ISMIR 2003

October 29th, 2003

if it sounds good, tag it

What is Shazam?

What is Shazam?

- Query by mobile phone
- Started in Year 2000
- Headquartered in London
- Launched Service in August of 2002
- 1.8M+ tracks
- Service live in UK, Gemany, Finland
- Coming soon to other countries in Europe and Asia

Shazam Connects you to Music

I love that song!

if it sounds good, tag it

Everywhere you have your mobile

"THE MOMENT"



Radio - Car, Home, Work

TV and Cinema

Clubs and Bars

Cafes, Shops, Restaurants



if it sounds good, tag it

Target Audience

Segmentation

Appeal

Core target: Music mobile

- 18-25 years old
- Struggle to keep up with *LATEST RELEASES*
- Enjoy new technologies

Music 'Experts'

Early Youth

- 14-17 years old
- Identify next purchase quickly
- Enjoy practical services

Music Community

More Mature

- 26-40 years old
- Identify classic hits as well as new music
- Need advice on what to buy

Music Confidence

User Experience

Shazam allows people to identify music over the mobile phone, anywhere and anytime.

- ✓ **Dial 2580 & let the phone listen to the music.**
- ✓ **Shazam will terminate the call and send an SMS back with the name of the track & artist – this is called tagging.**
- ✓ **Access further content – Ringtones, Songmail..**
- ✓ **List of tagged songs available on http://www.shazam.com**

Access your “tags”

- Track name, artist and album are currently displayed
- Shazam has more music data than currently used, prioritization will depend on consumer feedback and product roadmap
- Tags can be sorted in various ways
- User can buy CDs from a variety of online stores

And more...

Operating Constraints

Audio Source Constraints

- Imperfect audio source material
  - Physical media defects
  - Digital compression
  - Watermarks
- Imperfect audio equipment
  - Speed variation (turntables and drive mechanisms)
  - Poor speakers
  - Nonlinear phase
- Environmental factors
  - Propagation through air
  - Reverberation
  - Additive noise

Receiver Constraints

- Poor microphone
- Bandlimited sampling (8KHz)
  - 300-3500Hz telephone bandwidth
- AGC, VAD, and Squelch
- Background noise suppression and nonlinear voice enhancement
- Voice Codec
  - EFR, AMR, EVRC, QCP, etc.
- Network dropout, poor coverage, handoff

Search Constraints

- Be insensitive to offset (e.g. not just first or middle 30 seconds)
- Must have high sensitivity in the presence of noise and distortion
- Low probability of false positives
  - Not just "closest match"
  - Slightly challenging with respect to certain kinds of music, such as techno
  - Plagarism

Search Constraints

- Identify exact recording
  - (for many applications: rights mgmt, etc)
- Scale to millions of tracks
  - Statistical scaling (maintain high sensitivity and low false positives)
  - Computational scaling (must be fast to serve hundreds or thousands of requests per second without requiring inordinate CPU power).
    - log speed or better
    - parallelizeable
  - Reasonably small memory footprint

This Problem is Impossible

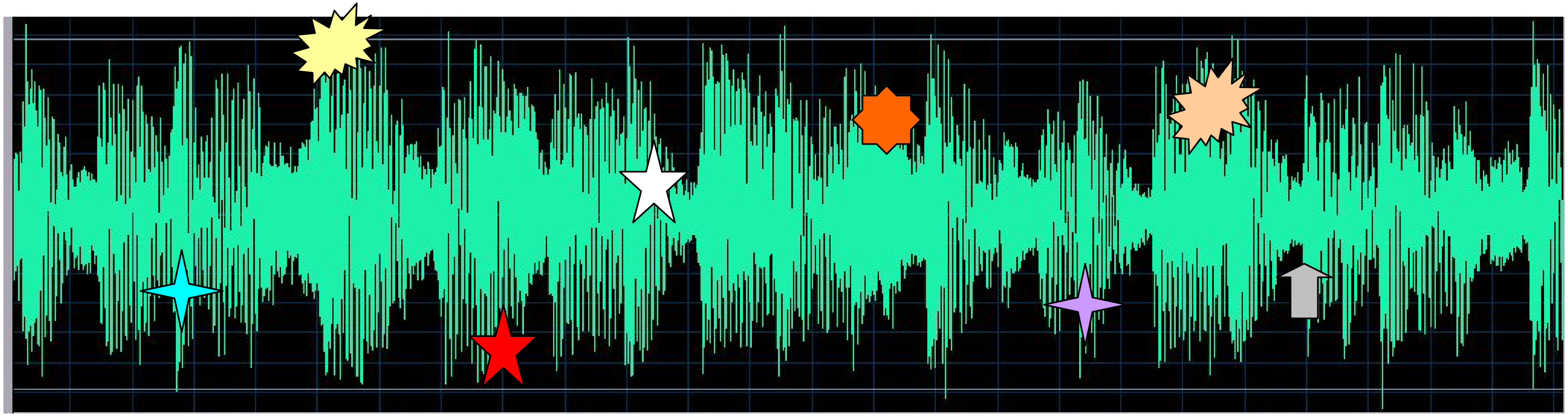
- A real-world sample:
- Extremely challenging, discouraging
- No known technique could work
- Break news gently to colleagues
- Find new job?
- But actually…

How does it work?

