Audio Fingerprinting

EECS 352: Machine Perception of Music & Audio

Credit

- Most of the content was stolen… I mean borrowed from:
  - Meinard Müller and Joan Serrà, "Audio Content-Based Music Retrieval (tutorial)," *12th International Society for Music Information Retrieval*, Miami, FL, USA, October 24-28, 2011
  - http://ismir2011.ismir.net/tutorials/2011_MuellerSerra_MusicRetrieval_Tutorial-ISMIR_handouts-2.pdf

Outline

- **Introduction**
  - **Context, literature, etc.**
- Shazam
  - Fingerprinting, matching, etc.
- Philips
  - Fingerprinting, matching, etc.
- Conclusion
  - Advantages, limitations, etc.

Problem

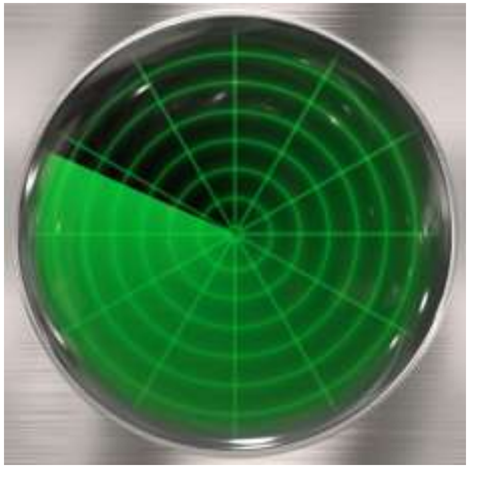
- You are at home, in your car, in a café, etc.
  - You hear an audio signal (e.g., a song)
  - You want to quickly know more about it (e.g., title)
  - You have a smart device (e.g., a smartphone)

Solution

- You use an audio identification system
  - You record an excerpt of the audio signal
  - It is compared against a database for a match
  - You get information about the audio signal

Principle

- Audio identification works as follows:
  - Convert the audio signal into an audio fingerprint
  - Generate a database of known references
  - Match an unknown query against the database



Requirements

- Audio fingerprints have to be:
  - Compact (= small storage and fast search)
  - Discriminative (= less false positives)
  - Robust (= invariance to audio degradations)

Literature

- Haitsma et al., 2002 (Philips)
  – Sign of energy differences in time and frequency
- Burges et al., 2003 (Microsoft)
  – Two-level Principal Component Analysis (PCA)
- Wang et al., 2003 (Shazam)
  – Pairs of time-frequency peaks from spectrogram
- Baluja et al., 2007 (Google)
  – Sign of wavelets from spectrogram

Literature

- Haitsma et al., 2002 (Philips)
  - Sign of energy differences in time and frequency
- Burges et al., 2003 (Microsoft)
  - Two-level Principal Component Analysis (PCA)
- Wang et al., 2003 (Shazam)
  - Pairs of time-frequency peaks from spectrogram
- Baluja et al., 2007 (Google)
  - Sign of wavelets from spectrogram

Outline

- Introduction
  – Context, literature, etc.
- **Shazam**
  – **Fingerprinting, matching, etc.**
- Philips
  – Fingerprinting, matching, etc.
- Conclusion
  – Advantages, limitations, etc.

Shazam

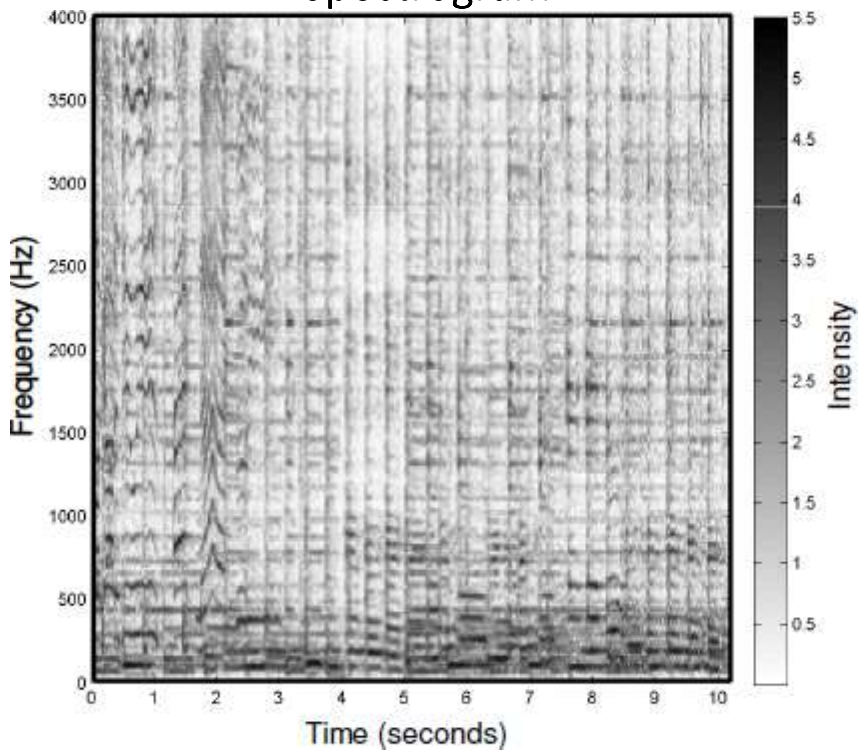
- Background
  - Based on the work of Avery Wang
  - Founded in 1999, commercialized in 2002
  - Database of more than 11 millions of songs

www.shazam.com

Fingerprinting

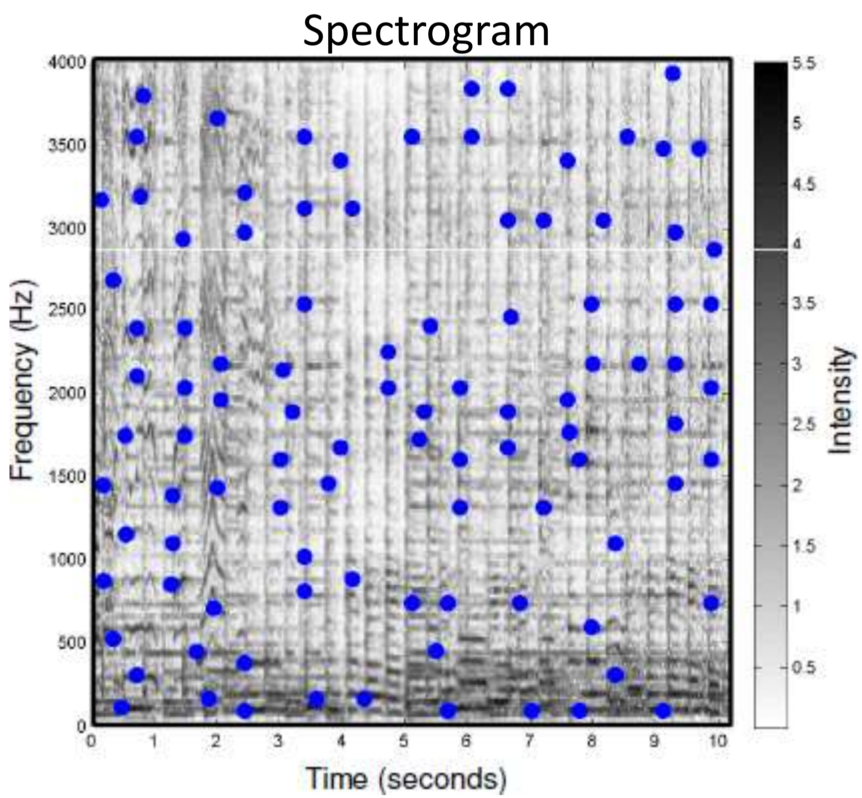
- The audio signal (e.g., a song) is first transformed into a spectrogram

Spectrogram



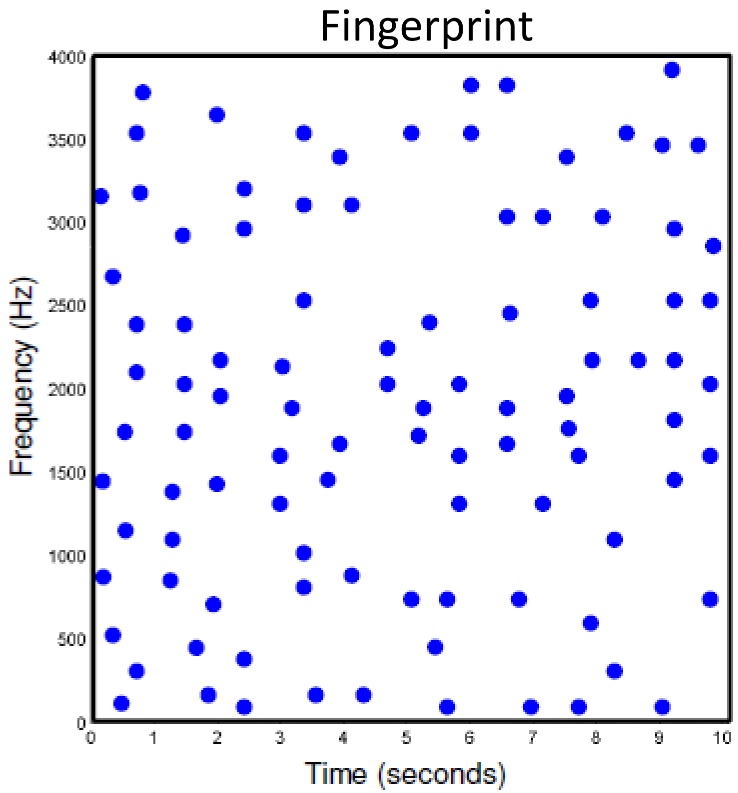
Fingerprinting

- Peak locations in the spectrogram are identified given some criteria (e.g., density)



