.

Peak is only defined by its time and frequency values, whereas magnitude values are no longer considered.
General database look-up strategy

- Given the constellation maps for a query fragment and all database documents, locally compare the query fragment to all database fragments of the same size
- Count matching peaks, i.e., peaks that occur in both constellation maps
General database look-up strategy

- Both constellation maps show a high consistency (many red and blue points coincide) at a fragment of the database document starting at time position 10 seconds.

Not all query and database peaks coincide. This is because the query was exposed to signal distortions on the transmission channel (white noise etc.)
Exhaustive search strategy!

- Not feasible for a large database as the run-time linearly depends on the number and sizes of the documents

- Reduce the retrieval time using indexing techniques – very fast operations with a sub-linear run-time

- Directly using the peaks as hash values is not possible as the temporal component is not translation-invariant and the frequency component clone does not have the required specificity
Consider pairs of peaks

- Fixes a peak to serve as “anchor peak” and then assigns a “target zone”
- Pairs are formed of the anchor and each peak in the target zone, and a hash value is obtained for each pair of peaks as a combination of both frequency values and the time difference between the peaks

Using every peak as anchor peak, the number of items to be indexed increases
Combinatorial hashing strategy

Query

$h(f_1, f_2, \Delta t_1) : t_1$

Hash value  Time of anchor point

Database

$h(f_1, f_2, \Delta t_1)$  $h(f_2, f_5, \Delta t_4)$  $h(f_5, f_6, \Delta t_3)$  $h(f_{10}, f_2, \Delta t_2)$

Identify!
Combinatorial hashing strategy

- Three advantages

  (1) The resulting fingerprints show a higher specificity than single peaks, leading to an acceleration of the retrieval as fewer exact hits are found

\[ h(f_1, f_2, \Delta t_1) : t_1 \]

  (2) The fingerprints are translation-invariant as no absolute timing information is captured

\[ h(f_1, f_2, \Delta t_1) : t_1 \]

  (3) The combinatorial multiplication of the number of fingerprints introduced by considering pairs of peaks as well as the local nature of the peak pairs increases the robustness to signal degradations
Shazam audio identification system

- Facilitates a high identification rate, while scaling to large databases

- One weakness of this algorithm is that it can not handle time scale modifications of the audio as frequently occurring in the context of broadcasting monitoring
  - Time scale modifications (also leading to frequency shifts) of the query fragment completely change the hash values

- Extensions of the original algorithms dealing with this issue
Audio Matching

- Less specific retrieval tasks are still mostly unsolved
- Highlight the difference between high-specific audio identification and mid-specific audio matching
  - Introduce chroma-based audio features
  - Sketch distance measures that can deal with local tempo distortions
  - Indicate how the matching procedure may be extended using indexing methods to scale to large datasets
Suitable descriptors

- To capture characteristics of the underlying piece of music, while being invariant to properties of a particular recordings

- Chroma-based audio features, (pitch class profiles)
  - A well-established tool for analyzing Western tonal music
  - Assuming the equal-tempered scale, the chroma attributes correspond to the \{C, C#, D, D#, E, F, F#, G, G#, A, A#, B\} that consists of the twelve pitch spelling attributes as used in Western music notation

- Capturing energy distributions in the twelve pitch classes, chroma-based audio features closely correlate to the harmonic progression of the underlying piece of music
Computing chroma features

- Decomposition of an audio signal into a chroma representation
  - Method 1: Using short-time Fourier transforms in combination with binning strategies
  - Method 2: Employing suitable multi-rate filter banks
- Computation of chroma features for a recording of the first five measures of Beethoven’s Symphony No. 5 in a Bernstein interpretation
Computing chroma features

- Fine-grained (and highly specific) signal representation is coarsened in a musically meaningful way
- Adapts the frequency axis to represent the semitones of the equal tempered scale

- Significantly more robust against spectral distortions than the original spectrogram
Computing chroma features

- Pitches differing by octaves are summed up to yield a single value for each pitch class

- Increased robustness against changes in timbre, as typically resulting from different instrumentations
Using suitable post-processing steps

- To increase the degree of robustness of the chroma features against musically motivated variations
  - Normalizing the chroma vectors makes the features invariant to changes in loudness or dynamics
  - Applying a temporal smoothing may significantly increase robustness against local temporal variations that occur as a result of local tempo changes or differences in phrasing and articulation
More variants of chroma features

- Applying logarithmic compression or whitening procedures enhances small perceptually relevant spectral components

