Technische Universität Braunschweig

Institute of Operating Systems and Computer Networks

Secure communication based on noisy input data

Feature extraction from audio contexts

Stephan Sigg

May 10, 2011

Overview and Structure

- Classification methods
- Feature extraction
  - Features from audio
  - Features from RF
- Fuzzy Commitment
- Fuzzy Extractors
- Authentication with noisy data
- Error correcting codes
- Entropy
- Physically unclonable functions

Outline

Introduction

- To classify situations from audio samples, sufficient features have to be derived
- Origin of audio feature extraction lies in the representation and processing of speech signals
- Later, also features for non-speech audio signals have been proposed
- Especially for processing of music samples, special features have been proposed
  - E.g. to derive the rhythmic structure, chord change or beat

Introduction

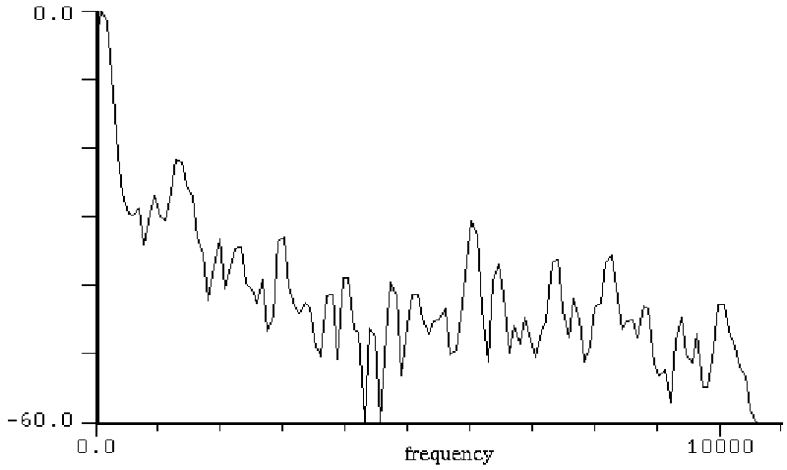
- Features from audio
  - Speech recognition
    - Linear prediction Coefficients (LPC)
    - Mel Frequency Cepstral Coefficients (MFCC)
  - Non-speech audio signals
    - Isolated instrument tones – Instrument classification (Statistical moments of the magnitude spectrum)
    - Sound effects (Spectral shape features (Centroid, Rolloff, Flux)

Introduction

- Features from audio
  - Methods for feature extraction
    - Fast Fourier Transform
    - Cochlear Models (Pitch detection)
    - Wavelets (Statistical characteristics of Wavelet sub-bands such as absolute mean and variance to model sound texture)
    - MPEG audio compression filterbank

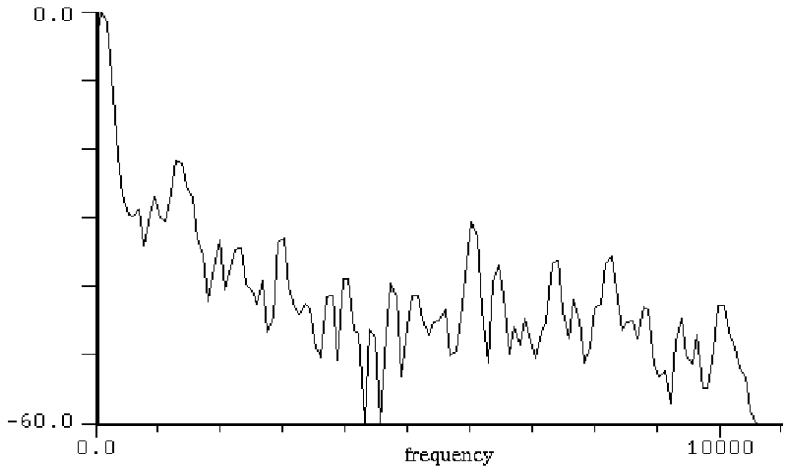
Outline

Features – Spectral shape features



- In time-frequency analysis techniques, signal is typically divided into segments in time and frequency
- Then, the frequency content of each segment is calculated
- E.g. Magnitude spectrum (energy distribution over frequency bands – c.f. figure)

Features – Spectral shape features



- In principle, the frequency spectrum can be utilised to represent and even reconstruct audio
- However, the information content is too high for classification
  - A lot of the information contained in the spectrum is not important
  - Machine learning algorithms typically work better with feature vectors of smaller dimensionality
- after spectrum calculation a small set of characteristic features is extracted

Features – Spectral shape features – STFT-based

- The Short Time Fourier Transform of a signal $x(n)$ is defined as

$$STFT(n,k) = \sum_{m=-\infty}^{\infty} x(m)h(n-m)e^{-j\left(\frac{2\pi}{N}\right)km}$$

- $h(n)$ is a sliding window at time n
- N is the size of the transformation

Features – Spectral shape features – STFT-based

$$STFT(n,k) = \sum_{m=-\infty}^{\infty} x(m)h(n-m)e^{-j\left(\frac{2\pi}{N}\right)km}$$

- Can be viewed as a filter bank where the k-th filter channel is obtained by multiplying the input $x(m)$ by a complex sinusoid at frequency k/N times the sample rate
- The output is for any value of k a frequency shifted, band-pass filtered version of the input
- For any particular value of n the STFT is the DFT of the windowed input at time n
- It can be seen as a partially overlapping Fourier Transform

Features – Spectral shape features – STFT-based

$$STFT(n,k) = \sum_{m=-\infty}^{\infty} x(m)h(n-m)e^{-j\left(\frac{2\pi}{N}\right)km}$$

- The output is a complex number
- For feature generation, typically only the magnitude of this number is considered

Features – Spectral shape features – STFT-based

Spectral Centroid

- The spectral centroid is defined as the center of gravity of the magnitude spectrum of the STFT:

$$SC_t = \frac{\sum_{n=1}^{N} M_t[n] \cdot n}{\sum_{n=1}^{N} M_t[n]}$$

- $M_t[n]$ is the magnitude of the Fourier Transform at frame t and frequency bin n

Features – Spectral shape features – STFT-based

Spectral Centroid

$$SC_t = \frac{\sum_{n=1}^{N} M_t[n] \cdot n}{\sum_{n=1}^{N} M_t[n]}$$

- The centroid is a measure of spectral shape
- higher values correspond to brighter textures with more high frequencies
- It was shown that this feature is an important attribute in the characterisation of musical instrument timbre (Klangfarbe)[1]

[1] J. M. Grey: An Exploration of Musical Timbre, 1975

Features – Spectral shape features – STFT-based

Spectral Rolloff

- The Spectral Rolloff is defined as the frequency F_t below which 85% of the magnitude is concentrated:

$$SR_t = \sum_{n=1}^{F_t} M_t[n] = 0.85 \cdot \sum_{n=1}^{N} M_t[n]$$

- Shows how much of the signal's energy is concentrated in lower frequencies

Features – Spectral shape features – STFT-based

Spectral Flux

- The Spectral Flux is defined as the squared difference between the normalised magnitudes of successive spectral distributions:

$$SF_t = \sum_{n=1}^{N} (N_t[n] - N_{t-1}[n])^2$$

- the $N_t[n]$ are normalised magnitudes of the Fourier Transform at frame t and $t-1$
- Measure of the amount of local spectral change
- It was shown that this feature is an important attribute in the characterisation of musical instrument timbre (Klangfarbe)[2]

[2] J. M. Grey, An Exploration of Musical Timbre, 1975

Features – Spectral shape features – STFT-based

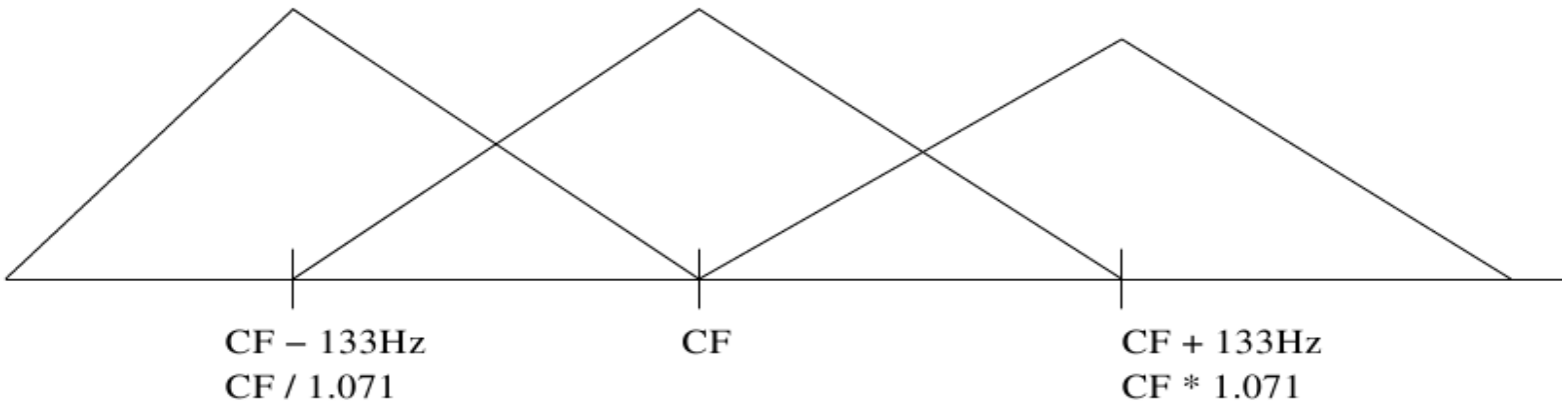
Mel-Frequency Cepstral Coefficients

- Mel-Frequency Cepstral Coefficients (MFCC) are perceptually motivated features that are also based on the STFT
- After taking the log-amplitude of the magnitude spectrum, FFT bins are grouped and smoothed according to the perceptually motivated Mel-frequency scaling
- In order to decorrelate the resulting feature vectors, a Discrete Cosine Transform (DCT) is performed
- Common approach used in many speech recognition systems
- The resulting representation has roughly properties ties similar to the human auditory systems