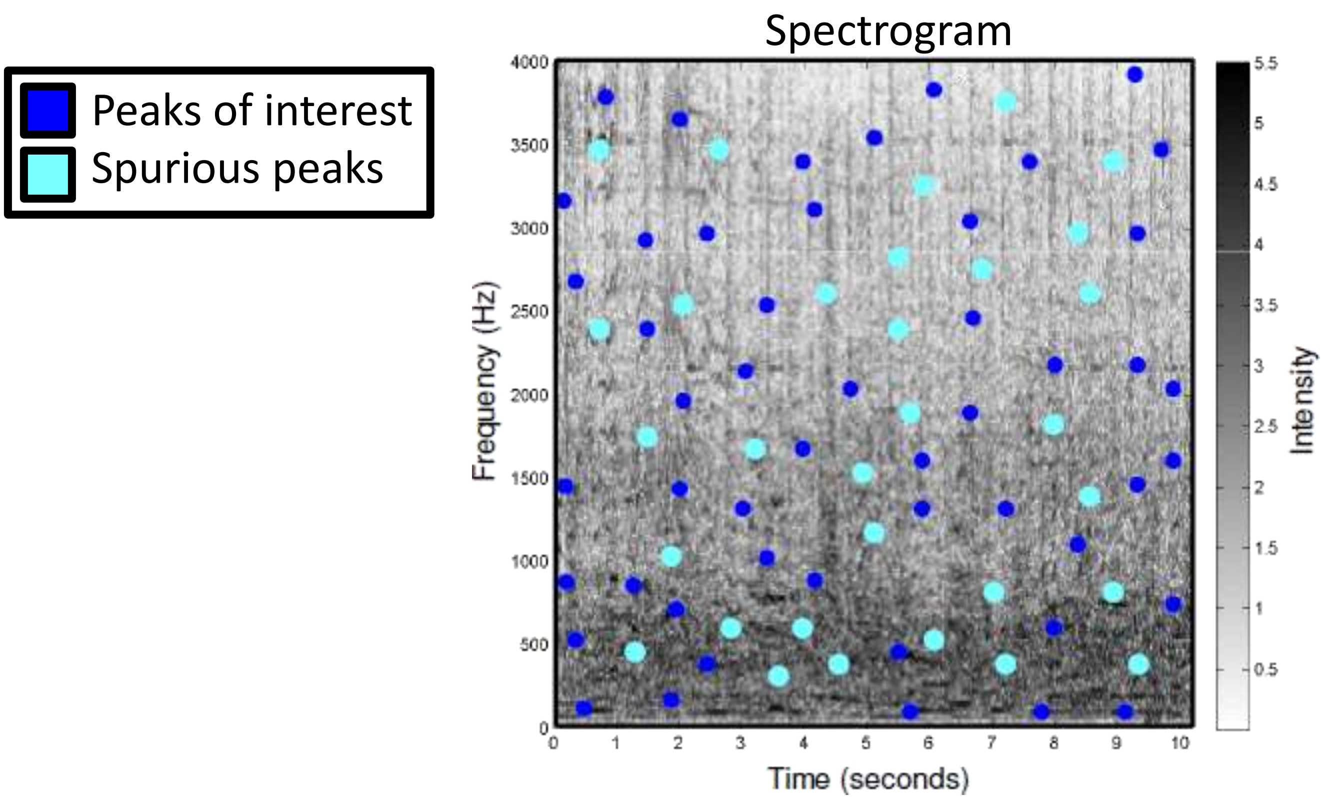
Fingerprinting

- This leads to an audio fingerprint that is both compact and robust to audio degradations



Fingerprinting

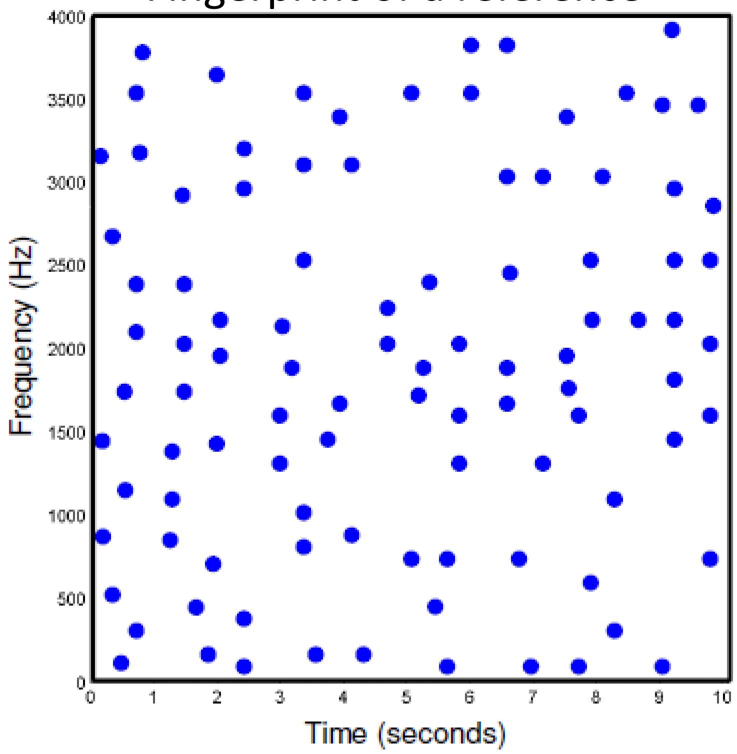
- In the presence of noise or distortion, most peaks should survive as they have high energy



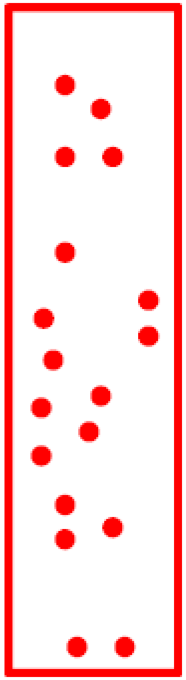
Matching

- A fingerprint is extracted from the query and compared to the fingerprints of the references

Fingerprint of a reference

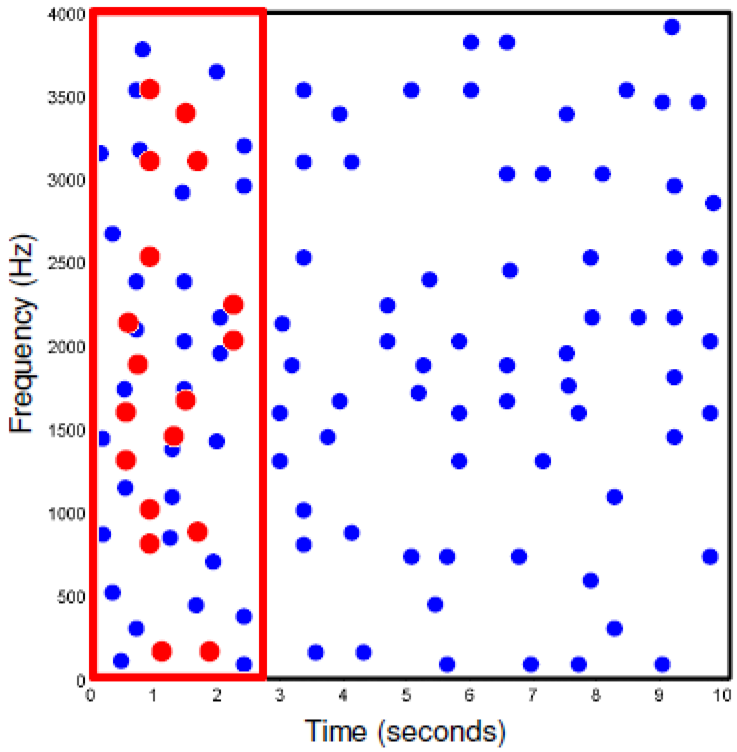


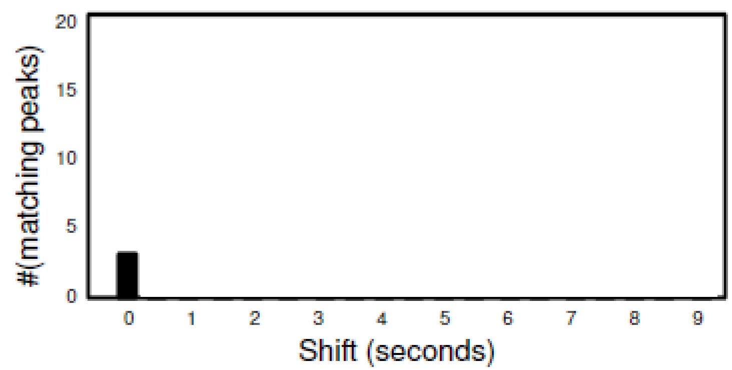
Fingerprint of the query



Matching

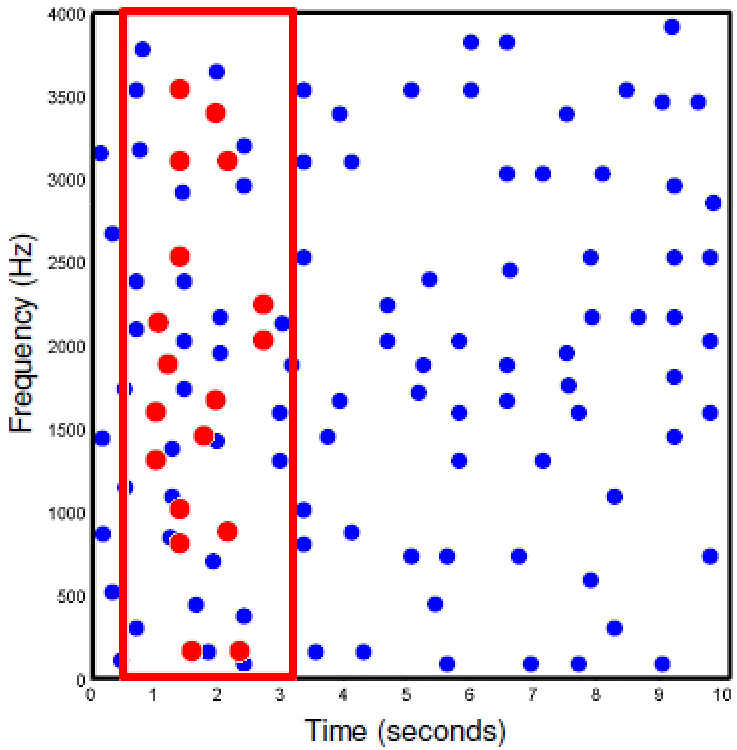
- The query fingerprint is shifted along time against every reference fingerprint

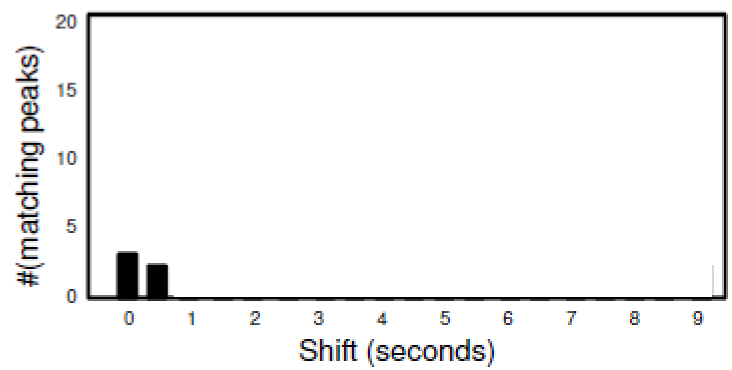




Matching

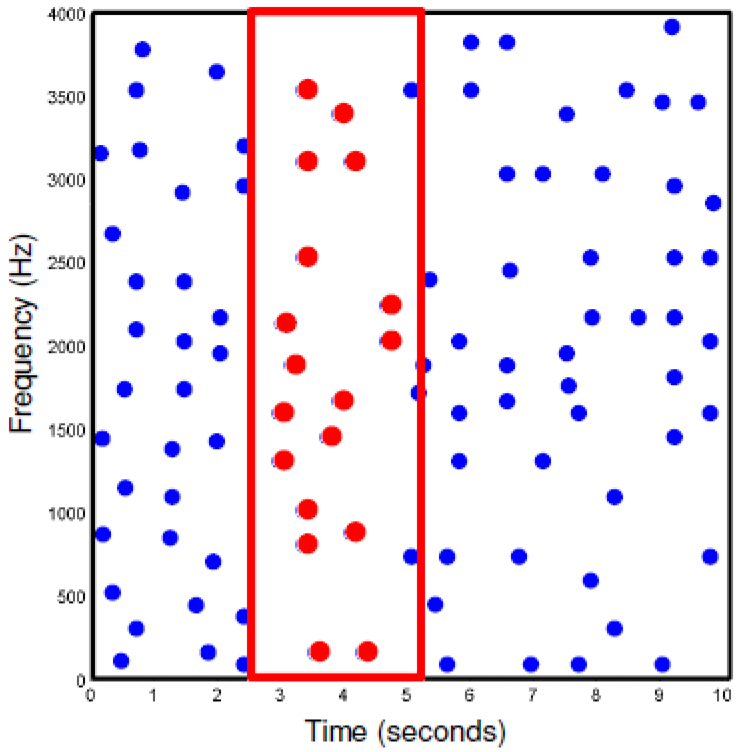
- The number of peaks that are matching is counted and saved for every possible shift