- Peak picking of spectrum’s local maxima can enhance harmonics while suppressing noise-like components

- Generalized chroma representations with 24 or 36 bins (instead of the usual 12 bins) allow for dealing with differences in tuning
Implementations

- Chroma Toolbox (MATLAB):
  http://resources.mpi-inf.mpg.de/MIR/chromatoolbox/

- MIR Toolbox (MATLAB):

- Python in MIR
  https://github.com/bmcfee/librosa
Spectrograms and chroma features

- Two different interpretations (by Bernstein and Karajan) of Beethoven’s Symphony No.5
  - Chroma features exhibit a much higher similarity than the spectrograms, revealing the increased robustness against musical variations
  - Fine-grained spectrograms reveal characteristics of the individual interpretations

Bernstein recording

Karajan recording
Spectrograms and chroma features

- **Fingerprint peaks**
  - Spectrogram peaks are very inconsistent for the different interpretations
  - Chromagram peaks show at least some consistencies, indicating that fingerprinting techniques could be applicable for audio matching

**Bernstein recording**

**Karajan recording**

Fragile peak picking step on the basis of the rather coarse chroma features may not lead to robust results
Subsequence search

- A query chromagram is compared with all subsequences of database chromagrams
  - Obtains a matching curve, where a small value indicates that the subsequence of the database starting at this position is similar to the query sequence
  - The best match is the minimum of the matching curve
Subsequence search

- Audio matching procedure for the beginning of Beethoven’s Symphony No. 5 using a query fragment corresponding to the first 22 seconds of a Bernstein interpretation and a database consisting of an entire recording of a Karajan interpretation.

- Strict subsequence matching

- DTW-based matching

- Multiple query scaling strategy
To speed up exhaustive matching procedure

- Require methods that allow for efficiently detecting *near* neighbors rather than exact matches
  - Method 1: Inverted file indexing and depends on a suitable codebook consisting of a finite set of characteristic chroma vectors

Training data

```
[ ] [ ] [ ] ... [ ]
```

e.g. k-means algorithm

Codebook

```
(1) (2) (3) (4) (5)
(256)
```

Query

```
Chroma
```

```
VQ
```

Identification

```
[3 1 4 2 2 4 ... 8]
```

Database

```
Song A 1 1 3 2 2 ... 4
Song B 3 5 4 4 4 ... 2
Song C 2 2 1 5 3 ... 1
```
To speed up exhaustive matching procedure

- **Method 1: Inverted file indexing**
  - The performance of the exact search using quantized chroma vectors greatly depends on the codebook
  - Require fault-tolerance mechanisms which partly eliminate the speed-up obtained by this method
  - This approach is only applicable for databases of medium size
Alternative approach

- Method 2: Using an index-based near neighbor strategy based on **locality sensitive hashing (LSH)**
  - Audio material is split up into small overlapping *shingles* that consist of short chroma feature subsequences
  - Shingles are indexed using locality sensitive hashing

- To cope with temporal variations, each shingle covers only a small portion of audio material and queries need to consist of a large number of shingles
- The high number of table look-ups induced by this strategy become problematic for very large datasets
Summary of audio matching

- Mid-specific audio matching using a combination of highly robust chroma features and sequence-based similarity measures that account for different tempo results in a good retrieval quality

Query fragment from a Bernstein recording of Beethoven’s Symphony No.5

Karajan recording of Beethoven’s Symphony No.5

- Low specificity of this task makes indexing much harder than in the case of audio identification
- This task becomes even more challenging when dealing with relatively short fragments on the query and database side
Version identification

- The degree of specificity
  - Very high for audio identification
  - More relaxed for audio matching
  - Even less specificity: version identification

- A version may differ from the original recording in many ways
  - Significant changes in timbre, instrumentation, tempo, main tonality, harmony, melody, and lyrics

Karajan’s rendition of Beethoven’s Symphony No.5

- One could be also interested in a live performance of it, played by a punk-metal band who changes the tempo in a non-uniform way, transposes the piece to another key, and skips many notes as well as most parts of the original structure

Despite numerous and important variations, one can still unequivocally glimpse the original composition
Version Identification

- Interpreted as a document-level retrieval task, where a single similarity measure is considered to globally compare entire documents

- However, successful methods perform this global comparison on a local basis
  - The final similarity measure is inferred from locally comparing only parts of the documents
  - Comparisons are performed either some representative part of the piece, on short, randomly chosen subsequences of it, or on the best possible longest matching subsequence
A common approach