Features – Spectral shape features – STFT-based

Mel-Frequency Cepstral Coefficients

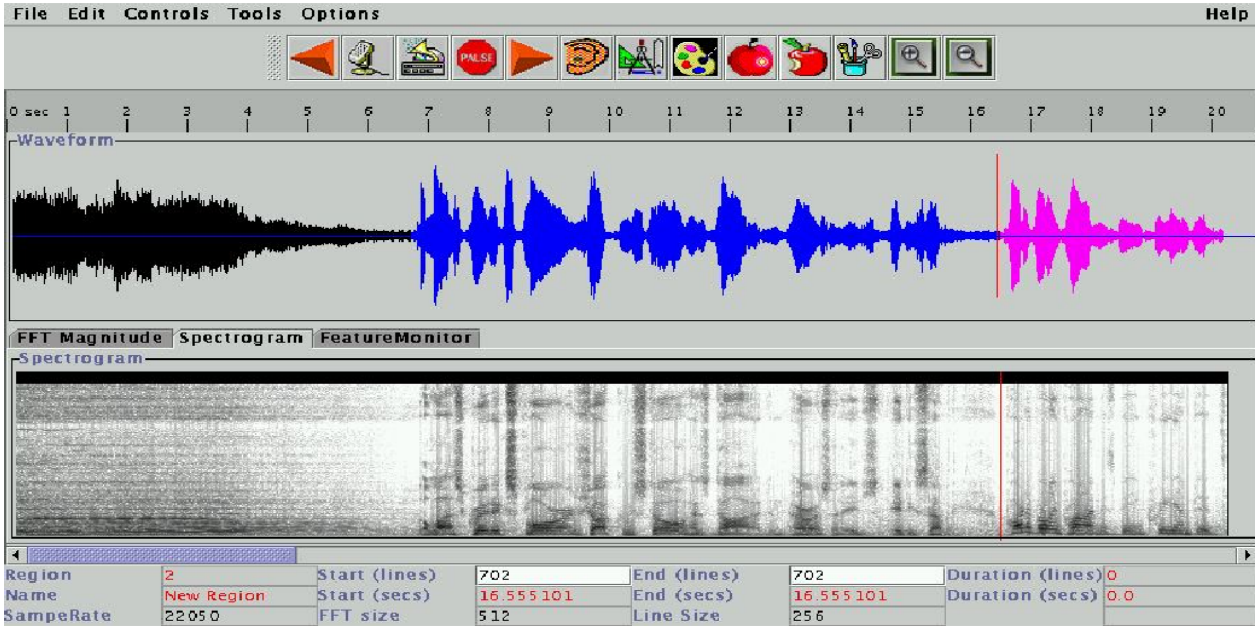
1. Audio data is windowed by a hamming window
2. Magnitude of the DFT is computed
3. FFT bins are applied to log-spaced filters approximating properties of the human ear
4. These bins are summed using a triangular weighting function that starts at the center point of the previous filter.
5. DCT of the output is used to reduce dimensionality



Features – Representing sound texture

Sound texture

- The term Sound texture describes spectral characteristics of audio
- For instance, energy in distinct frequency bands or frequency of changes in energy over time



Features – Representing sound texture

Sound texture

- To compute a sound texture, the signal is broken into small, possibly overlapping segments in time
- Segments have to be small enough so that the frequency characteristics of the magnitude spectrum are relatively stable
- For instance, speech contains vowel and consonant sections each of which have different spectral characteristics
- A texture is a pattern composed of multiple short-time spectrums

Features – Representing sound texture

Sound texture

- Texture patterns can be combined with the mean or variance of the extracted features
- It was shown that the use of texture windows improves the result of automatic musical genre classification [3]

[3] G. Tzanetakis, Manipulation, Analysis and Retrieval systems for audio signals, 2002

Features – Wavelet transform features

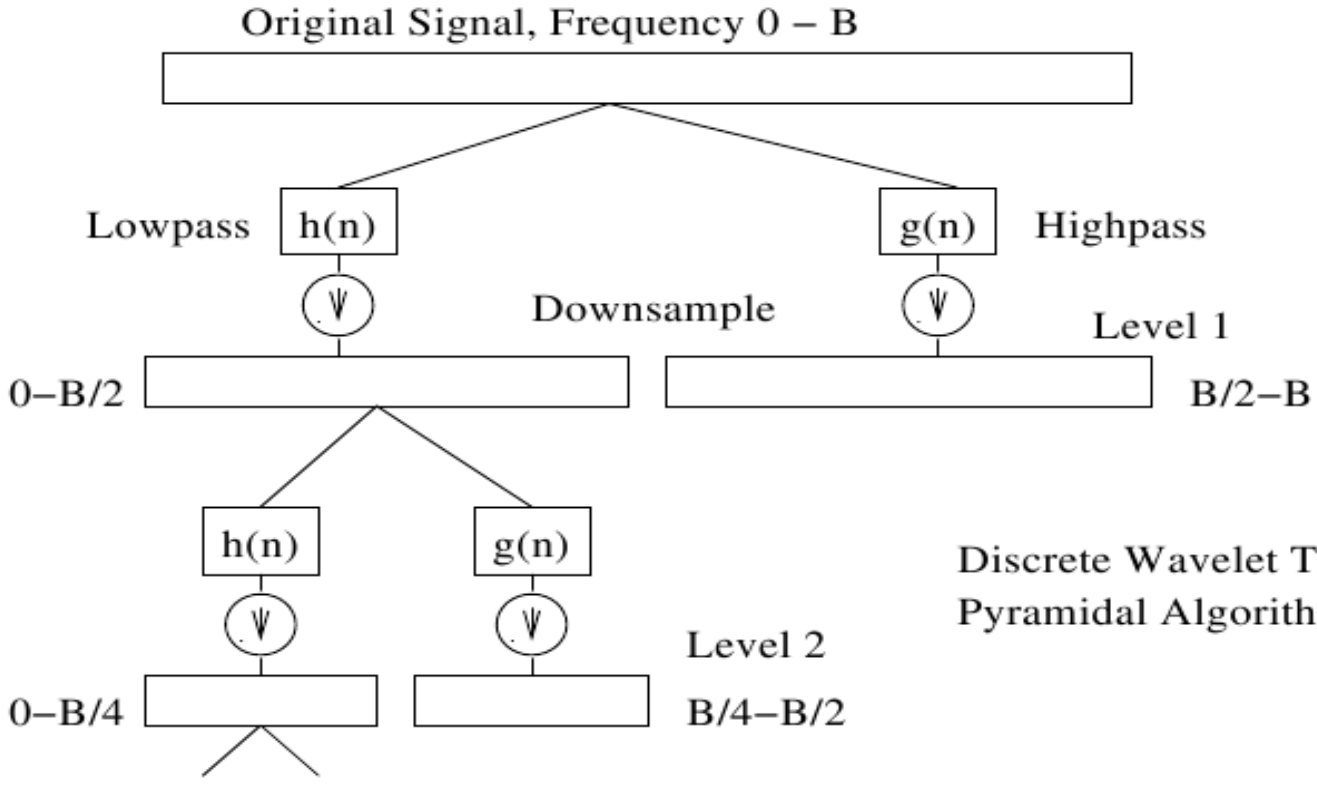
Wavelet transform

- Wavelet transform developed as an alternative to the STFT
- STFT: Uniform resolution for all frequencies
- WT: High time resolution and low frequency resolution for high frequencies, Low time and high frequency resolution for low frequencies
- Time-frequency resolution characteristic similar to human ear

Features – Wavelet transform features

Discrete Wavelet Transform (DWT)

- Fast, pyramidal algorithm[4]



Discrete Wavelet Transform
Pyramidal Algorithm Schematic

[4] A wavelet Tour of Signal Processing. 1999

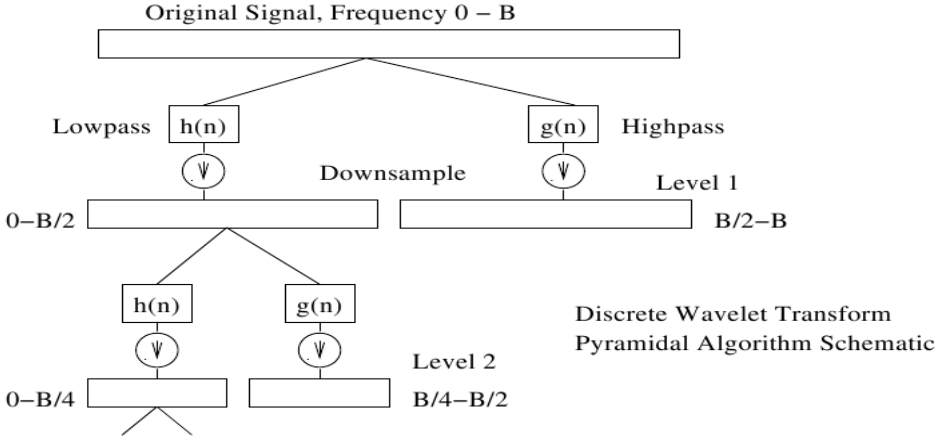
Features – Wavelet transform features

Discrete Wavelet Transform (DWT)