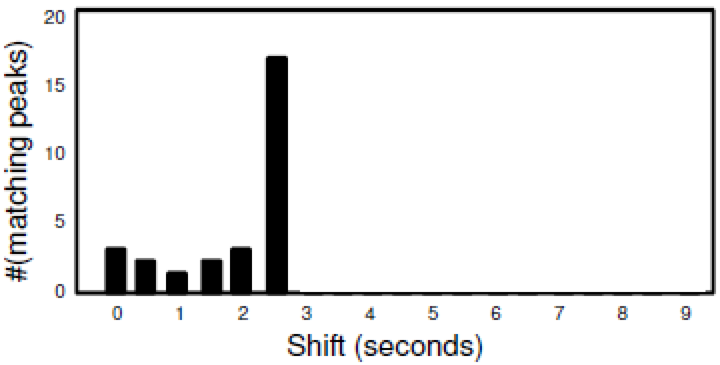




Matching

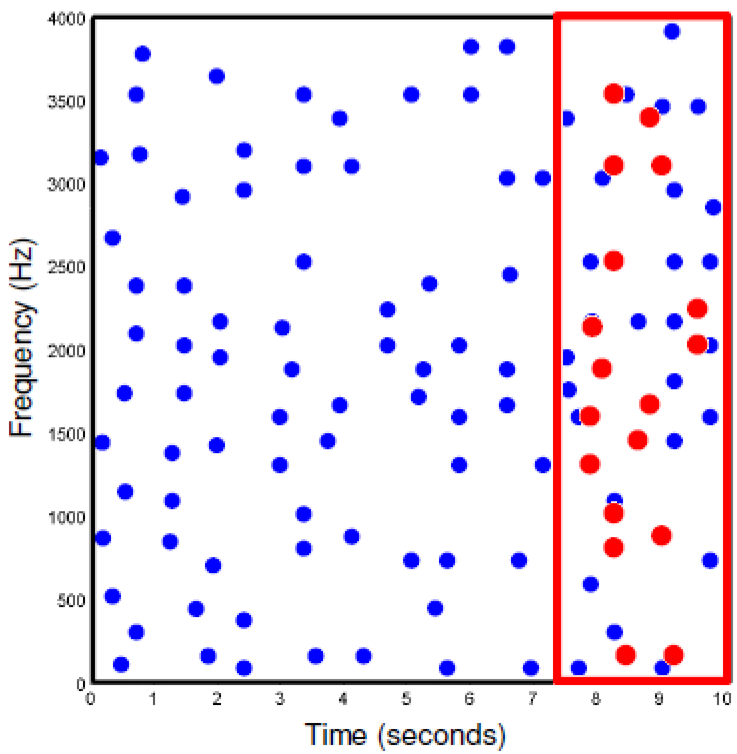
- The number of peaks that are matching is counted and saved for every possible shift

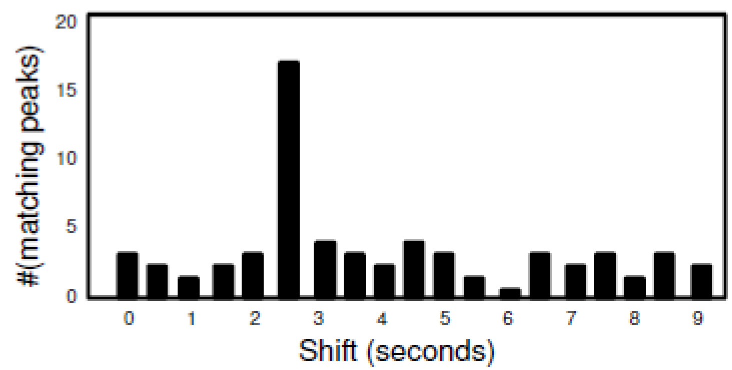




Matching

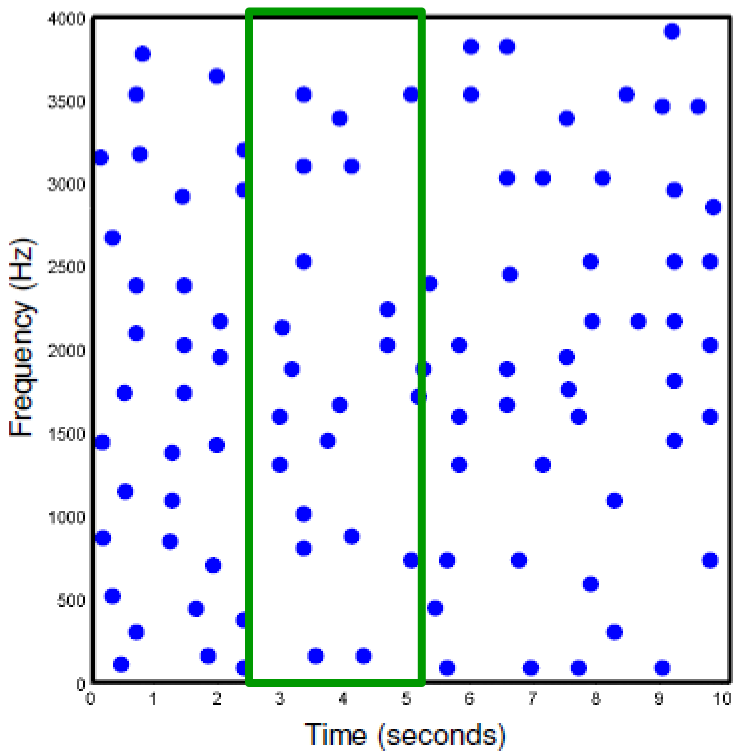
- The number of peaks that are matching is counted and saved for every possible shift

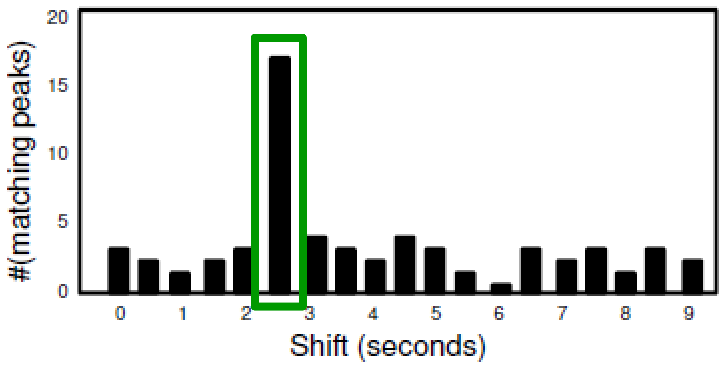




Matching

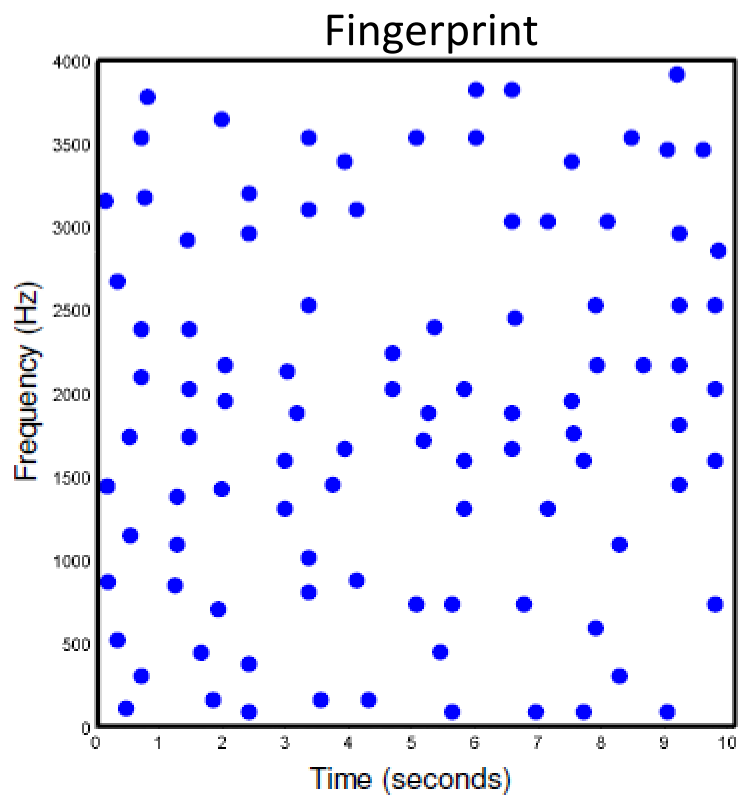
- A high count indicates a match, and the corresponding reference is identified