- Starts from the previously introduced chroma features
- Also more general representations of the tonal content such as chords or tonal templates have been used
- Melody-based approaches have been suggested, although recent findings suggest that this representation may be suboptimal
Tempo and timing deviations

- Have a strong effect in the chroma feature sequences, hence making their direct pairwise comparison problematic
- An intuitive way to deal with global tempo variations is to use beat-synchronous chroma representation
  - The required beat tracking step is often error-prone for certain types of music and therefore may negatively affect the final retrieval result

As for the audio matching task, dynamic programming algorithms are a standard choice for dealing with tempo variations
Alignment procedure

- “Act naturally” example by The Beatles
  - The chroma features of this version (c)

- This song is originally not written by The Beatles but a cover version of a Buck Owens song of the same name
  - The chroma features of the original version (a)

The Beatles  Buck Owens
Alignment algorithms

- Rely on some sort of scores for matching individual chroma sequence elements
  - Scores can be real-valued or binary
  - Fig. (b) shows a binary score matrix encoding pair-wise similarities between chroma vectors of the two sequences
  - The binarization of score values provides some additional robustness against small spectral and tonal differences
Correspondences between versions

- Are revealed by the score matrix in the form of diagonal paths of high score
- Observes a diagonal path indicating that the first 60 seconds of the two versions exhibit a high similarity
For detecting path structures

- Dynamic programming strategies make use of an accumulated score matrix
  - Example of the accumulated score matrix obtained for the score matrix
  - The highest-valued element of the matrix corresponds to the end of the most similar matching subsequences
For detecting path structures

- The highest value is chosen as the final score for the document-level comparison of the two pieces
  - Used for ranking candidate documents to a given query
- The specific alignment path can be obtained by backtracking from this highest element
  - The alignment path is indicated by the red line
Summary of version identification

- Recently, rankings can be improved by combining different scores obtained by different methods

- One of the most challenging problems that remains to be solved is to achieve high accuracy and scalability at the same time, allowing low-specific retrieval in large music collections

- The accuracies achieved with today’s non-scalable approaches have not yet been reached by the scalable ones
Outlook

- Discussed three representative retrieval strategies based on the query-by-example paradigm
  - Provide mechanisms for discovering and accessing music even in cases where the user does not explicitly know what he or she is actually looking for
  - Complement traditional approaches that are based on metadata

- Search tasks of high specificity lead to exact matching problems, which can be realized efficiently using indexing techniques

- Search tasks of low specificity need more flexible and cost-intensive mechanisms for dealing with spectral, temporal, and structural variations
  - The scalability to huge music collections comprising millions of songs still poses many yet unsolved problems
Outlook

- A comprehensive framework that allows a user to adjust the specificity level at any state of the search process
  - The system should be able to seamlessly change the retrieval paradigm from high-specific audio identification, over mid-specific audio matching and version identification to low-specific genre identification
  - The user should be able to flexibly adapt the granularity level to be considered in the search
  - Adjusting the musical properties of the employed similarity measure to facilitate searches according to rhythm, melody, or harmony or any combination of these aspects
Integrated content-based retrieval framework

- A joystick allows a user to continuously and instantly adjust the retrieval specificity and granularity

  (1) A user listens to a recording of Beethoven’s Symphony No. 5, which is first identified to be a Bernstein recording using an audio identification

  (2) Being interested in different versions of this piece, the user moves the joystick upwards and to the right, which triggers a version identification
A joystick allows a user to continuously and instantly adjust the retrieval specificity and granularity.

(3) Shifting towards a more detailed analysis of the piece, the user selects the famous motif as query and moves the joystick downwards to perform some mid-specific fragment-based audio matching.

The system returns the positions of all occurrences of the motif in all available interpretations.
Integrated content-based retrieval framework

- A joystick allows a user to continuously and instantly adjust the retrieval specificity and granularity

(4) Moving the joystick to the rightmost position, the user may discover recordings of pieces that exhibit some general similarity like style or mood

Novel strategies for exploring, browsing, and interacting with large collections of audio content
Another major challenge

- Cross-modal music retrieval scenarios
  - One might use a small fragment of a musical score to query an audio database for recordings that are related to this fragment
  - A short audio fragment might be used to query a database containing MIDI files

- In the future, comprehensive retrieval frameworks are to be developed that offer multi-faceted search functionalities in heterogeneous and distributed music collections containing all sorts of music-related documents
Thank you!

Any question?