- Signal analysed at different frequency bands with different resolutions for each band
- Successive highpass and lowpass filtering of the time domain signal:

$$y_{high}(k) = \sum_n x(n) highpass(2k - n)$$

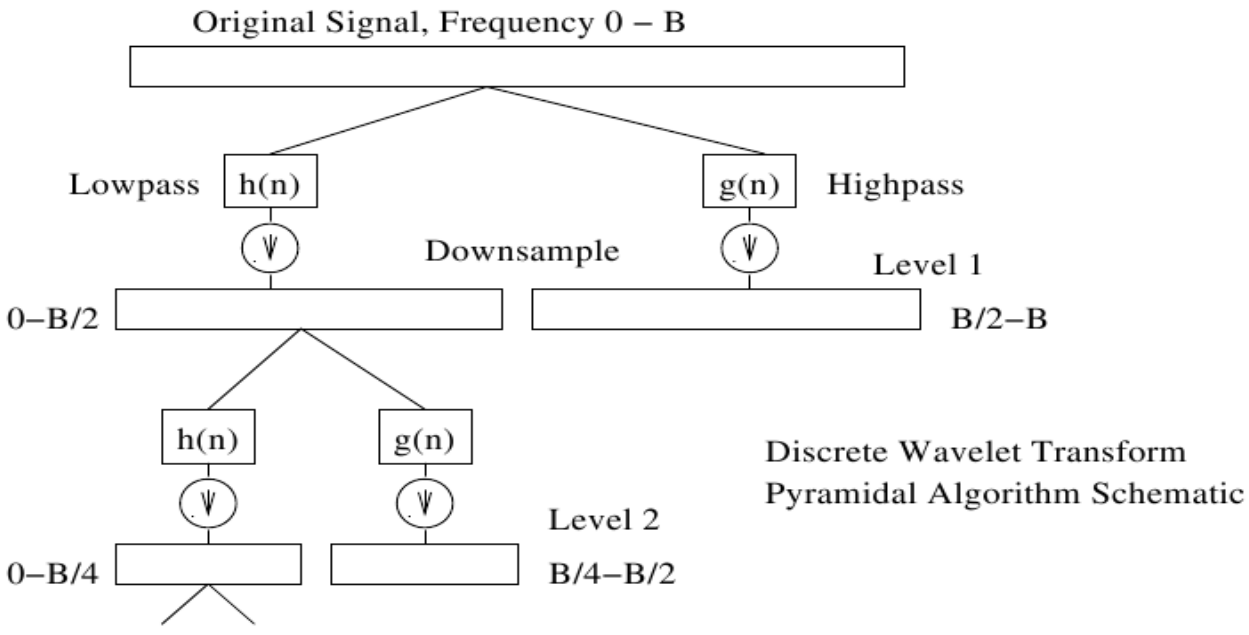
$$y_{low}(k) = \sum_n x(n) lowpass(2k - n)$$



Features – Wavelet transform features

Discrete Wavelet Transform (DWT)

- Each level of the pyramid corresponds roughly to frequency bands spaced in octaves
- DWT can be performed in $\mathcal{O}(N)$ for N input data points



Discrete Wavelet Transform
Pyramidal Algorithm Schematic

Features – Wavelet transform features

Wavelet transform

- The extracted wavelet coefficients provide compact representation for the energy of the signal in time and frequency
- For instance: Distribution of energy in time and frequency is different between music and speech
- Features to represent sound texture:
  - Mean of the absolute value of the coefficients in each subband (information on the frequency distribution of the audio signal)
  - Standard deviation of the coefficients in each subband (information on the amount of change of the frequency distribution over time)
  - Ratio of the mean absolute values between adjacent subbands (Information about frequency distribution)

Features – Zero crossing

Time domain zero crossings

- Time domain zero crossings provide a measure of the noisiness of the signal

$$TDZC_t = \frac{1}{2} \sum_{n=1}^{N} |sign(x(n)) - sign(x(n-1))|$$

LPC coefficients are used as an estimate of the vocal tract filter [5]

[5] Linear prediction: A tutorial overview, Proceedings of the IEEE, 63,561-580 1975

Features – Other features

RMS

- RMS is a measure of the loudness of a window:

$$RMS = \sqrt{\frac{\sum_{i=1}^{N}(M(i)^2)}{N}}$$

- Important to detect new sound events which are often accompanied by loudness changes
- Not well suited for classification which is often required to be loudness invariant

Features – Other features

Pitch

- Three general approaches to detect Pitch (Tonlage)[6]
  - Utilise time-domain properties
    - Peak measurements, Zero crossings
  - Utilise frequency-domain properties
    - Analyse impulses on the frequency spectrum
  - Hybrid approaches

[6] L. Rabiner, M. Cheng, A. Rosenberg, C. McGonegal, A comparative performance study of several pitch detection

Features – Other features

Pitch

- Autocorrelation [7]
- Maximum Likelihood [8]
- Cepstrum analysis [9]

[7] Rabiner, Dubnowdki, Schafer, Real-time digital hardware pitch detector, In IEEE Transactions on Acoustics, Speech and Signal Processing, ASSP-24(1), 1976

[8] Wise, Caprio, Parks, Maximum likelihood pith estimation, In Ieee Transactions on Acoustic, Speech, Signal processing, 24(5), 1976

[9] Oppenheim, A speech analysis-sythesis system based on hormomorphic filtering, In Journal of the acoustical society

Features – Other features

Pitch detection by Autocorrelation

- Isolates and tracks the peak energy levels of the signal
- Tracking the frequency of the peaks can provide the pitch
- With autocorrelation it can be obtained as

$$R(l) = \sum_{k=-\infty}^{\infty} h(k)h(l+k)$$

Features – Other features

Pitch detection by Autocorrelation

$$R(l) = \sum_{k=-\infty}^{\infty} h(k)h(l+k)$$

- A problem with autocorrelation is that is is subject to picking an integer multiple of the actual pitch

Features – LPC

Linear prediction reflection coefficients (LPC)

- LPC coefficients are used as an estimate of the vocal tract filter [10]
  - Linear prediction: Derive a linear model of a time series description for a time series of observed samples
  - Utilise parameters of the time series description as characteristic feature

[10] J. Makhoul, Linear prediction: A tutorial overview, Proceedings of the IEEE, 63,561-580 1975

Features – LPC

- Despite recent developments with nonlinear models, some of the most common stochastic models in time series prediction are parametric linear models as autoregressive (AR), moving average (MA) or autoregressive moving average (ARMA) processes[11]
- Examples for application scenarios:
  - Financial time series prediction
  - Wind power prediction.

[11] Jan Peter KreiÃ and Georg Neuhaus, *EinfÃ¼hrung in die Zeitreihenanalyse, Springer, 2006.*

Features – LPC

- Assume a stochastic process $\pi(t)$ that generates outputs $\chi(t)$ at each point t in time
  - Random values $\chi(t)$ can be univariate or multivariate and can take discrete or continuous values
  - Time can also be either discrete or continuous.
- Task: Find parameters $\Theta = \{\theta_1, \ldots, \theta_n\}$ that describe the stochastic mechanism[12]
- Prediction accomplished by calculating conditional probability density $P(\chi(t)|\{\Theta, \{\chi(t-1), \ldots, \chi(t-m)\}\})$.

[12] R.O. Duda, P.E. Hart and D.G. Stork, *Pattern Classification*, Wiley Interscience, 2001.

Features – LPC

Moving average (MA) models

- Let $Z(t)$ be some fixed zero-mean, unit-variance random process.
- $\chi(t)$ is a $MA(k)$ process (MA-process of order k), if