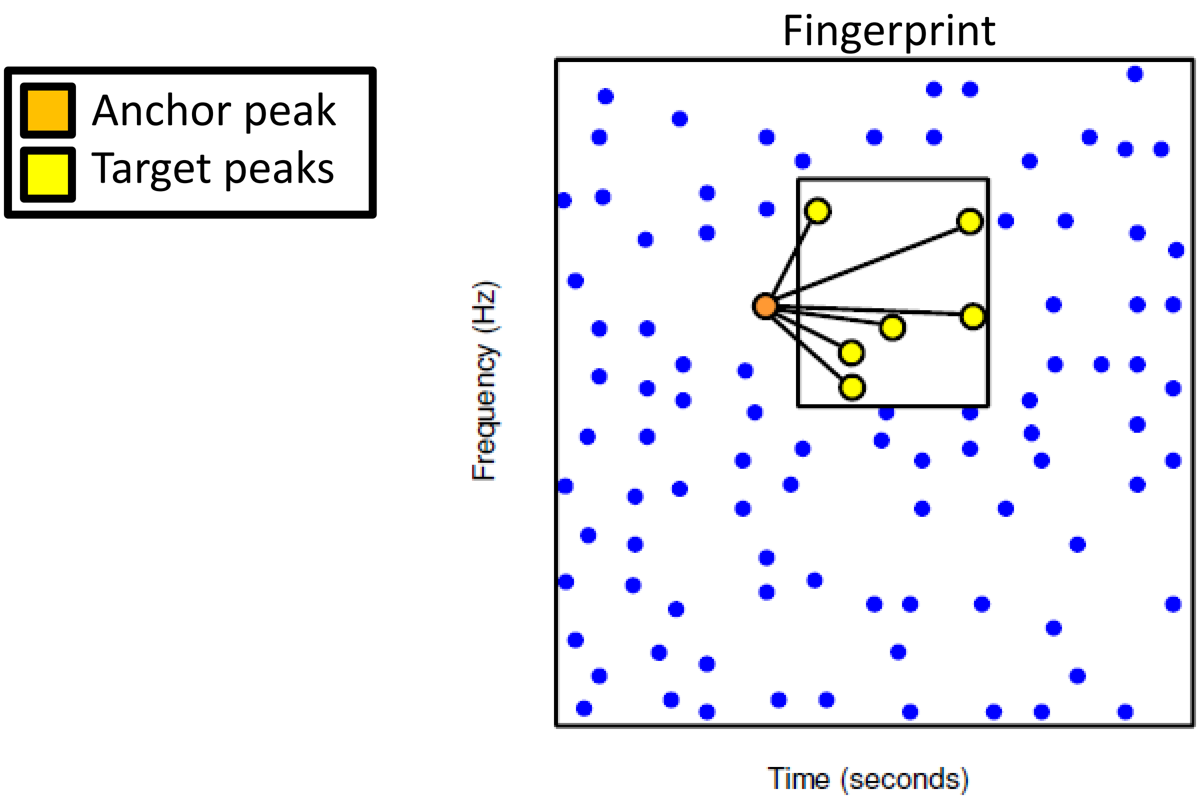
Indexing

- In practice, the fingerprints are encoded by using pairs of peaks to speed up the matching



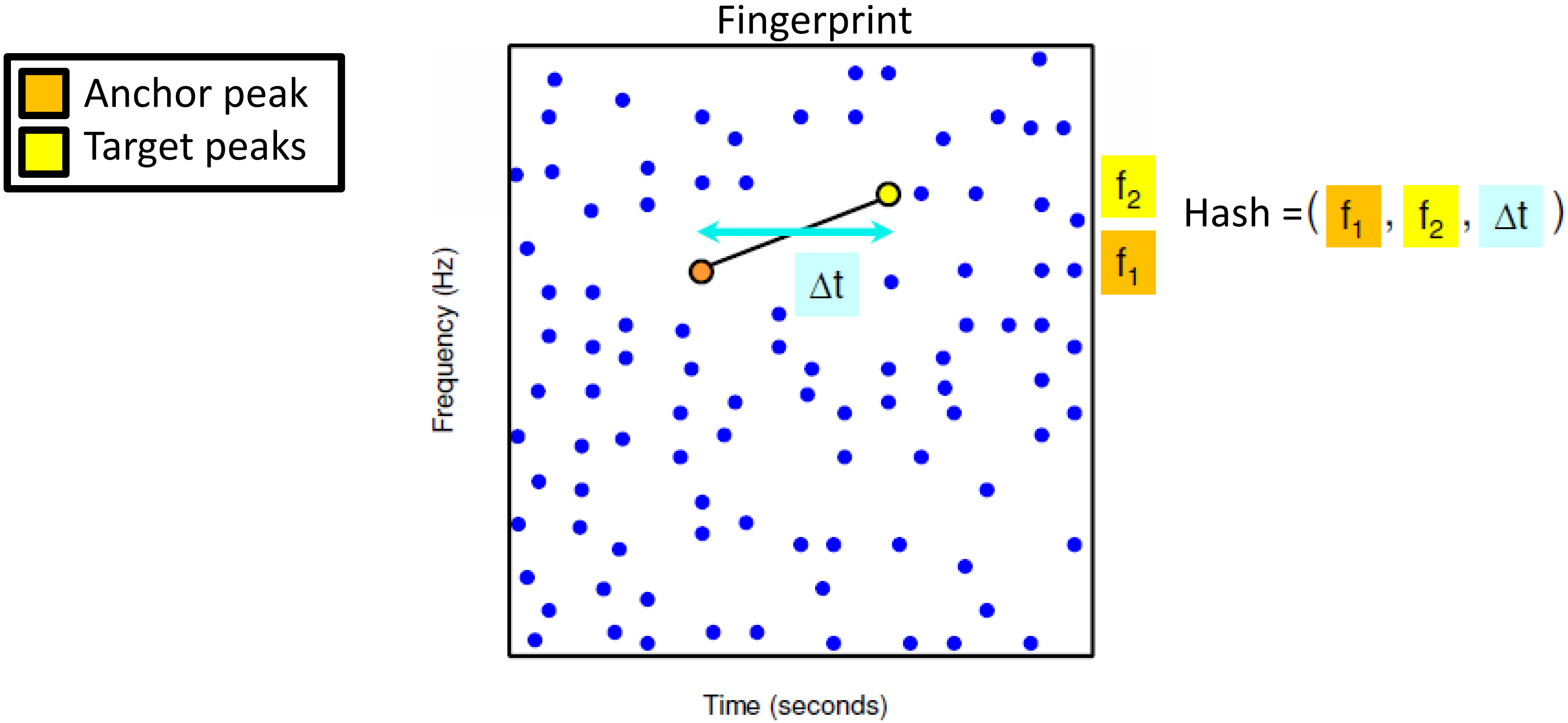
Indexing

- For every peak, pairs of peaks are formed by choosing an anchor point and a target zone



Indexing

- For every pair of peaks, a hash is formed using two frequency values and a time difference

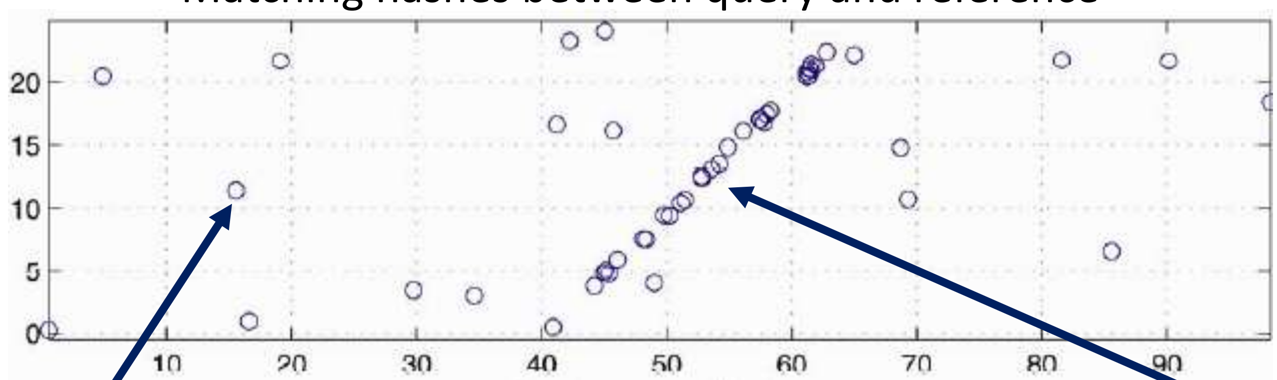


Indexing

- Hashes from a query are compared to hashes from every reference, given their offset times

Matching hashes between query and reference

Time for query



Time for reference

Isolated match between
a query hash and
a reference hash

Cluster of matches between
the query hashes and
the reference hashes

Outline

- Introduction
  - Context, literature, etc.
- Shazam
  - Fingerprinting, matching, etc.
- **Philips**
  - **Fingerprinting, matching, etc.**
- Conclusion
  - Advantages, limitations, etc.

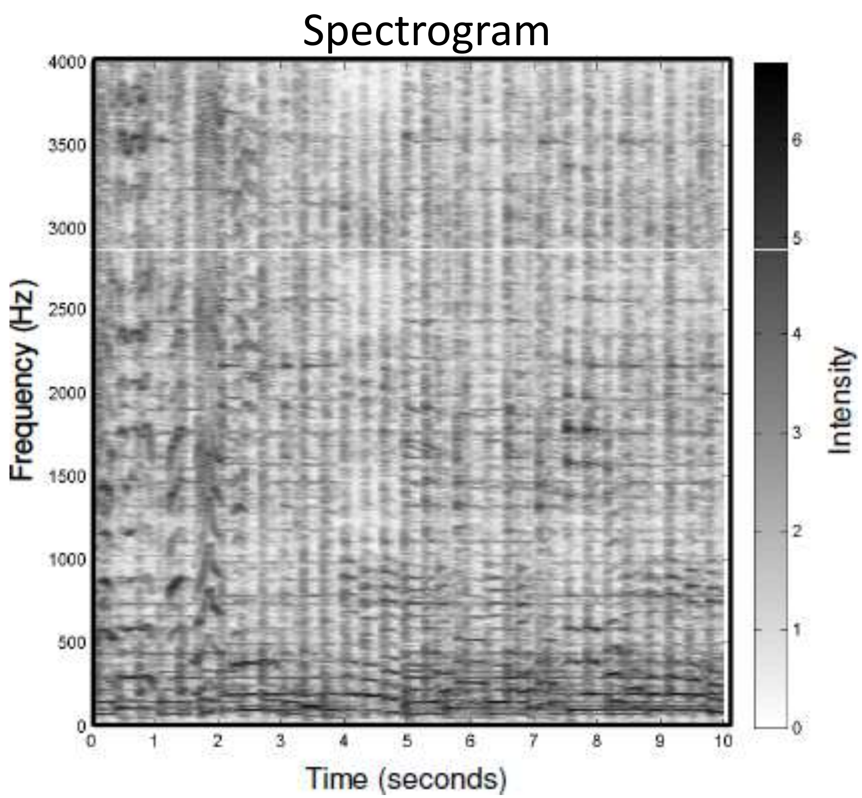
Philips

- Background
  - Based on the work of Jaap Haitsma and Ton Kalker
  - Technology sold to Gracenote, Inc. in 2005
  - Not (really) commercialized (yet)