Desired “Fingerprint” and System Properties

- Survives all the obstacles going from source material to recording received at our IVR
- Mostly reproducible, even in presence of noise
- Informative (reasonably high entropy)
- Tolerates shredded or partially missing features
- Tolerates spurious features
- Translation invariant
- Self-framing

Aligned Tagged Landmarks



- For each audio file, generate reproducible landmarks
  - – Each landmark occurs at a time offset
- For each landmark, generate a “fingerprint” tag that characterizes its location

Aligned Tagged Landmarks

- Do same for sample
- Generate list of matching fingerprints
- Each correctly matching fingerprint must have same relative time offset

$$time_{db} - time_{sample} = Constant$$

- Incorrectly matching fingerprints have random relative time offset
- Filter out cruft by doing a histogram on time differences!
- Score is size of biggest histogram peak

Non-matching: No alignment

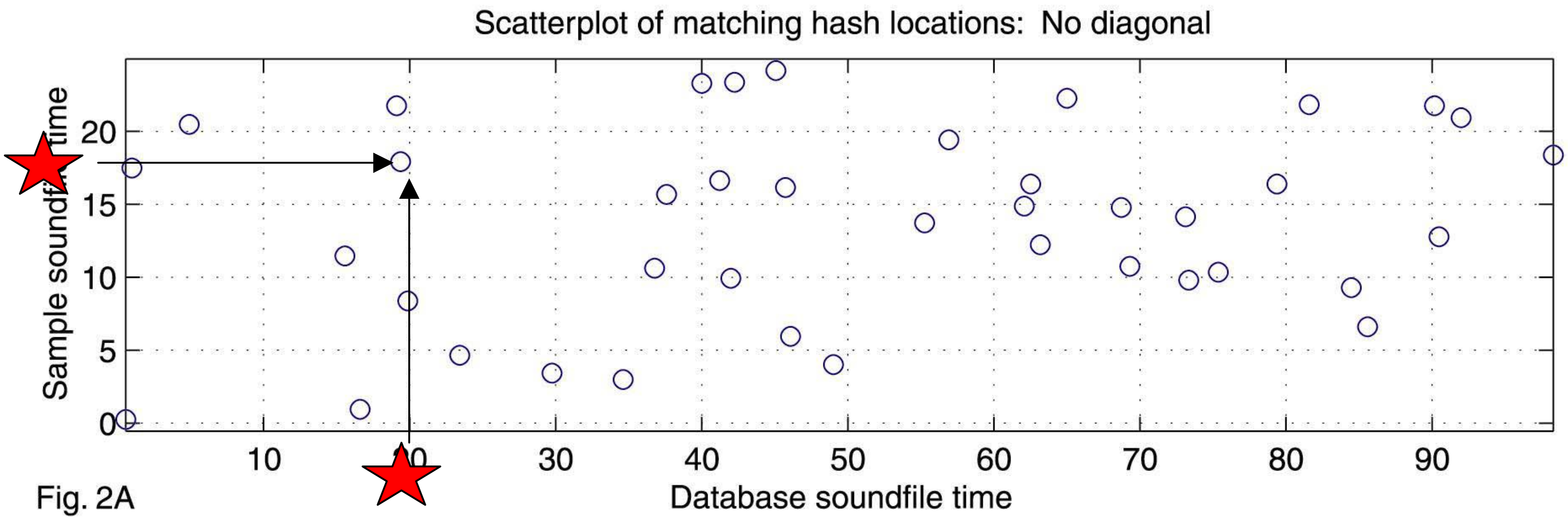


Fig. 2A

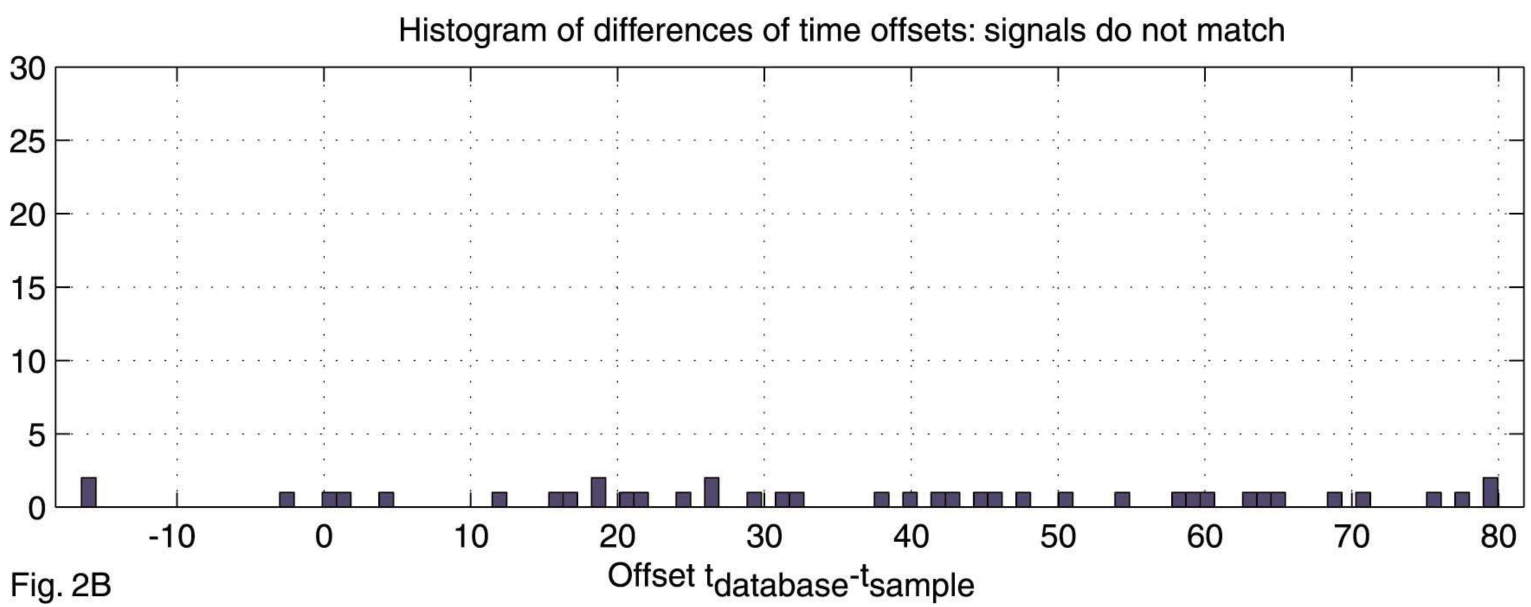


Fig. 2B

Matching: alignment

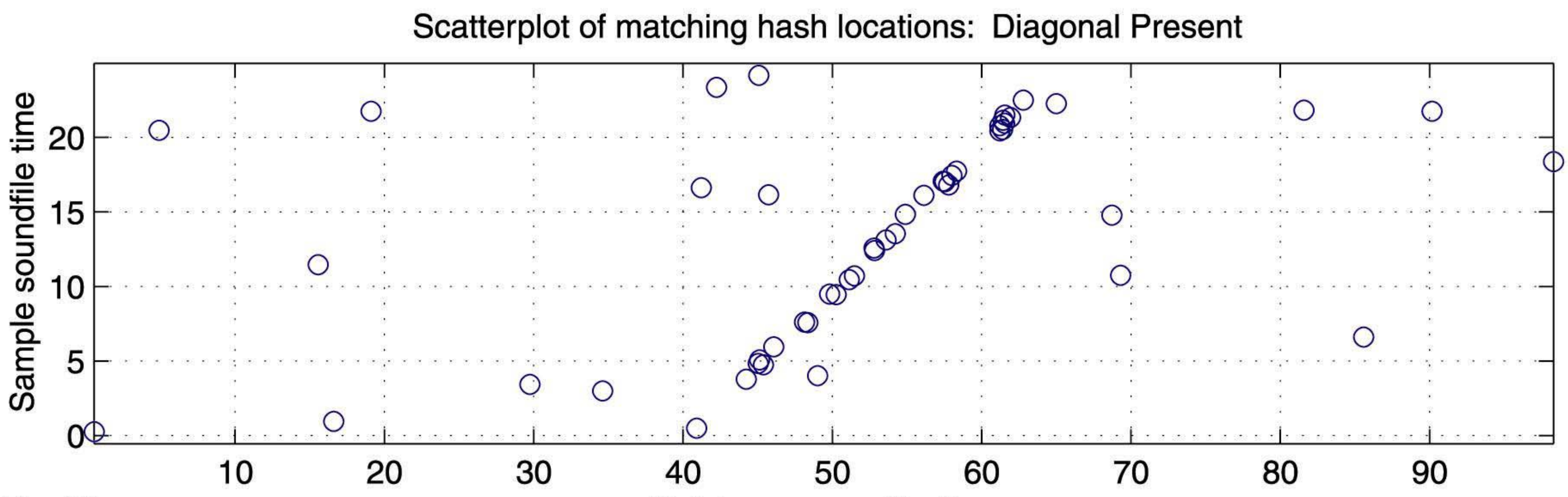


Fig. 3A

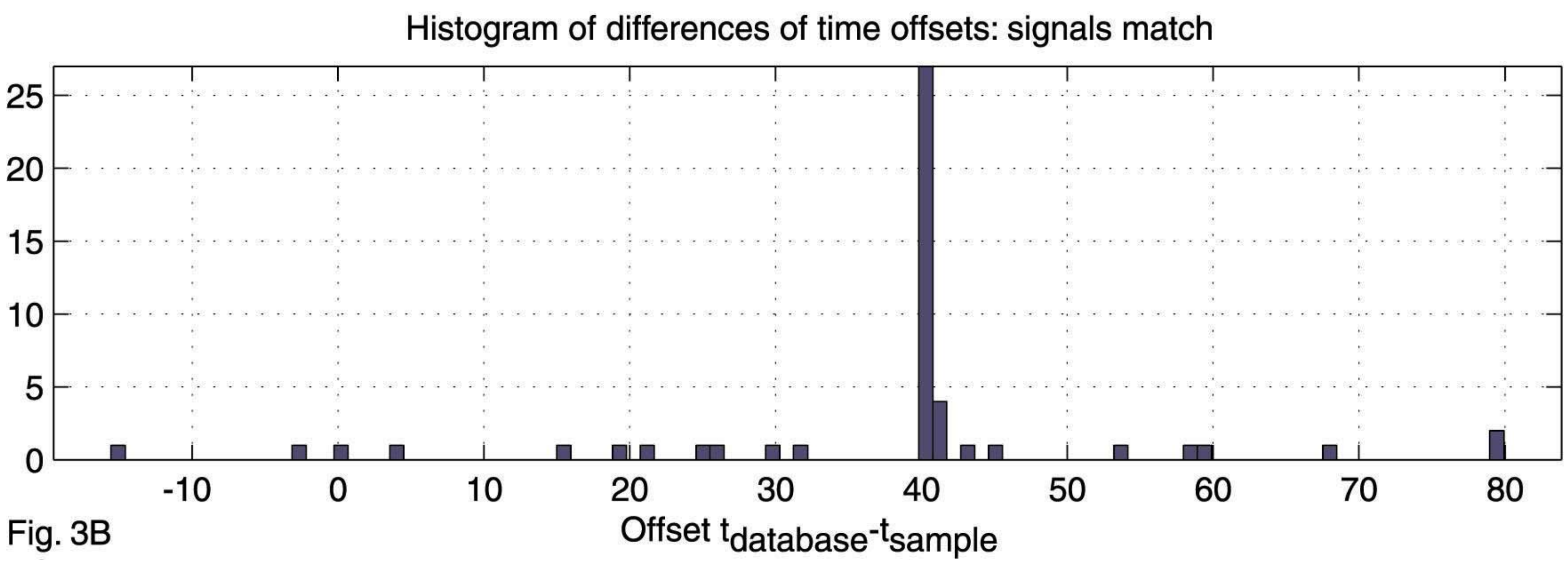
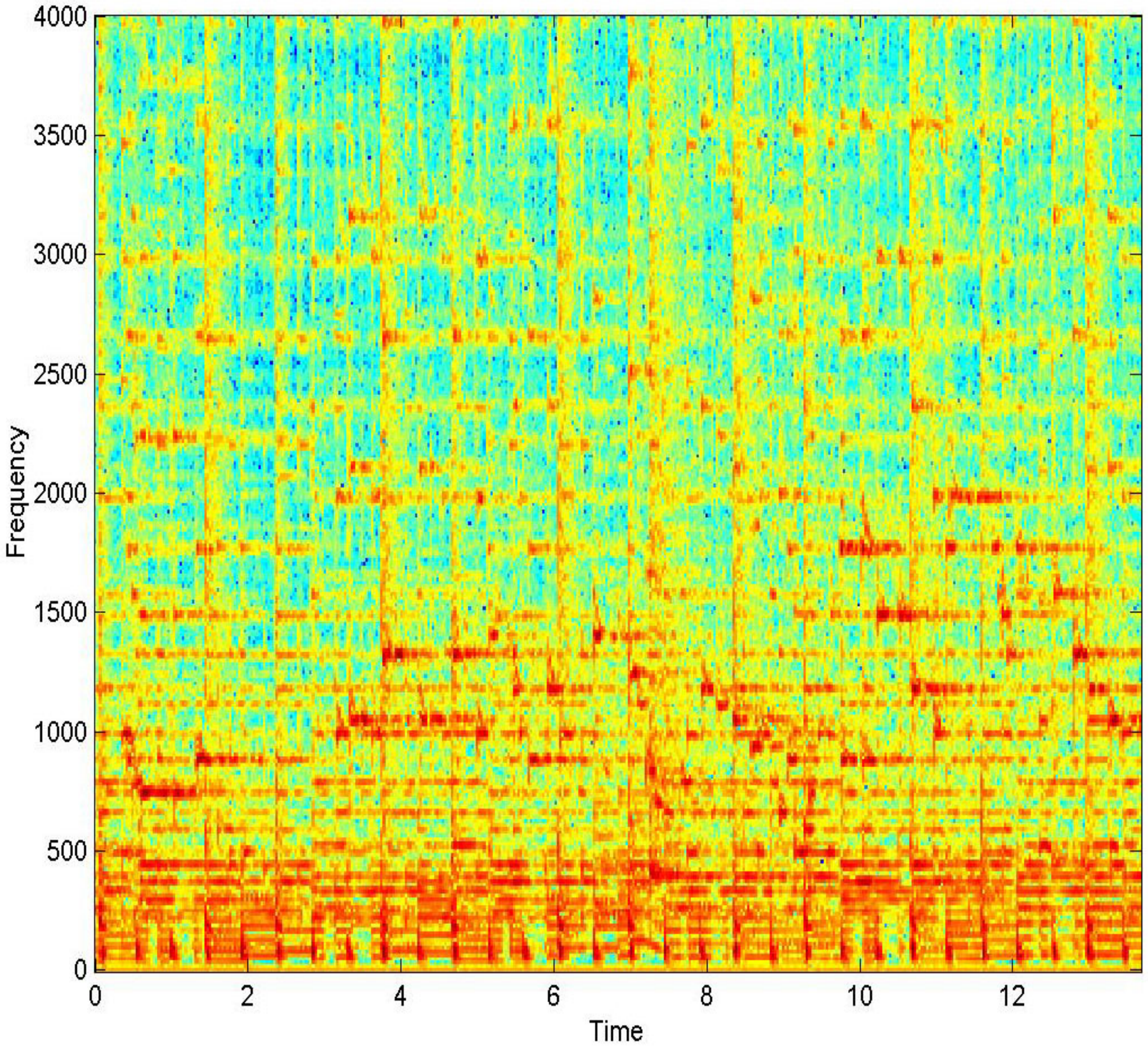