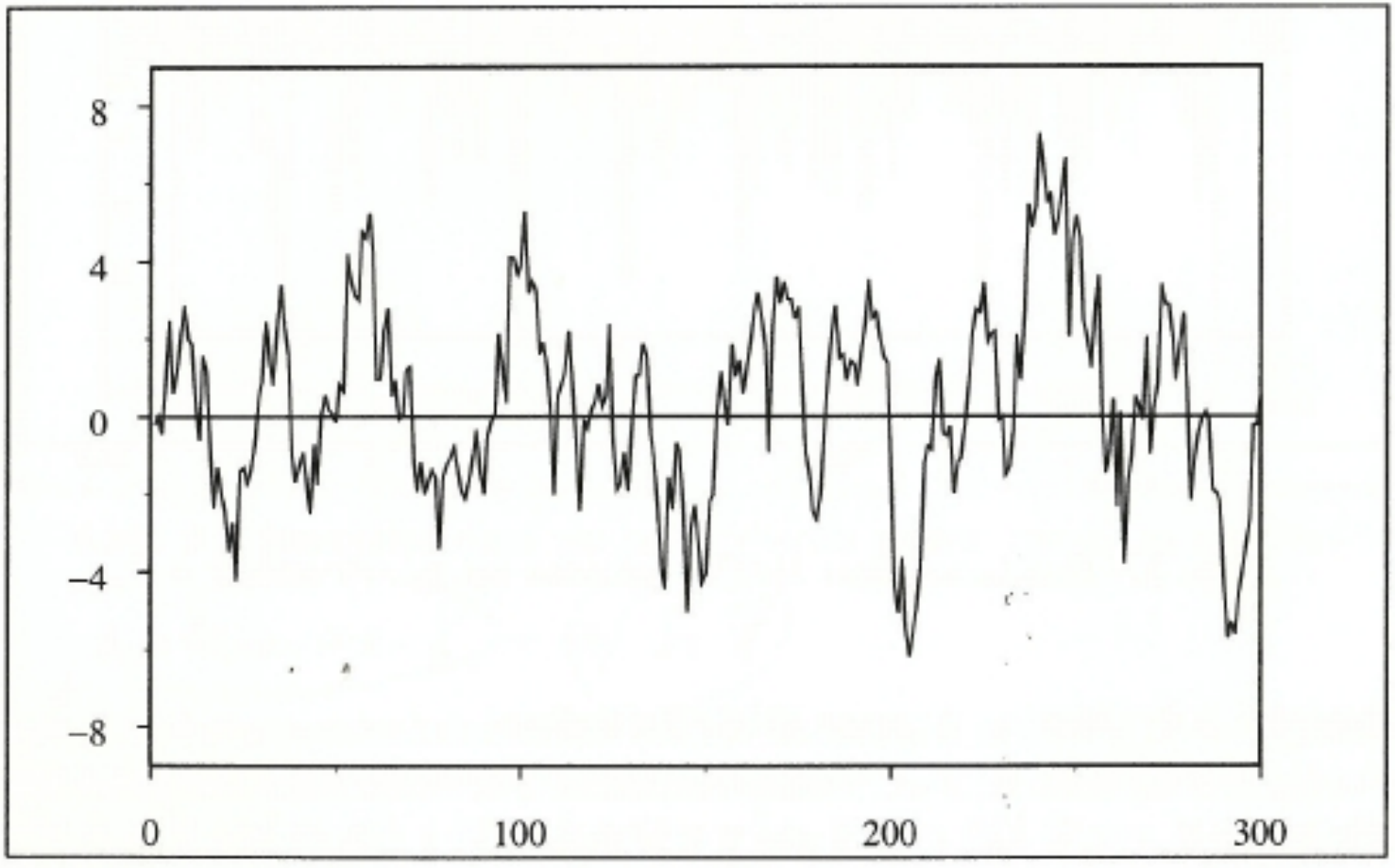
$$\chi(t) = \sum_{\tau=0}^{k} \beta_\tau Z(t-\tau)$$

where the β_τ are constants.

- Moving average processes are utilised to describe stochastic processes with finite, short-term linear memory [13]

[13] C. Chatfield, *The Analysis of Time Series: An Introduction*, Chapman and Hall, 1996.

Features – LPC



Features – LPC

Autoregressive (AR) models

- AR processes, the values at time t depend linearly on previous values
- $\chi(t)$ is an $AR(k)$ process of order k, if

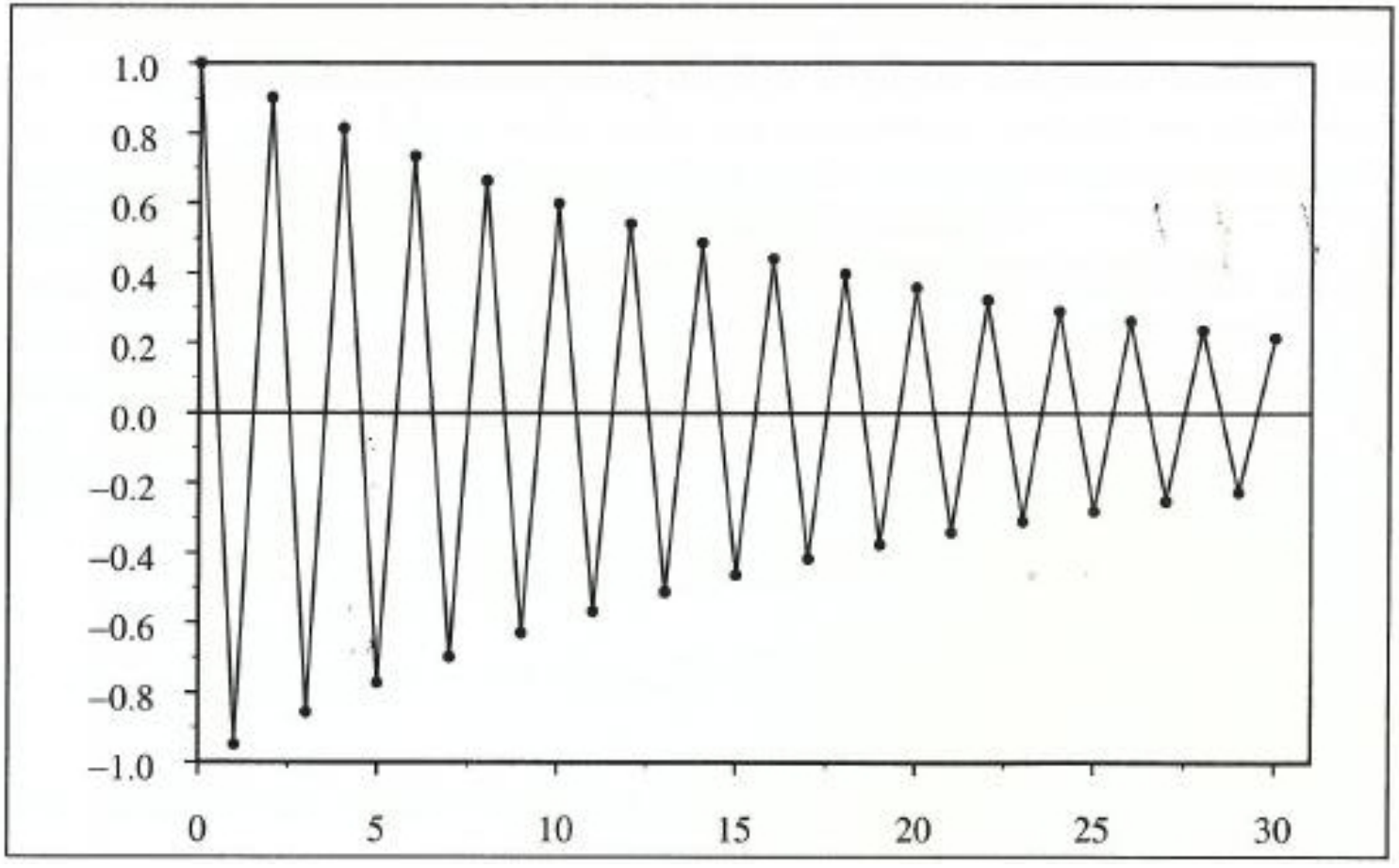
$$\sum_{\nu=0}^{k} \alpha_\nu \chi(t-\nu) = \chi(t)$$

where α_ν are constants.

- Autoregressive processes are used to capture exponential traces [14]

[14] C. Chatfield, *The Analysis of Time Series: An Introduction*, Chapman and Hall, 1996.

Features – LPC



Features – LPC

ARMA models

- ARMA processes are a combination of AR and MA processes.
- An $ARMA(p, q)$ process is a stochastic process $\chi(t)$ in which

$$\sum_{\nu=0}^{p} \alpha_\nu \chi(t-\nu) = \sum_{\tau=0}^{q} \beta_\tau Z(t-\tau)$$

, where $\{\alpha_\nu, \beta_\tau\}$ are constants [15]

[15] C. Chatfield, *The Analysis of Time Series: An Introduction*, Chapman and Hall, 1996.

Features – LPC

- ARMA methods provide a powerful tool to approximate stochastic processes
- Computational complexity can be estimated as $O(k \log(k))$ [16]
- No prior pre-processing or separate learning tool required.

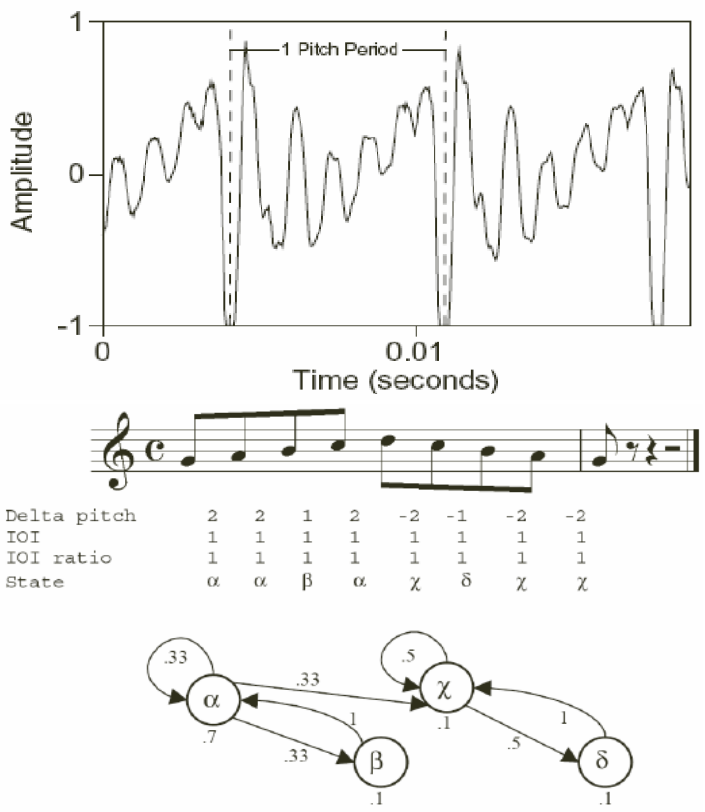
[16] J. Cadzow and K. Ogino, *Adaptive ARMA spectral estimation* Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, 1981.