www.gracenote.com

Fingerprinting

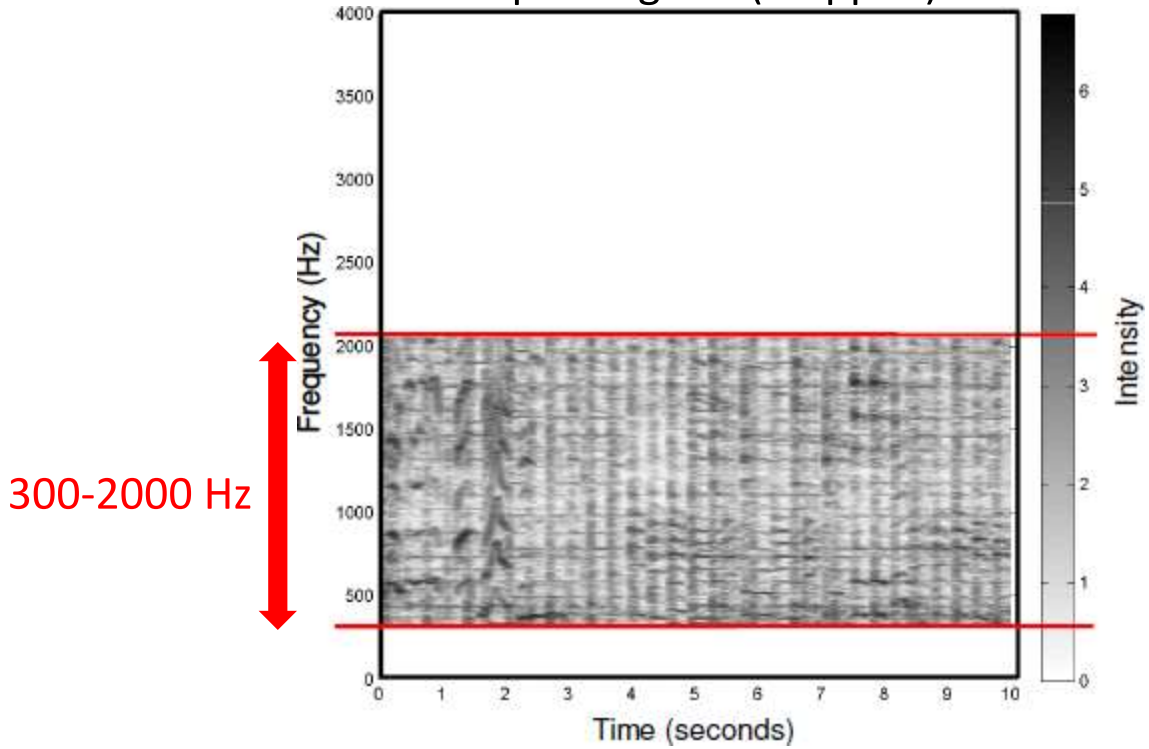
- The audio signal (e.g., a song) is first transformed into a spectrogram



Fingerprinting

- A perceptually relevant frequency range is selected from the spectrogram (300-2000 Hz)

Spectrogram (cropped)

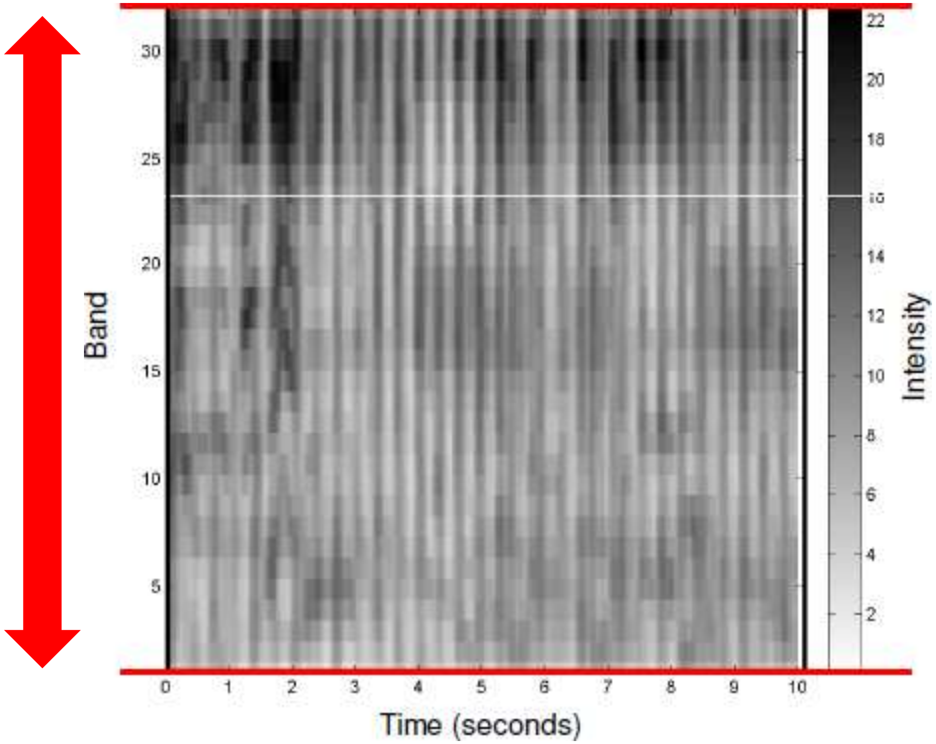


Fingerprinting

- 33 logarithmically-spaced frequency bands are extracted from that frequency range

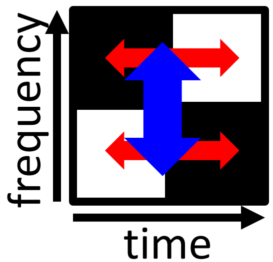
Spectrogram (log-scaled)

33 log-spaced frequency bands

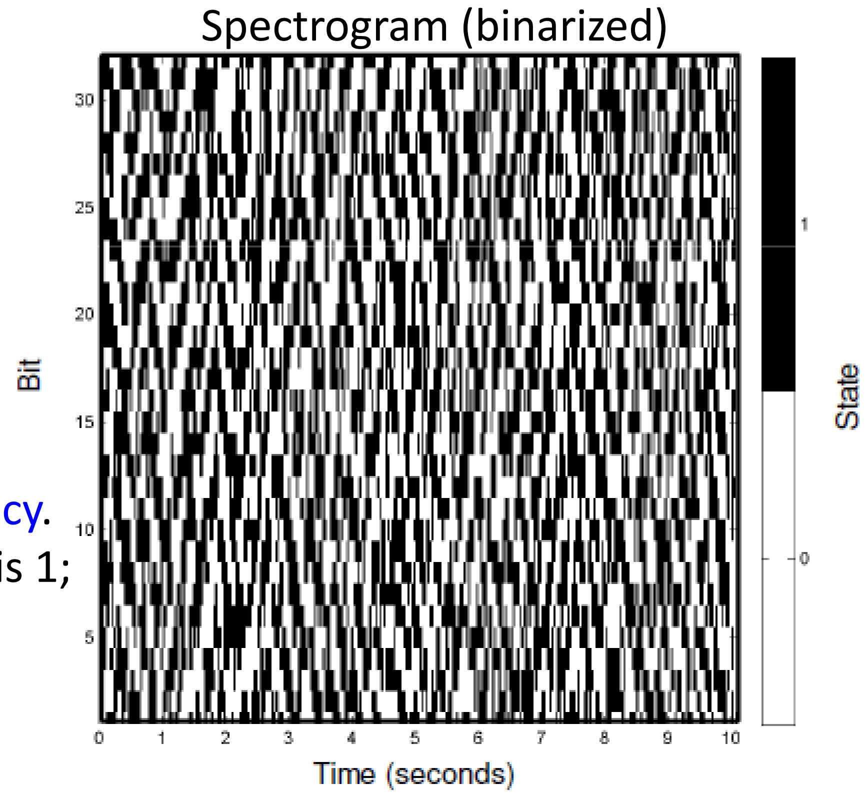


Fingerprinting

- The sign of the energy difference together along time and frequency is computed

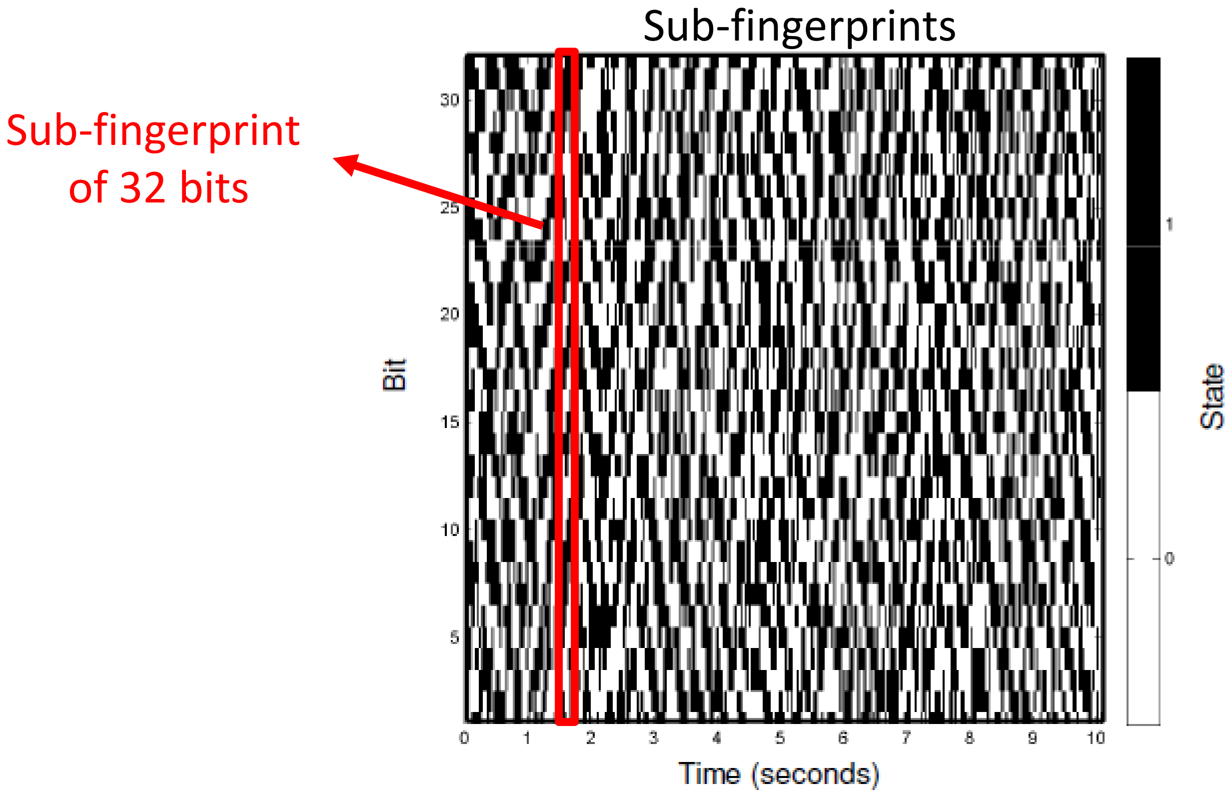


First, difference in time;
then, difference in frequency.
If result higher than 0, bin is 1;
otherwise, bin is 0.