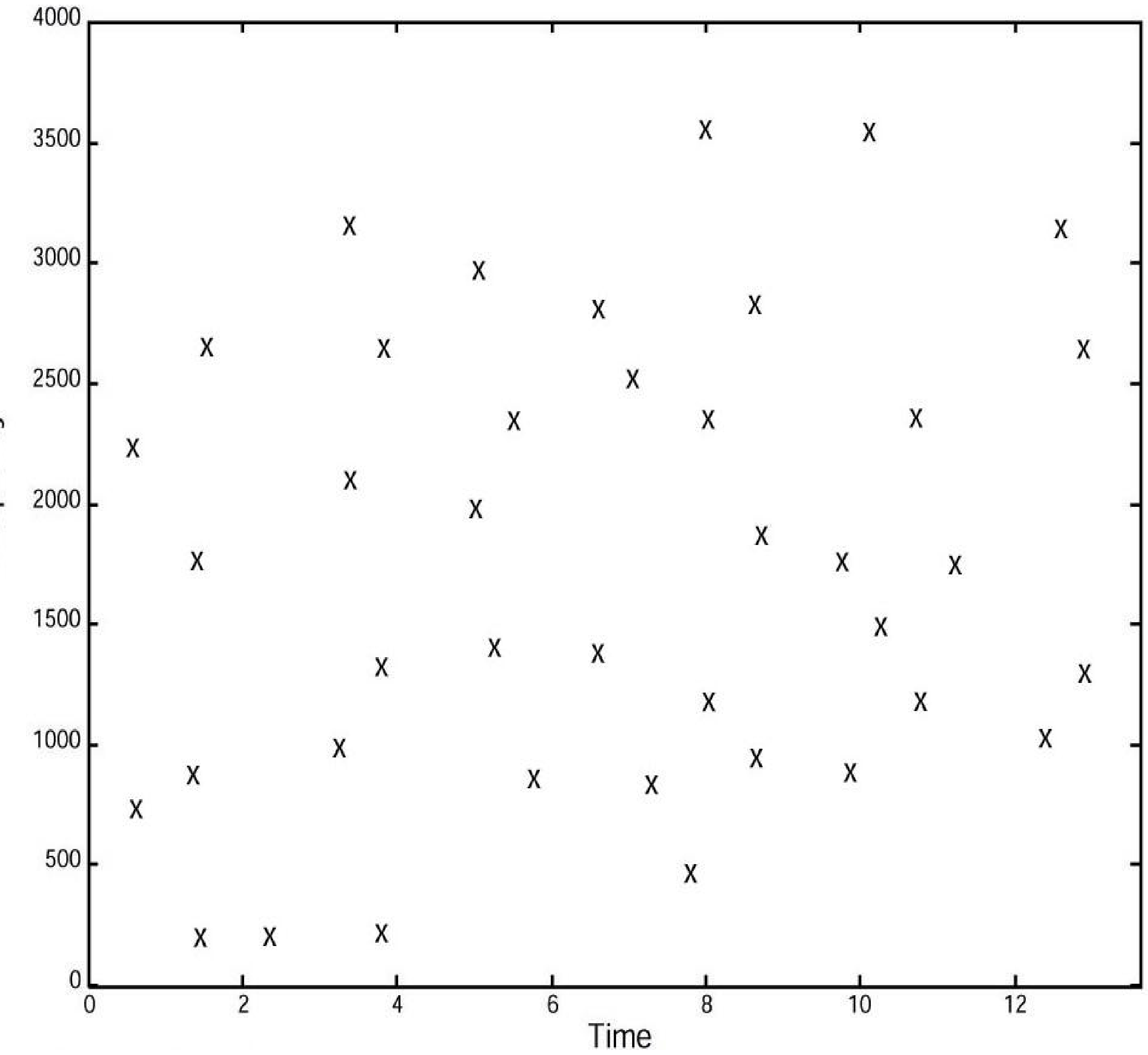
Fig. 3B

Spectrogram Peaks



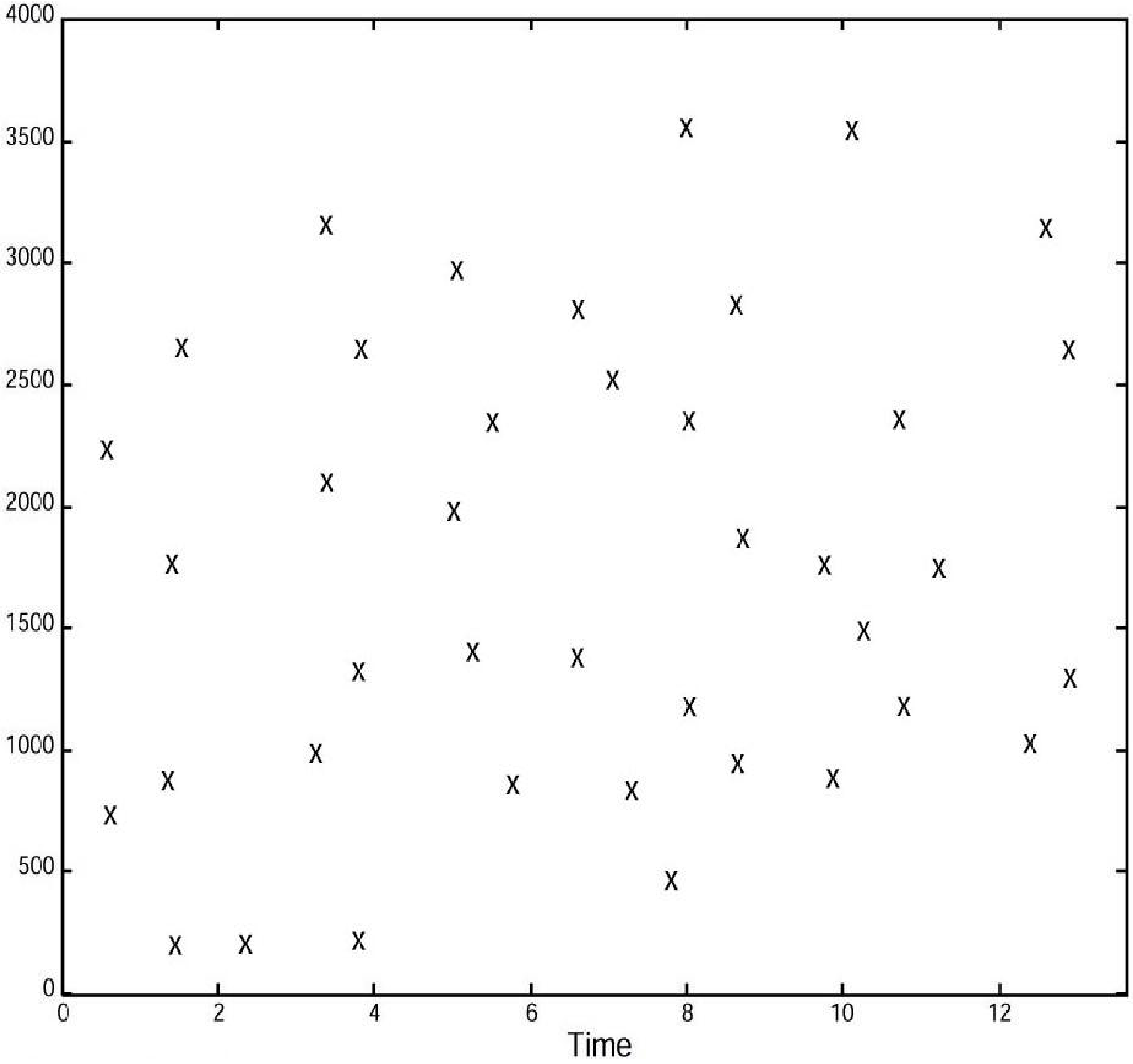
- Extremely robust
  - Against noise
  - Against reverb, room dynamics
  - Against nonlinear distortion
- Reproducible
  - Everything you want
- Tend to survive through voice codec