Outline

Audio-fingerprinting

Some approaches to Query by humming

- Detect coarse melodic contour [a]
- Add rhythm information [b]
- Detect beat [c]



[a] Ghias, Logan, Chamberlin, Smith, Query by humming - musical information retrieval in an audio database. In ACM Multimedia, 1995

[b] McNab, Smith, Witten, Henderson, Cunningham, Toward the digital music library: tune retrieval from acoustic input. Proceedings of the ACM Digital Libraries, 1996

[c] Chai, Vercoe, Melody Retrieval On the Web. In Proceedings of ACM/SPIE Conference on multimedia computing and networking, 2002

Designed for music description – not for general audio files

Audio-fingerprinting

Audio-fingerprinting

- An Audio-fingerprint is a characteristic representation for an audio sequence
- Comparison to watermarking
  - Original audio need not be modified
  - Robust to encoding of the audio (MP3,Ogg,...)
- Applications
  - Search in an audio database (Identify Title and author) [17]
  - Duplicate detection [18]

[17] Bellettini, Mazzini, A Framework for Robust Audio Fingerprinting. In Journal of Communications, vol. 5, No. 5, May 2010

[18] Burges, Plastina, Platt, Renshaw, Malvar, Using Audio Fingerprinting for Duplicate Detection and Thumbnail [illegible] dings of the IEEE International Conference on Acoustics, Speech and Signal Processing, 2005

Audio-fingerprinting

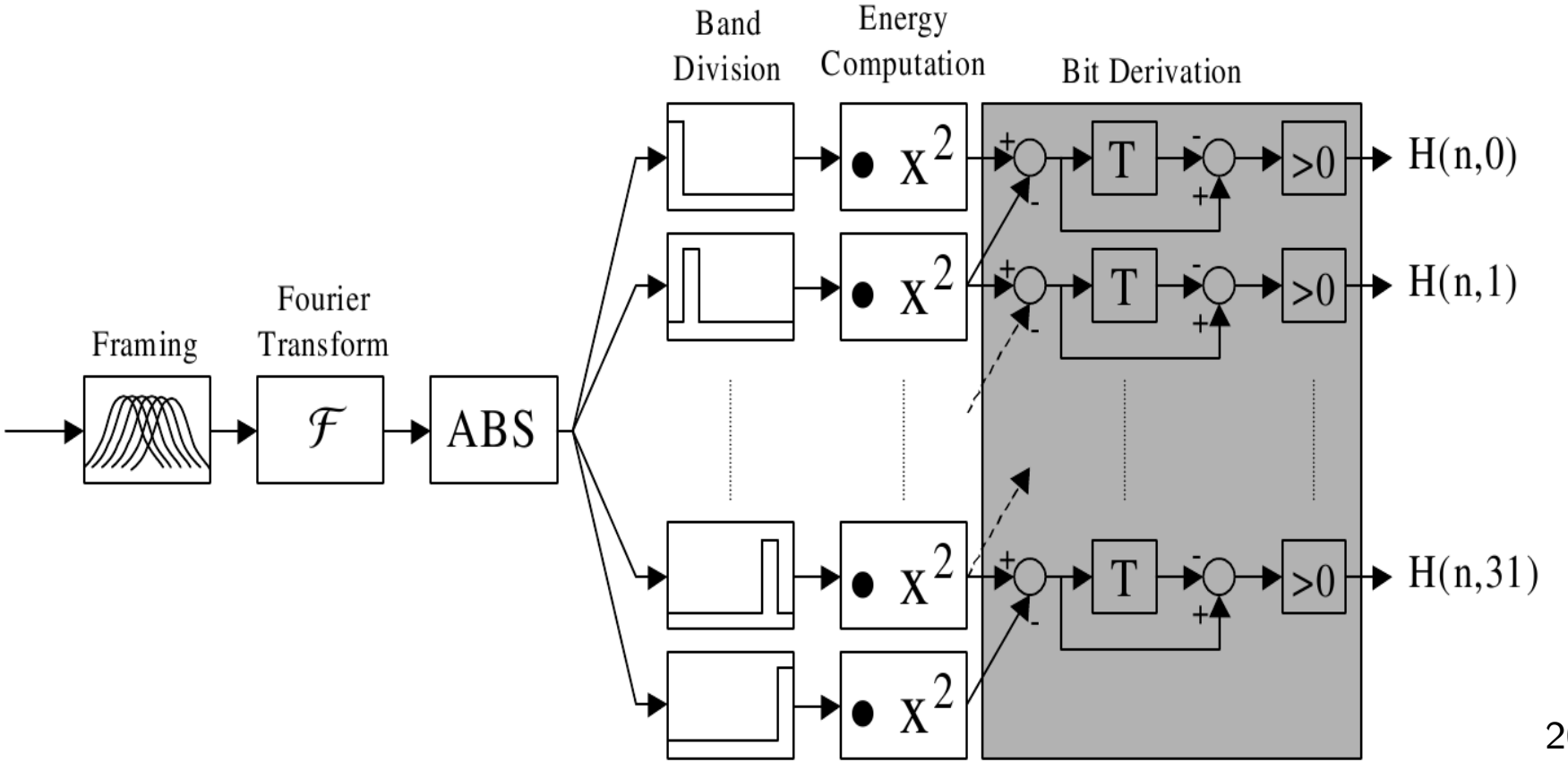
Audio-fingerprinting

- Most approaches based on similar scheme:
  1. Audio signal is segmented into frames
  2. Features are computed for every frame
  3. Features should be invariant to some degree of signal degradation
  4. Popular features:[19]
     - Fourier coefficients
     - Mel Frequency Cepstral Coefficients (MFCC)
     - Spectral Flatness
     - Sharpness
     - Linear Predictive Coding (LPC) coefficients

[19] ker, A highly robust audio fingerprinting system. In Proceedings of the IRCAM, 2002

Audio-fingerprinting

Audio fingerprinting based on energy differences in frequency bands



20

[20] Haitsma, Kalker, Oostveen, Robust Audio Hashing for Content Identification, Content-based multimedia indexing, 2001

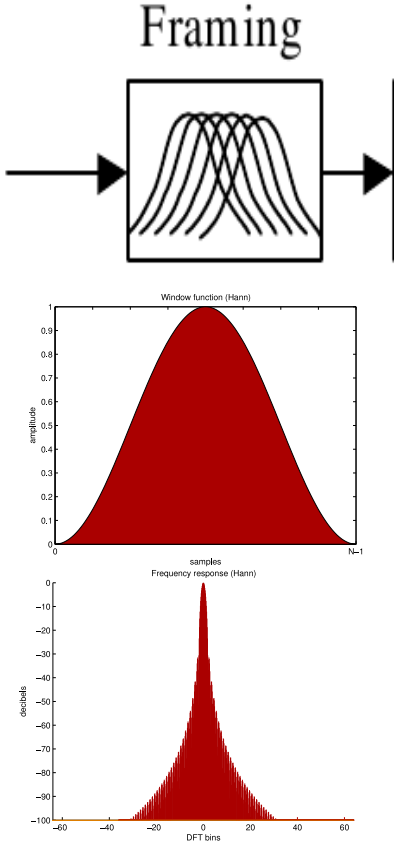
Audio-fingerprinting

Audio fingerprinting based on energy differences in frequency bands

- Audio sequence is divided into frames
- Weighted by a Hanning window

$$w(n) = 0.5 \cdot \left(1 - \cos\left(\frac{2\pi n}{N-1}\right)\right)$$

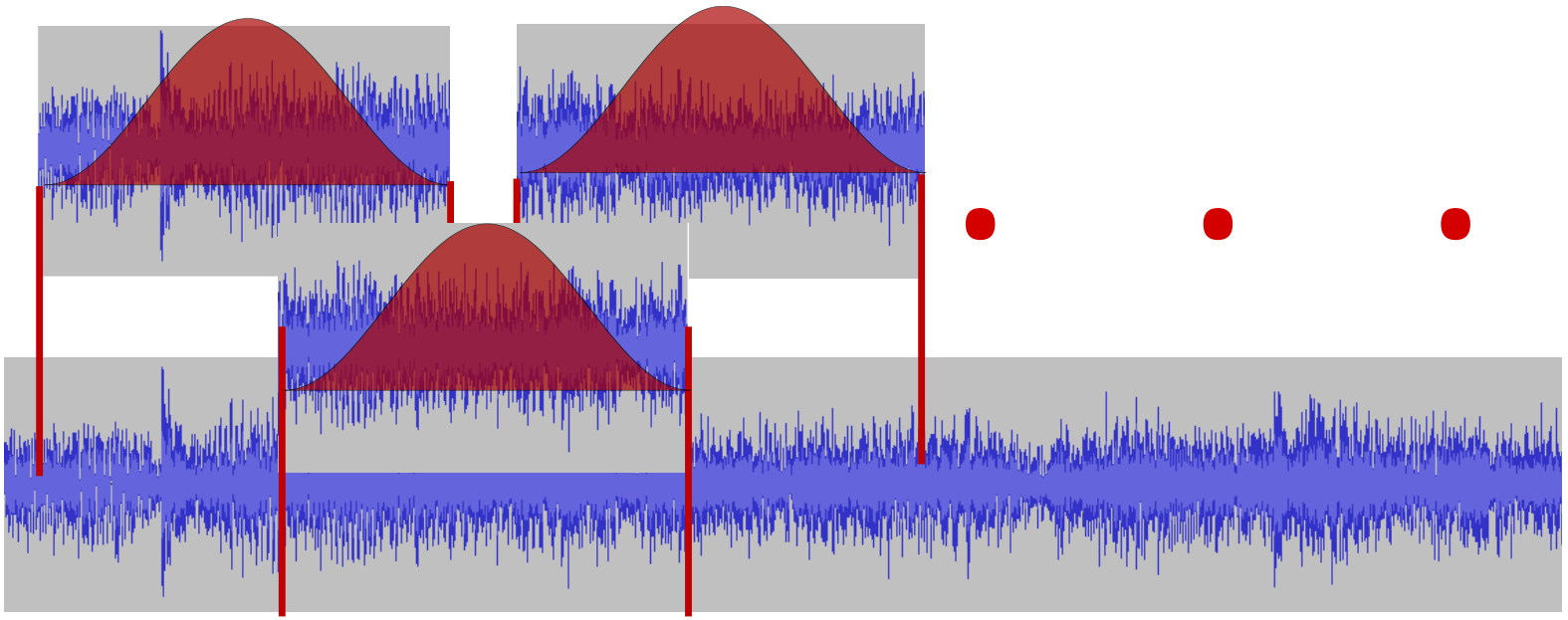
- Possible overlap between windows
  - Might impair the entropy of the resulting fingerprint![a]