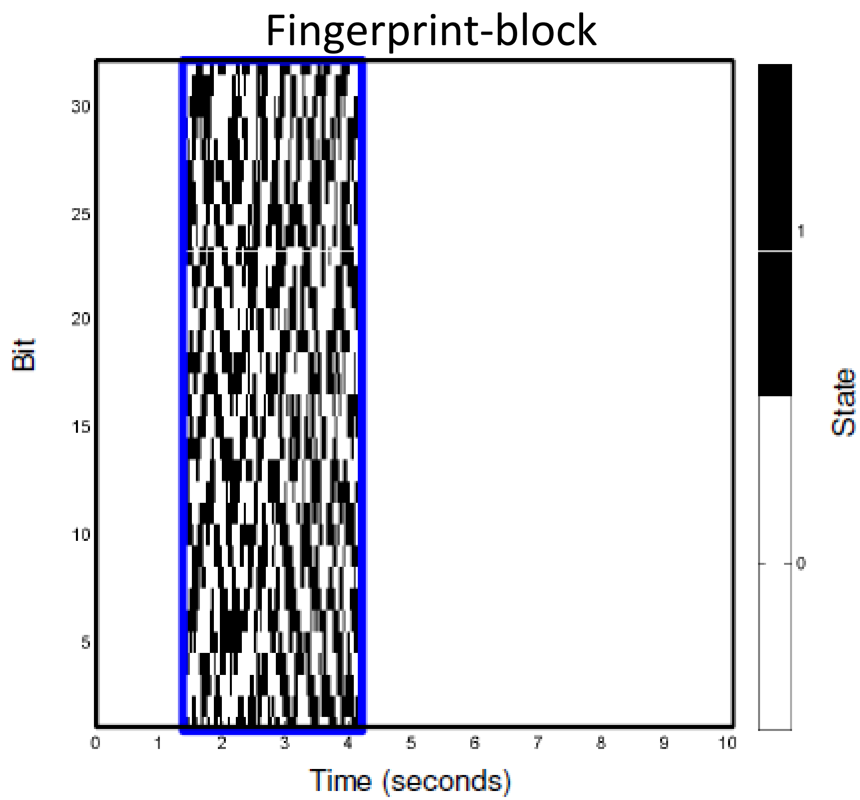
Fingerprinting

- This leads to a sub-fingerprint of 32 bits for every time frame in the spectrogram



Fingerprinting

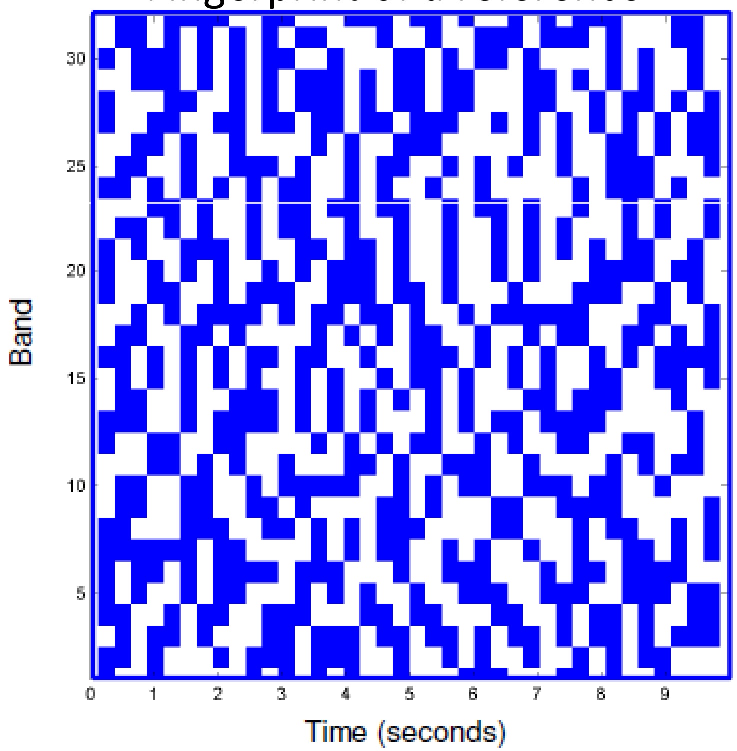
- A fingerprint-block is derived by grouping 256 successive sub-fingerprints (= 3 seconds)



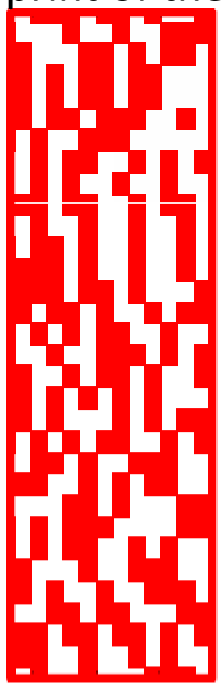
Matching

- A fingerprint is extracted from the query and compared to the fingerprints of the references

Fingerprint of a reference

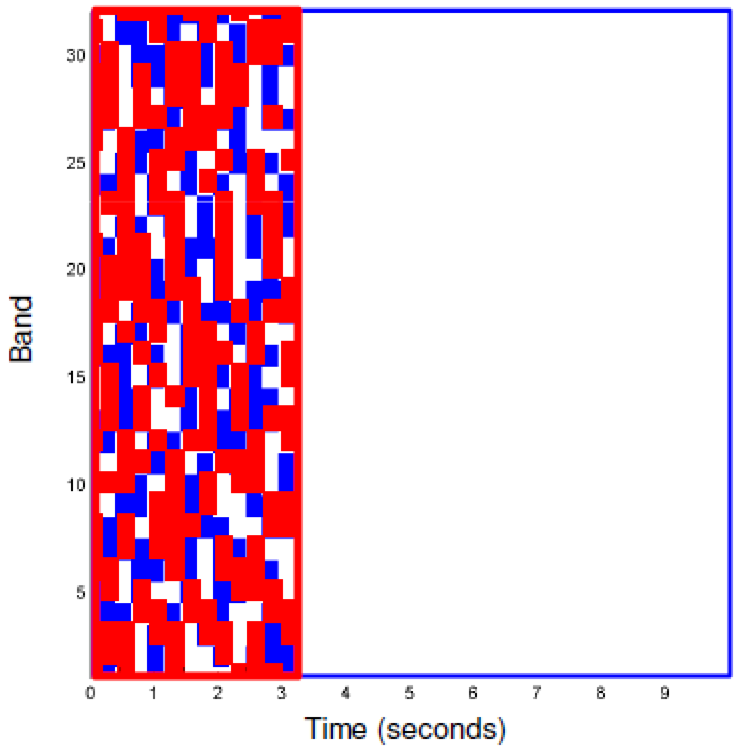


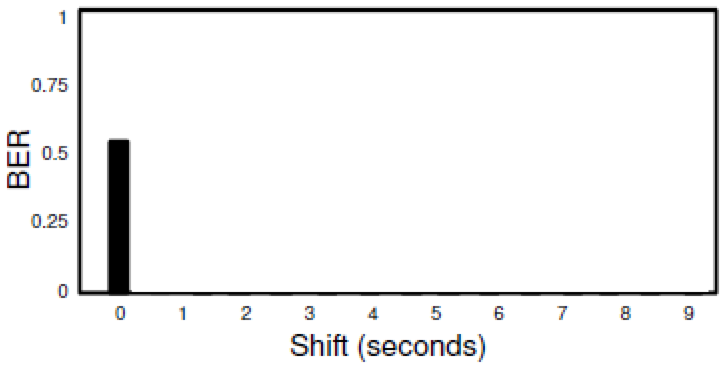
Fingerprint of the query



Matching

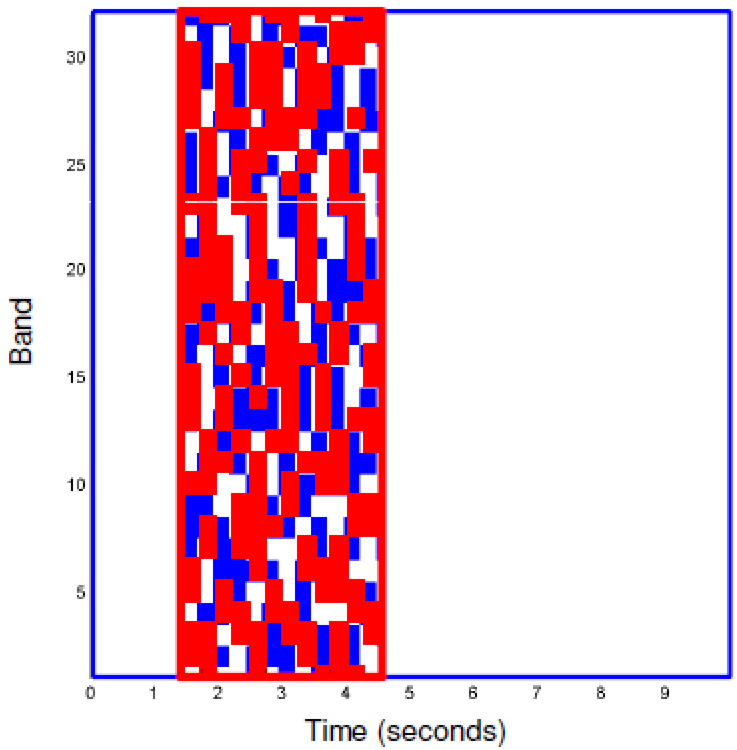
- The query fingerprint-block is shifted along time against every reference fingerprint-block

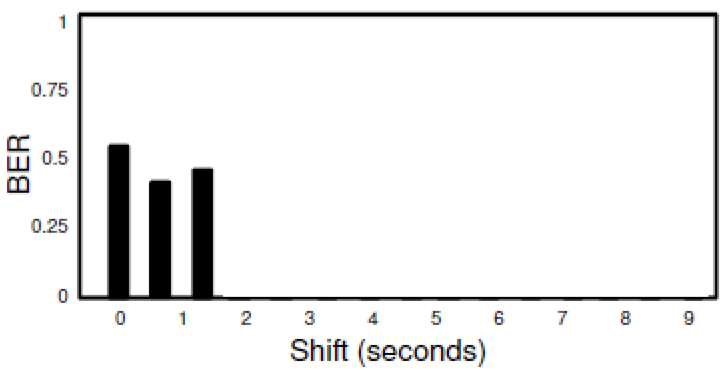




Matching

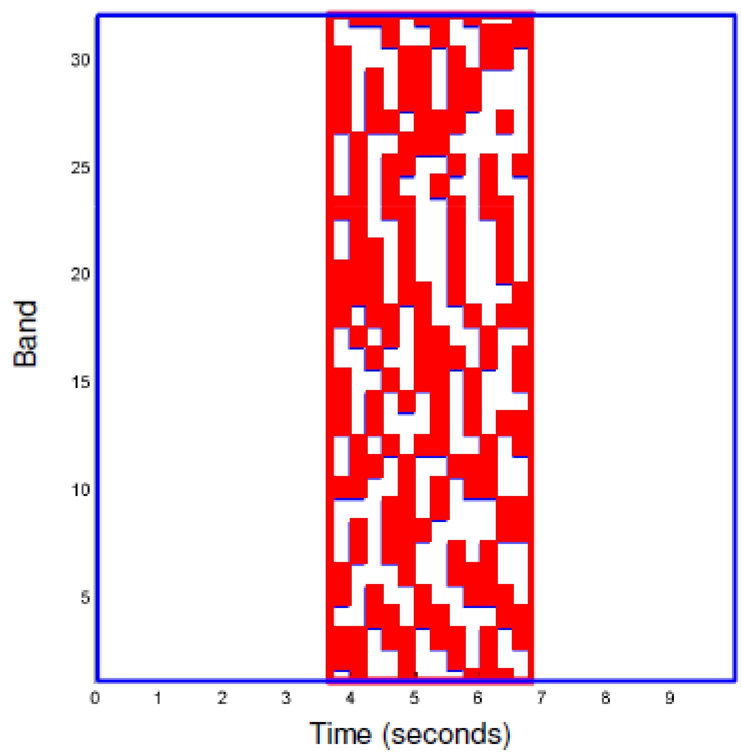
- The Bit Error Rate (BER) (% non-matching bits) is computed and saved for every possible shift