Spectrogram Peaks



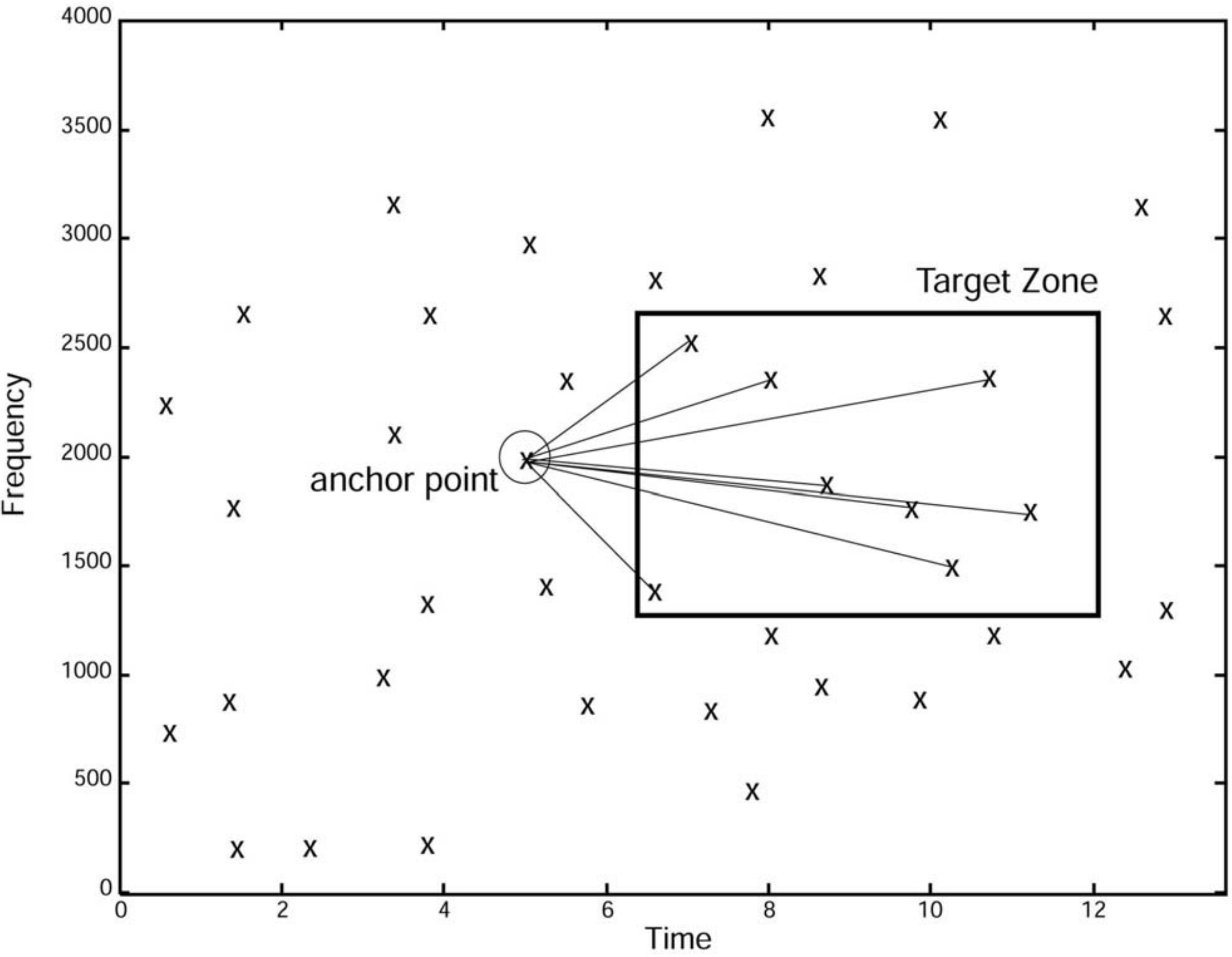
- So, we could let features be the peaks themselves:
  - Extract time-frequency coordinates as skeletonized “constellation map” of “landmarks”
  - Frequency value is “fingerprint”
  - “sliding transparency”