[a] Ibarrola, Chavez, A robust entropy-based audio-fingerprint. Proceedings of the 2006 International Conference on Multimedia and Expo, 2006

Audio-fingerprinting

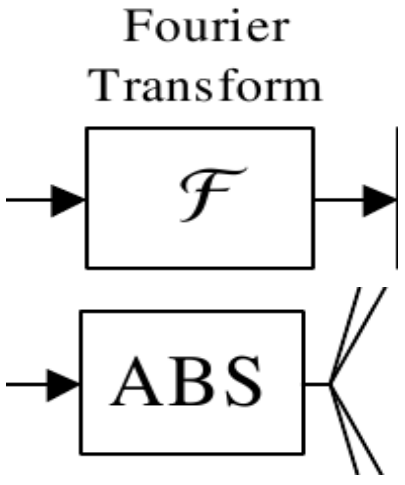
Audio fingerprinting based on energy differences in frequency bands

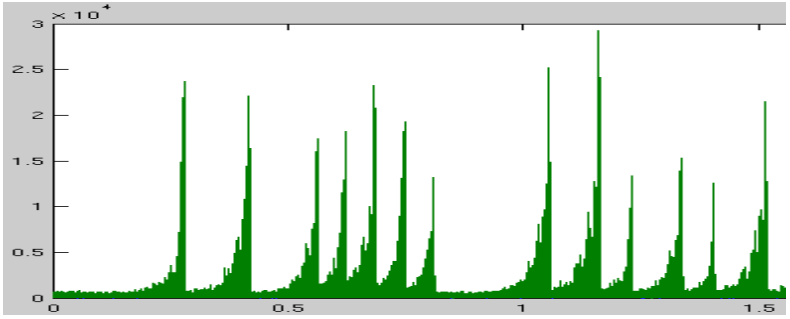


Audio-fingerprinting

Audio fingerprinting based on energy differences in frequency bands

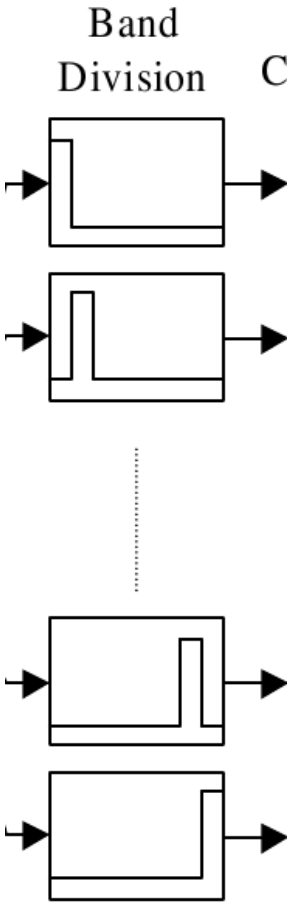
- Transformation of frames into spectral representation
  - Fourier Transform
  - Cosine transform
- Typically, absolute value is utilised





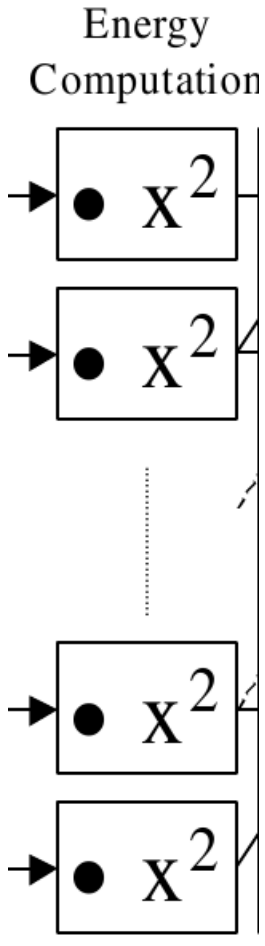
Audio-fingerprinting

Audio fingerprinting based on energy differences in frequency bands



- Each frame is divided into separate non-overlapping frequency bands
- For each frequency band, the energy is created
- Stored to an energy matrix
  - Energy of frame i in band j:

$$E(i,j)$$

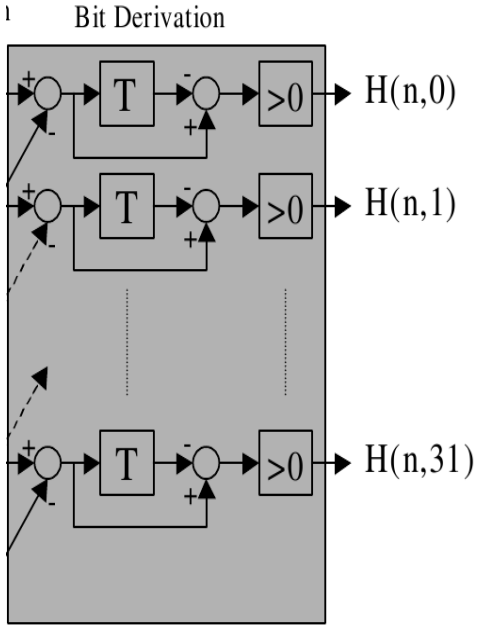


Audio-fingerprinting

Audio fingerprinting based on energy differences in frequency bands

- Fingerprint created from the energy difference in concurrent frames of each frequency bands

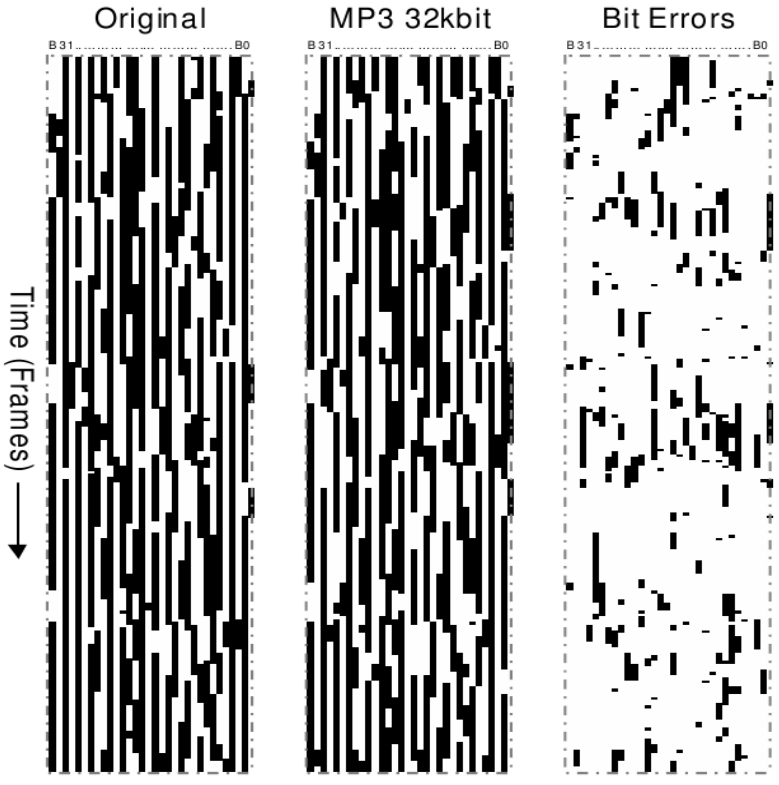
$$F(i,j) = \begin{cases} 1 & E(i,j) - E(i,j+1) - \\ & (E(i-1,j) - E(i-1,j+1)) > 0 \\ 0 & \text{otherwise.} \end{cases}$$



Audio-fingerprinting

Audio fingerprinting based on energy differences in frequency bands

- Discussion
  - Not sensitive to loudness changes
  - Applicable to general audio



Audio-fingerprinting

Audio fingerprinting based on spectrogram peaks[21]