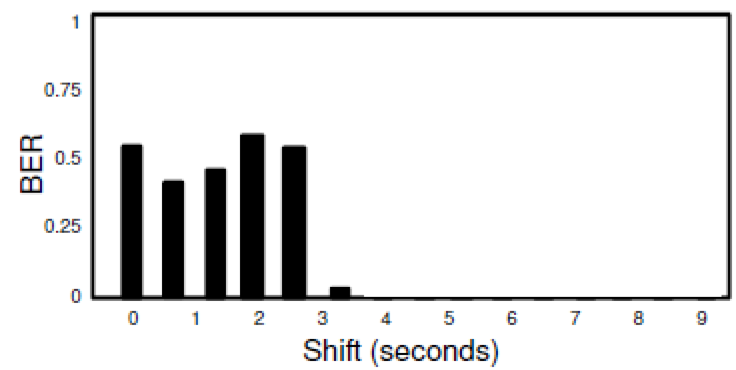




Matching

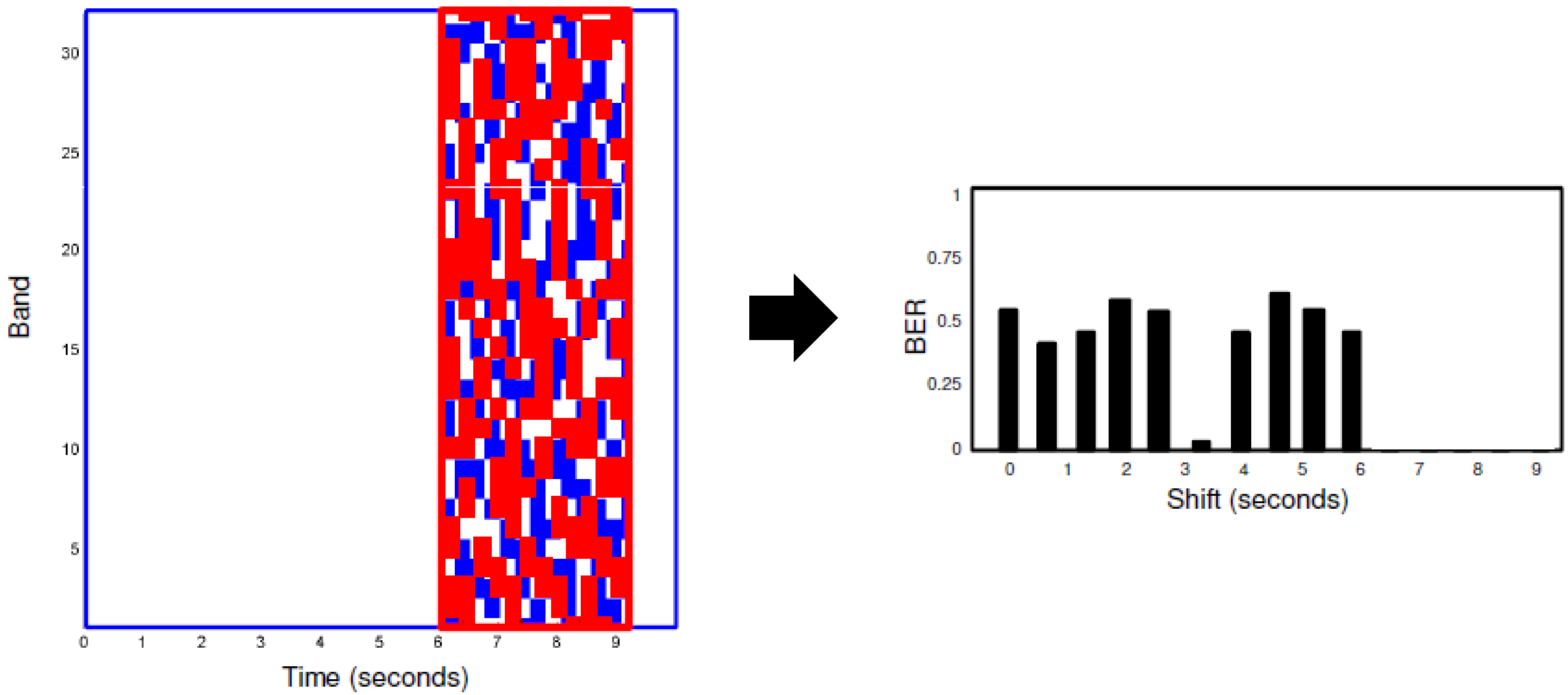
- The Bit Error Rate (BER) (% non-matching bits) is computed and saved for every possible shift





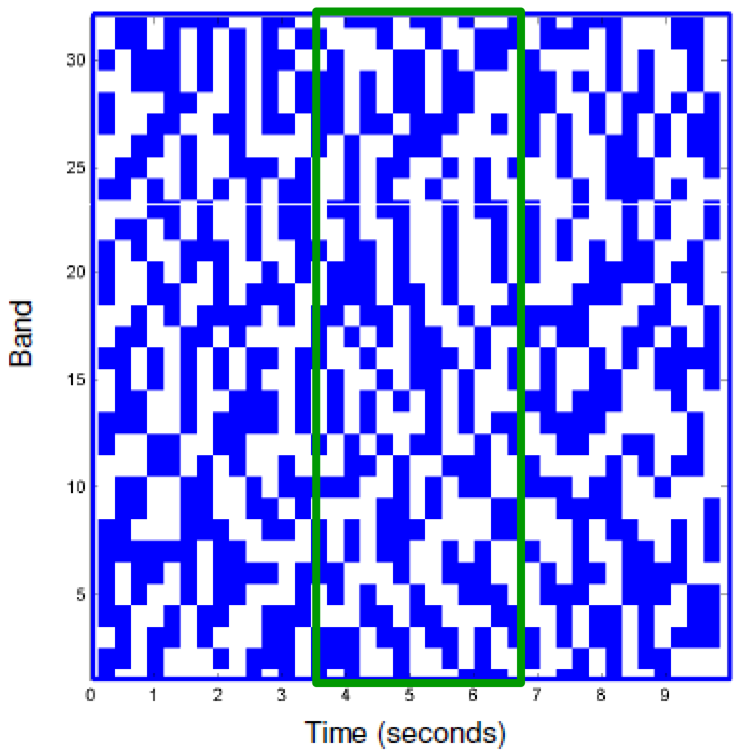
Matching

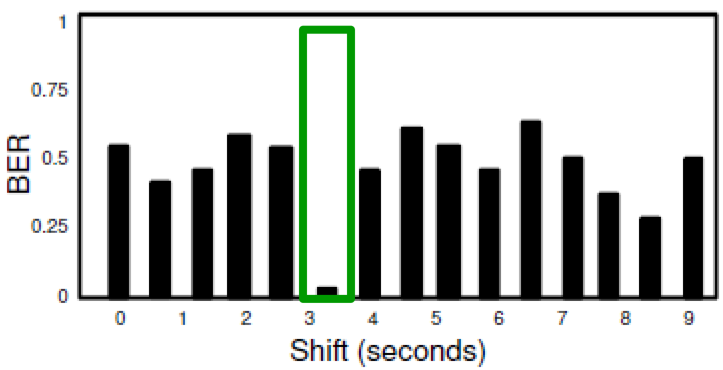
- The Bit Error Rate (BER) (% non-matching bits) is computed and saved for every possible shift



Matching

- A low BER indicates a match, and the corresponding reference is identified

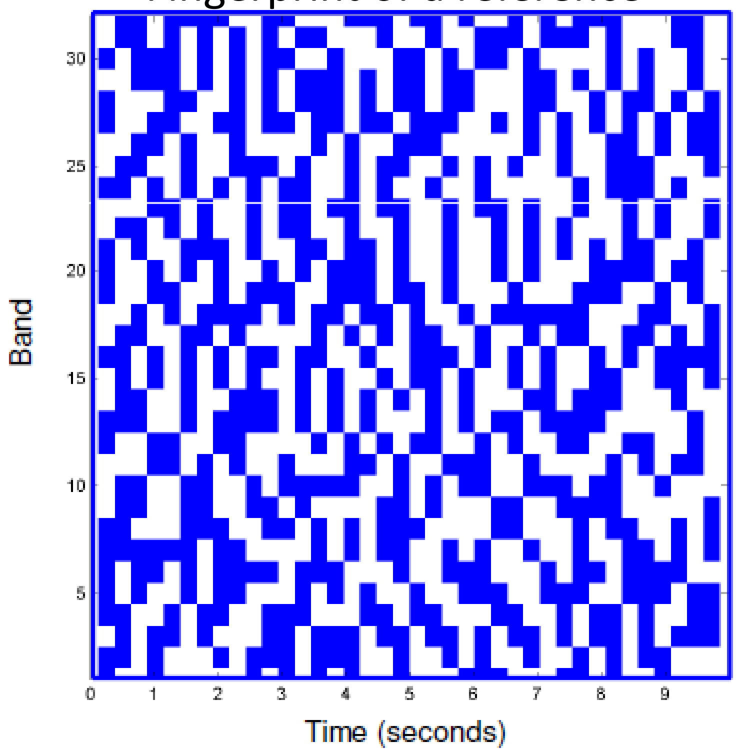




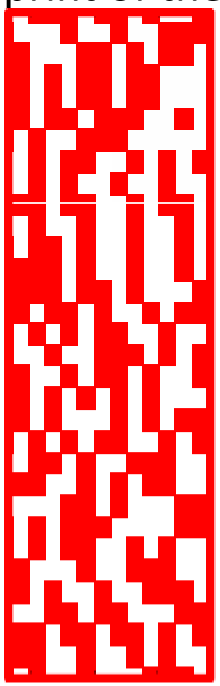
Indexing

- In practice sub-fingerprints are encoded using hashing to speed up the matching