Spectrogram Peaks



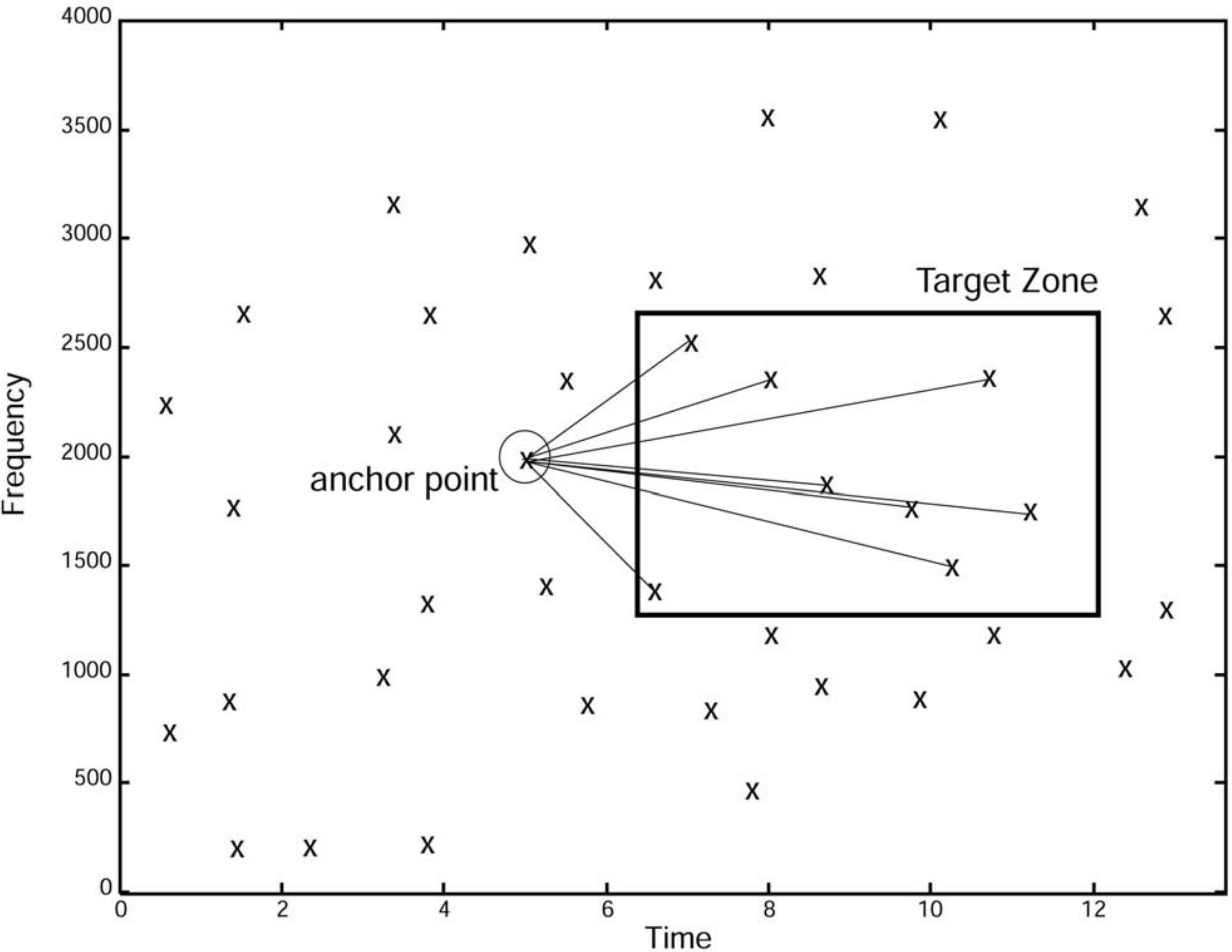
- However, this is a little slow since individual peaks have low entropy

Combinatorial Hashing



- Fix speed problem by increasing entropy of feature space
- Use combinations of a small number (2-3) of constellation points
- Each point is taken as an "anchor point"
- Each anchor point has a "target zone"