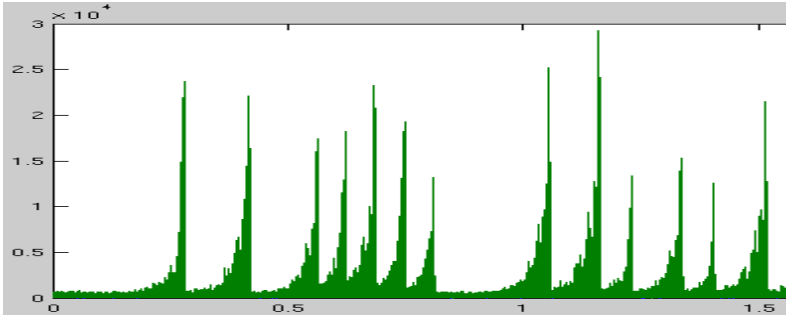
1. Convert signal into Spectrogram and split it into possibly overlapping frames with a Hanning window
2. Perform FFT on windowed segments and take absolute values
3. Plot the instantaneous power as a function of time
4. Identify peaks in the power plot
5. Extract frequency components near each peak
6. Pass vectors through a set of comb filters representing different pitch levels
7. Construct characteristic sequence

[21] Cheng Yang, MACS: music audio characteristic sequence indexing for similarity retrieval, IEEE Workshop on the Applications of Signal Processing, 2001

Audio-fingerprinting

Audio fingerprinting based on spectrogram peaks

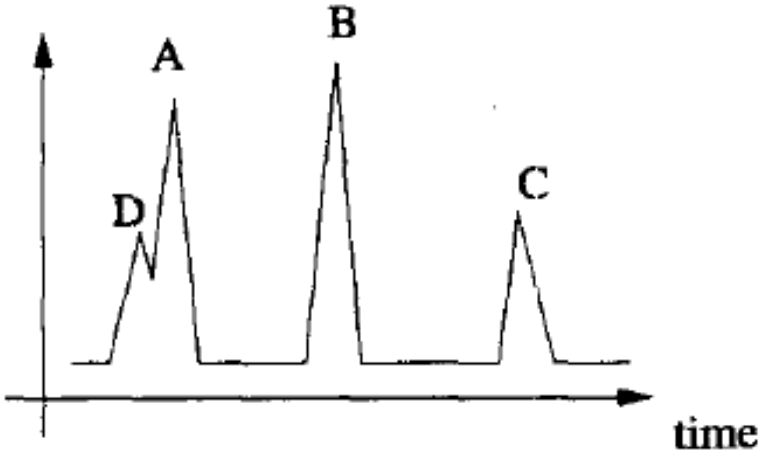
1. Convert signal into Spectrogram and split it into possibly overlapping frames with a Hanning window
2. Perform a zero-padded FFT on each windowed segment and take the absolute values
3. Plot the instantaneous power as a function of time



Audio-fingerprinting

Audio fingerprinting based on spectrogram peaks

4. Identify peaks in the power plot

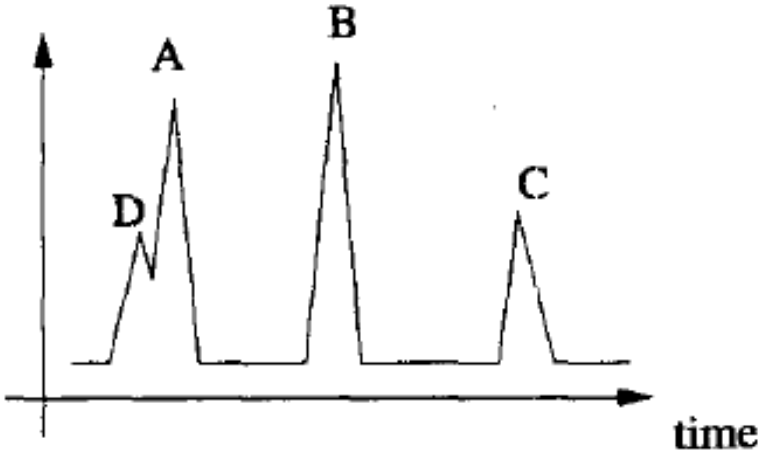


- Typically, 100-200 power peaks in 60 seconds audio file

Audio-fingerprinting

Audio fingerprinting based on spectrogram peaks

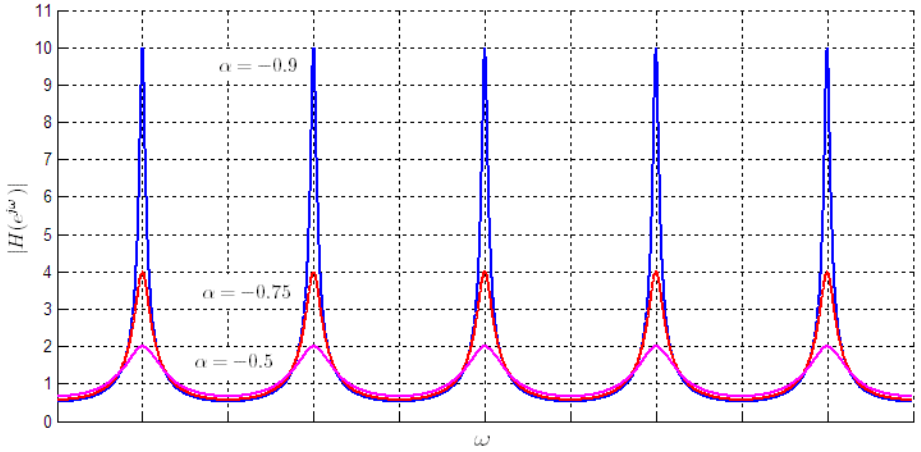
5. Extract frequency components near each peak
   - k samples of frequency components are taken
   - Average values over a short time period following the peak are used
   - This step generates a list of n (count of peaks) k-dimensional vectors



Audio-fingerprinting

Audio fingerprinting based on spectrogram peaks

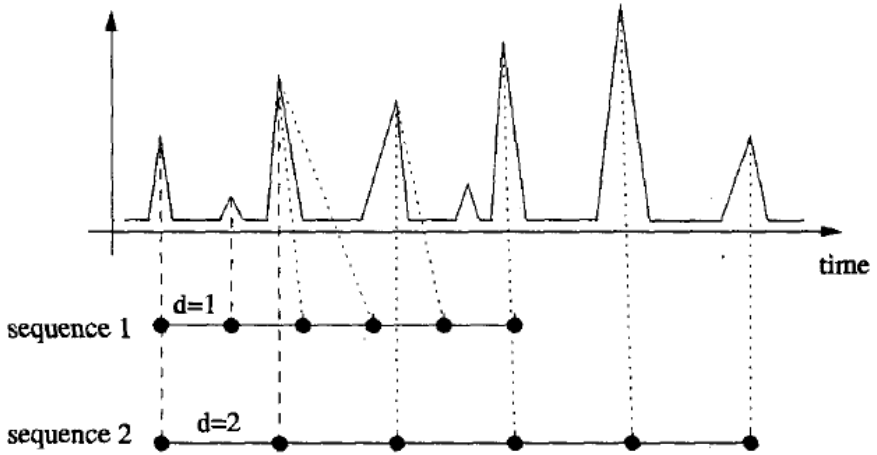
6. Pass vectors through a set of comb filters representing different pitch levels
   - Comb filter adds a delayed version of a signal to itself
   - Causing constructive and destructive interference
   - Frequency response of a comb filter consists of a series of regularly spaced spikes
7. Normalise all vectors to have a mean 0 and variance 1



Audio-fingerprinting

Audio fingerprinting based on spectrogram peaks

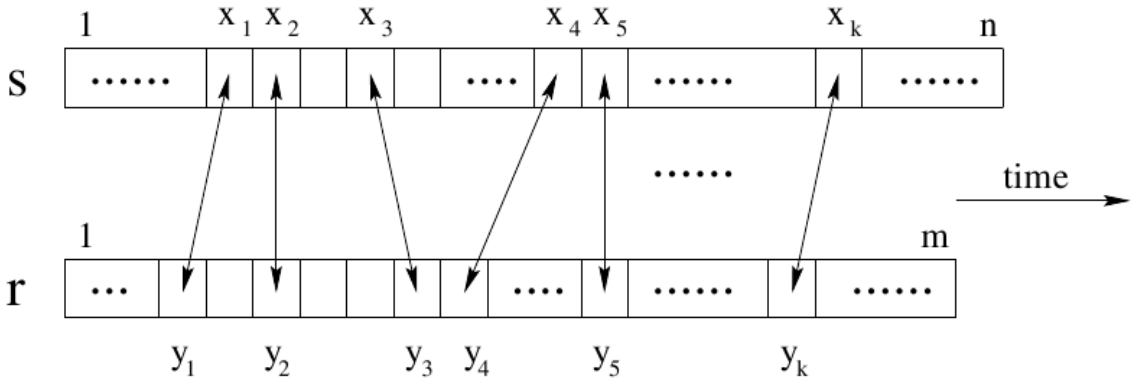
8. Construct characteristic sequences
   - For any two nearby peaks separated by fewer than D other peaks
   - Identify sequence of follow-up peaks which maintain roughly equal distance as the first two peaks
     - $\{v_s, v_{s+d}, V_{t_s+2(t_{s+d}-t_s)}, \ldots, V_{t_s+(M-1)(t_{s+d}-t_s)}\}$