Fingerprint of a reference



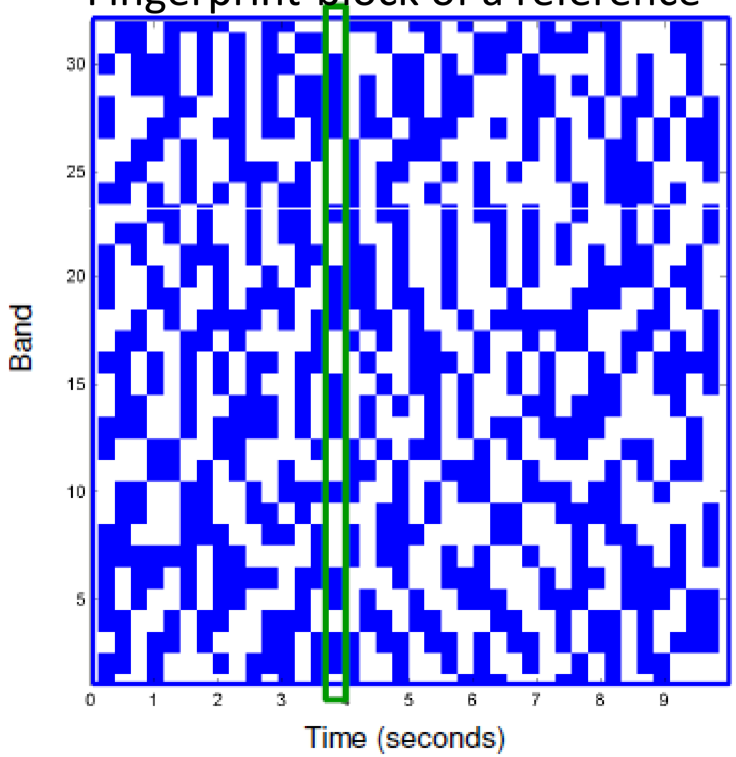
Fingerprint of the query



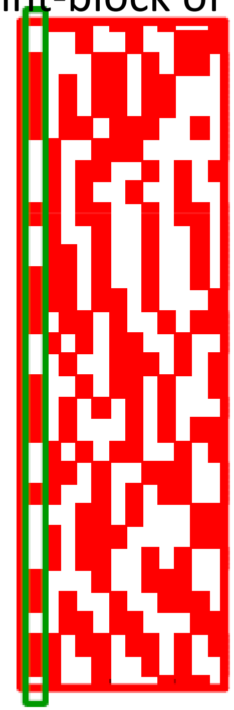
Indexing

- Exact sub-fingerprint matches are used to identify candidate reference fingerprint-blocks

Fingerprint-block of a reference



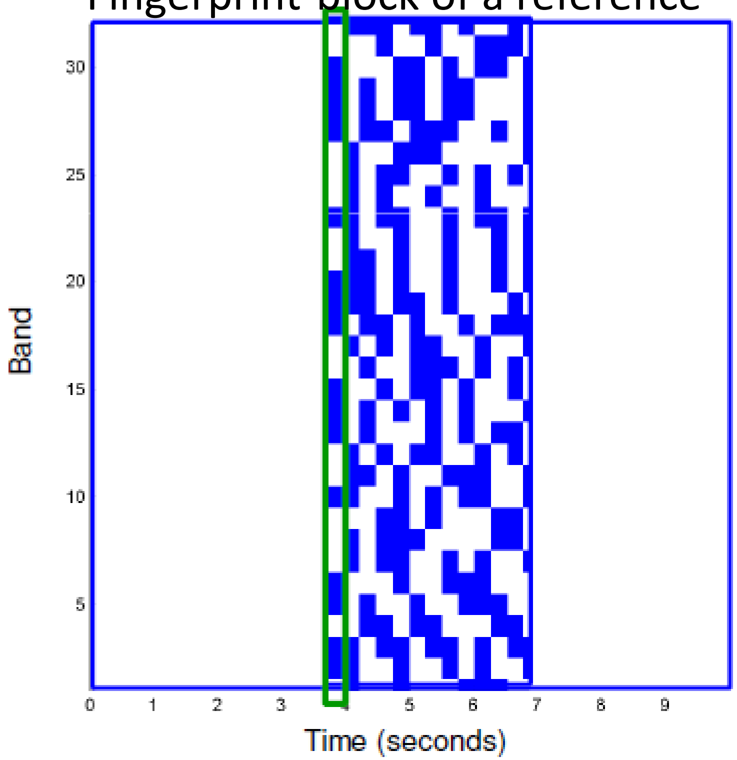
Fingerprint-block of the query



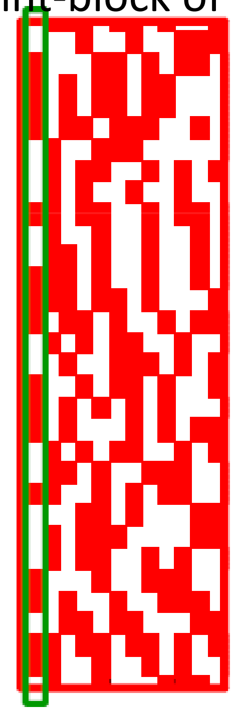
Indexing

- BER is computed only for the candidate reference fingerprint-blocks

Fingerprint-block of a reference

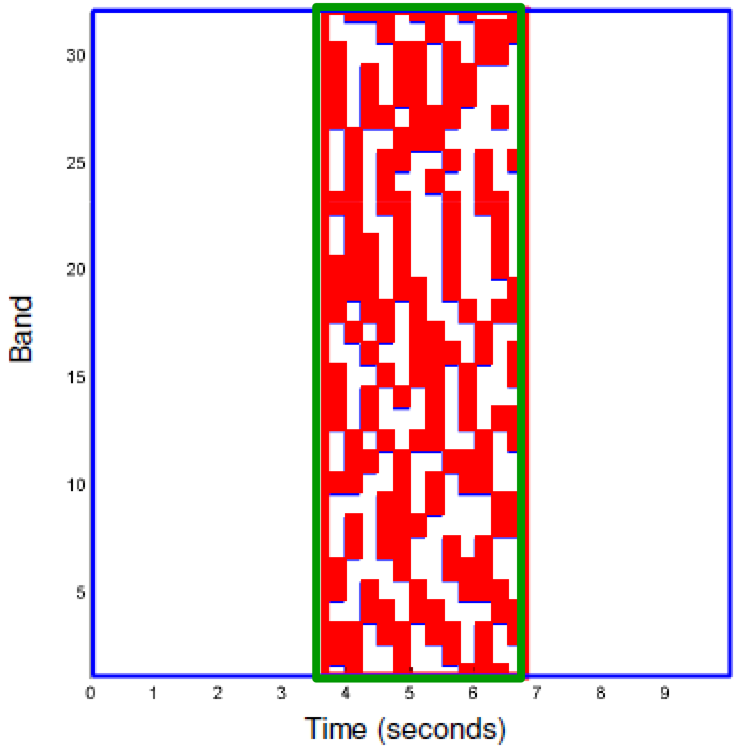


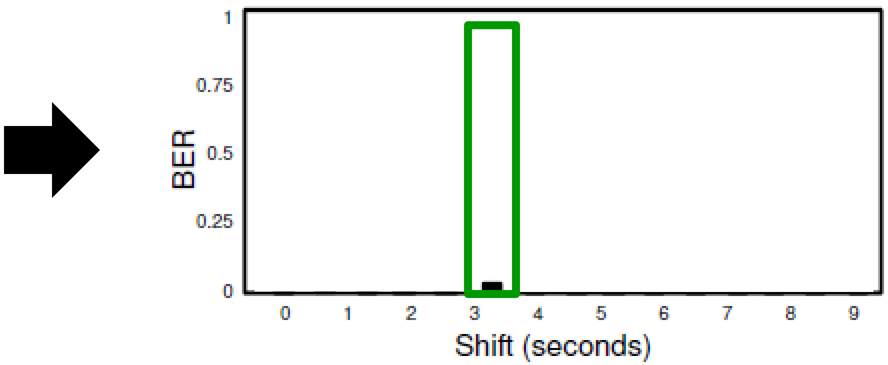
Fingerprint-block of the query



Indexing

- A match is identified when BER falls below a certain threshold





Outline

- Introduction
  - Context, literature, etc.
- Shazam
  - Fingerprinting, matching, etc.
- Philips
  - Fingerprinting, matching, etc.
- **Conclusion**
  - **Advantages, limitations, etc.**

Advantages

- Audio identification systems
  - Robust to distortion and noise
  - Short queries (3-10 seconds)
  - Fast matching (3-10 seconds)

Limitations

- Needs exact same rendition!
  - No live version (different key or tempo)
  - No cover version (different instruments)
  - No hummed version (single melody)

Solutions

- Fingerprints robust to key or tempo deviations
  - Log-frequency spectrogram for pitch shifting
  - Fingerprint invariant to time-scaling
  - Etc.

Solutions

- Cover identification
  - Chromagram to handle key/instrument variations
  - Sequence alignment to handle tempo variations
  - Etc.

Solutions

- Query-by-humming
  - Relative pitch intervals to handle key deviations
  - Relative length ratios to handle tempo deviations
  - Etc.

References

- Shumeet Baluja and Michele Covell, “Audio Fingerprinting: Combining Computer Vision & Data Stream Processing,” 32nd International Conference on Acoustics, Speech and Signal Processing, Honolulu, HI, USA, April 15-20 2007, pp. II–213 – II–216.
- Christopher J. C. Burges, John C. Platt, and Soumya Jana, “Distortion Discriminant Analysis for Audio Fingerprinting,” *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 11, no. 3, pp. 165–174, May 2003.
- Pedro Cano, Eloi Batlle, Ton Kalker, and Jaap Haitsma, “A Review of Audio Fingerprinting,” *Journal of VLSI Signal Processing Systems*, vol. 41, no. 3, pp. 271–284, November 2005.
- Peter Grosche, Meinard Müller, and Joan Serrà, “Audio Content-based Music Retrieval,” *Multimodal Music Processing,* Meinard Müller, Masataka Goto, and Markus Schedl, Eds, vol. 3 of *Dagstuhl Follow-Ups*, chapter 9, pp. 157-174. Dagstuhl Publishing, Wadern, Germany, April 2012.
- Jaap Haitsma and Ton Kalker, “A Highly Robust Audio Fingerprinting System,” *3rd International Conference on Music Information Retrieval*, Paris, France, October 13-17 2002, pp. 107–115.
- Avery Li-Chun Wang, “An Industrial-strength Audio Search Algorithm,” *4th International Conference on Music Information Retrieval*, Baltimore, MD, USA, October 26-30 2003, pp. 7–13.