Audio-fingerprinting

Audio fingerprinting based on spectrogram peaks

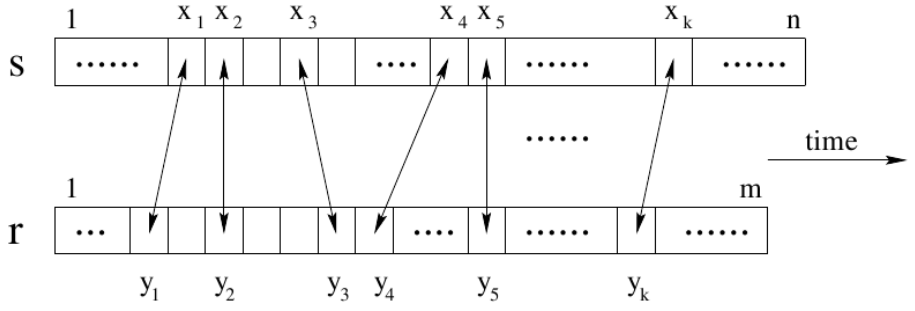
- Matching distinct sequences



- For two sequences $s_1, s_2, \ldots, s_n$ and $r_1, r_2, \ldots, r_m$, e_{ij} is the RMS between s_i and r_j
- Smaller e_{ij} correspond to larger correlation coefficient

Audio-fingerprinting

Audio fingerprinting based on spectrogram peaks



- Given: 2 subsets $s^a = s_1, s_2, \ldots, s_a$ and $r^b = r_1, r_2, \ldots, r_b$ of s and r
- and a matching sequence $M_k = \{(x_1, y_1), \ldots, (x_k, y_k)\}$
- The distance of s^a and r^b with respect to M_k is defined as

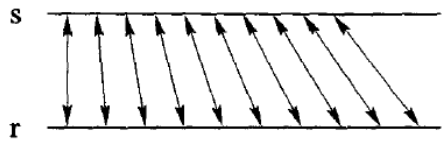
$$D_{a,b,M_k} = \left(\sum_{i=1}^{k} e_{x_i y_i} \right) + \beta(a + b - 2k)$$

- Minimum distance:

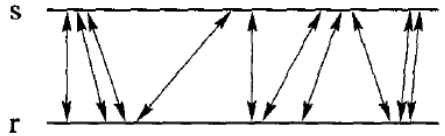
$$D_{a,b} = \min_{M} D_{a,b,M}$$

Audio-fingerprinting

Audio fingerprinting based on spectrogram peaks



A "good" match



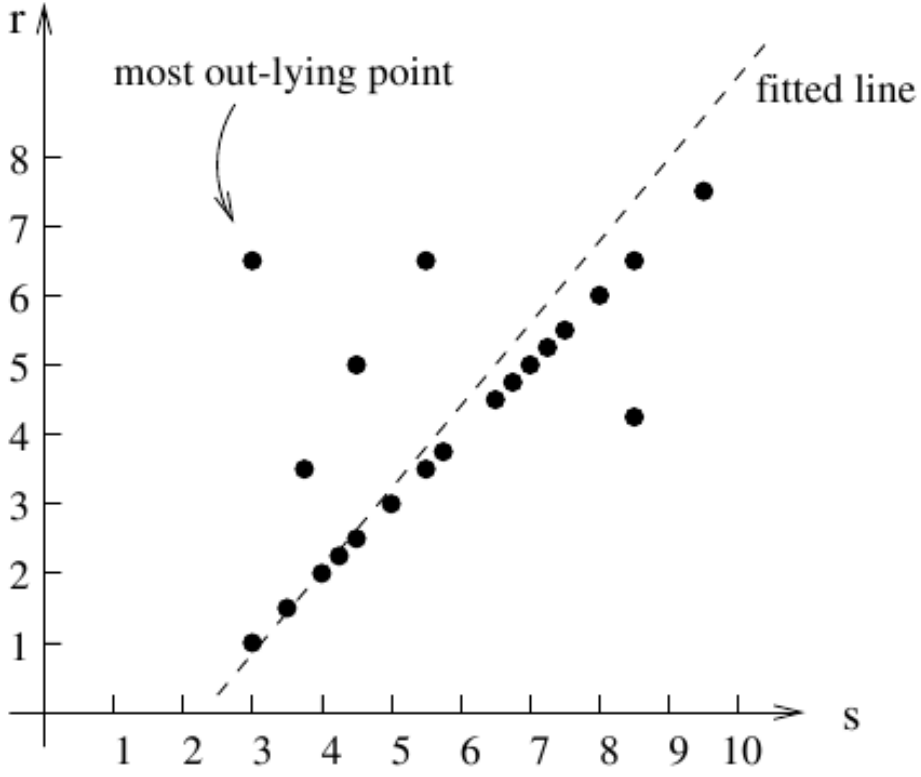
A "bad" match

- Even when the sum of the RMS is small for two pieces of audio, there is a probability that they do not match
  - The reason is that the points at which the vectors match are differently distributed
- The matching set $M_k = \{(x_1, y_1), \ldots, (x_k, y_k)\}$ can be plotted on a 2D graph to test for similar distribution of peak vectors

Audio-fingerprinting

Audio fingerprinting based on spectrogram peaks

- The matching set $M_k = \{(x_1, y_1), \ldots, (x_k, y_k)\}$ can be plotted on a 2D graph to test for similar distribution of peak vectors



Audio-fingerprinting

Similar approaches also implemented by other authors[22]

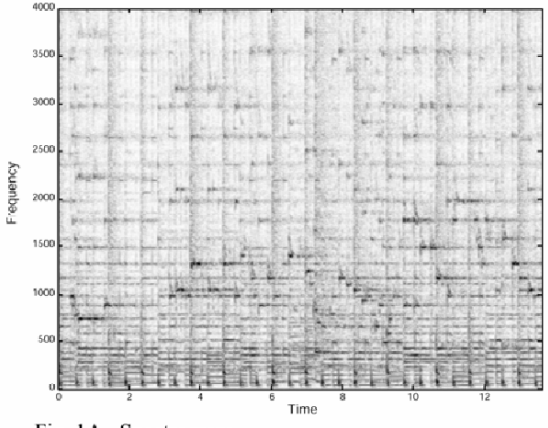


Fig. 1A - Spectrogram

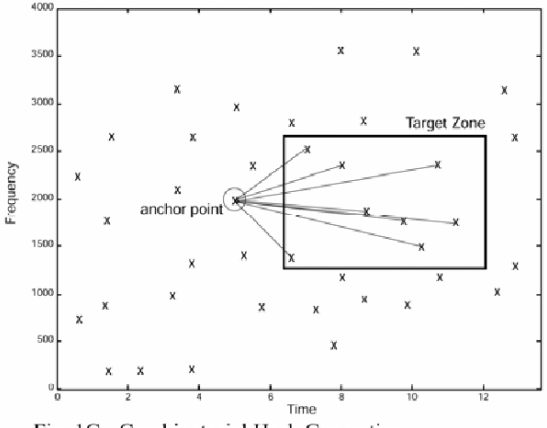


Fig. 1C - Combinatorial Hash Generation

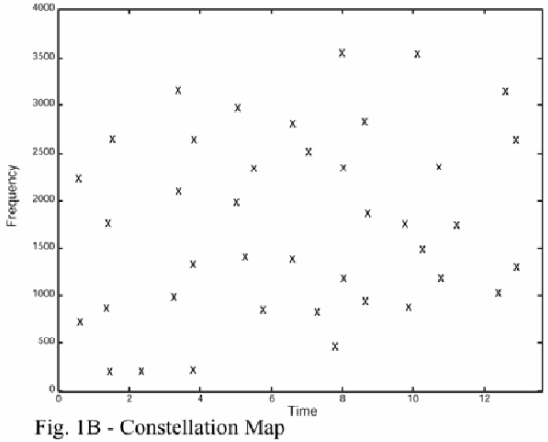


Fig. 1B - Constellation Map

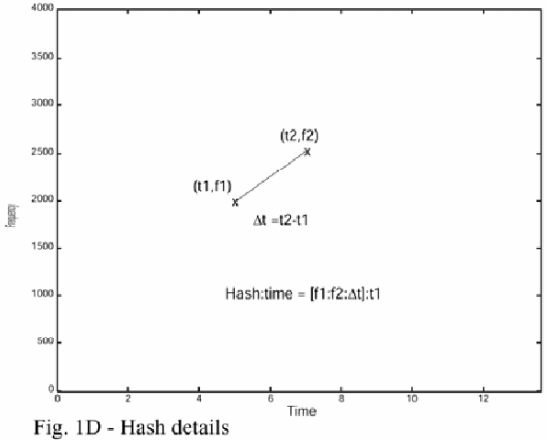


Fig. 1D - Hash details

[22] Wang, An industrial-strength audio search algorithm, proceedings of the 4th Symposium Conference on Music ...al, 2003

Audio-fingerprinting

Audio fingerprinting based on spectral flatness[23]

1. Spectral flatness defines the energy of a signal in an infinitesimal small band
2. Sketched over the whole frequency spectrum, we obtain the Power spectrum density (PSD(k))
3. The spectral flatness measure is defined as

$$SFM = \frac{\left[\prod_{k=0}^{N-1} PSD(k)\right]^{\frac{1}{N}}}{\frac{1}{N}\sum_{k=0}^{N-1} PSD(k)}$$

[23] Herre, Allamanche, Hellmuth, Robust matching of audio signals using spectral flatness features, Proceedings of the IEEE Workshop Applications of Signal Processing to Audio and Acoustics, 2001

Outline

Questions?

Stephan Sigg
sigg@ibr.cs.tu-bs.de

Literature

- C.M. Bishop: Pattern recognition and machine learning, Springer, 2007.
- P. Tulys, B. Skoric, T. Kevenaar: Security with Noisy Data – On private biometrics, secure key storage and anti-counterfeiting, Springer, 2007.
- R.O. Duda, P.E. Hart, D.G. Stork: Pattern Classification, Wiley, 2001.