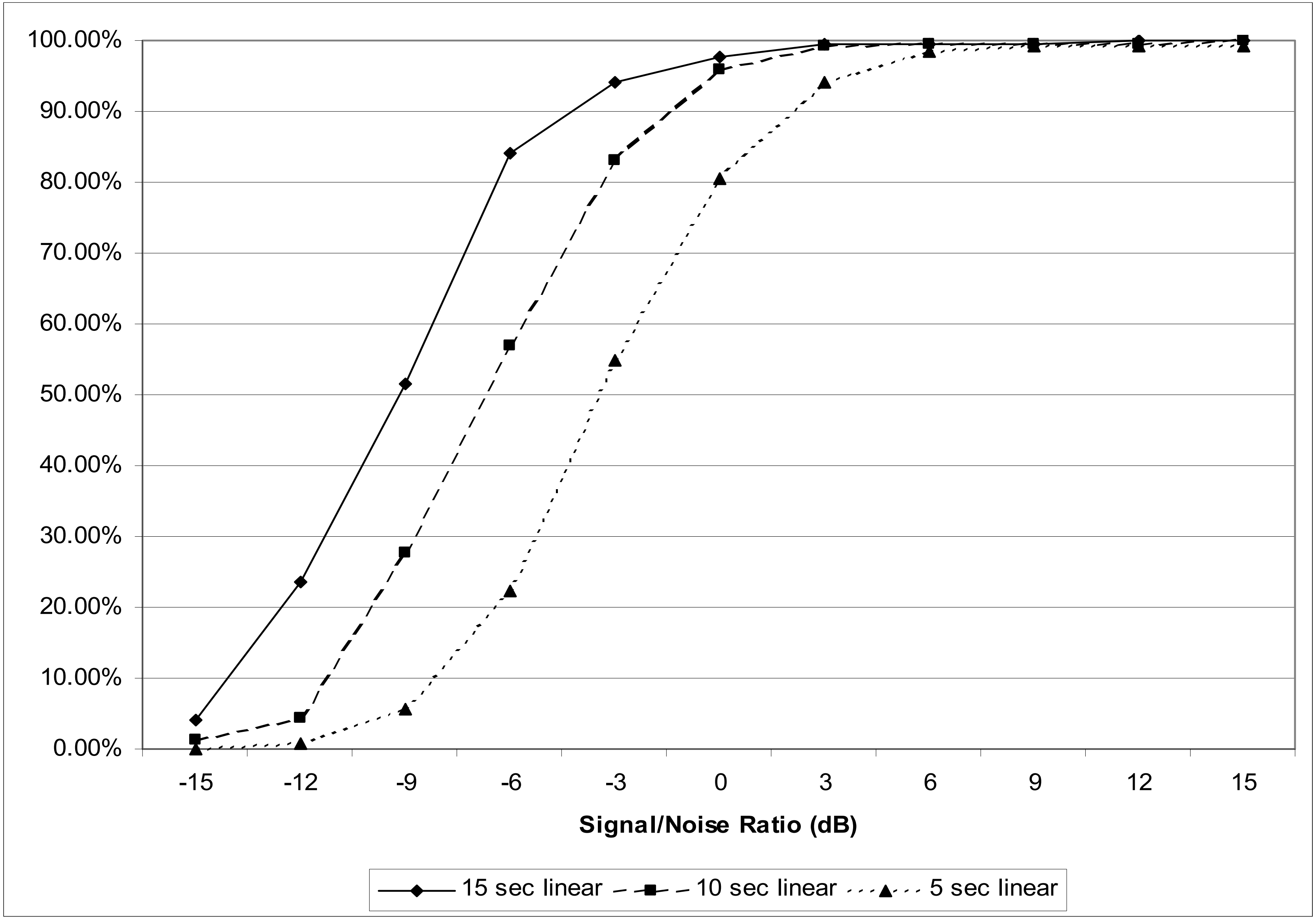
Combinatorial Hashing



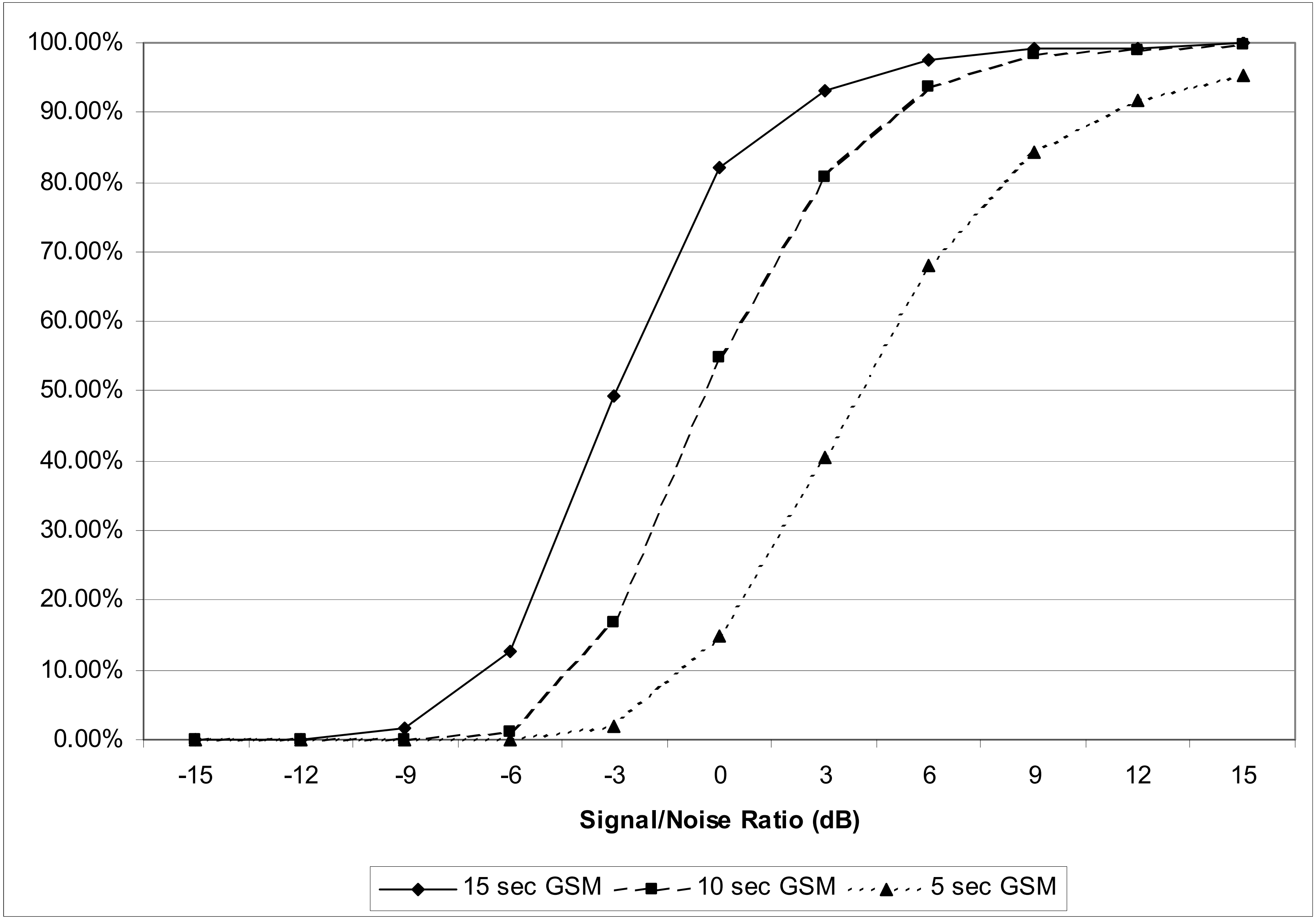
- Hash is formed between anchor point and each point in target zone, using frequency values and time delta
- Fan-out causes mini “combinatorial explosion” in number of tokens
- But compensated for by nearly 1e6 increase in speed and specificity.

What can it do?

Recognition rate – Linear PCM

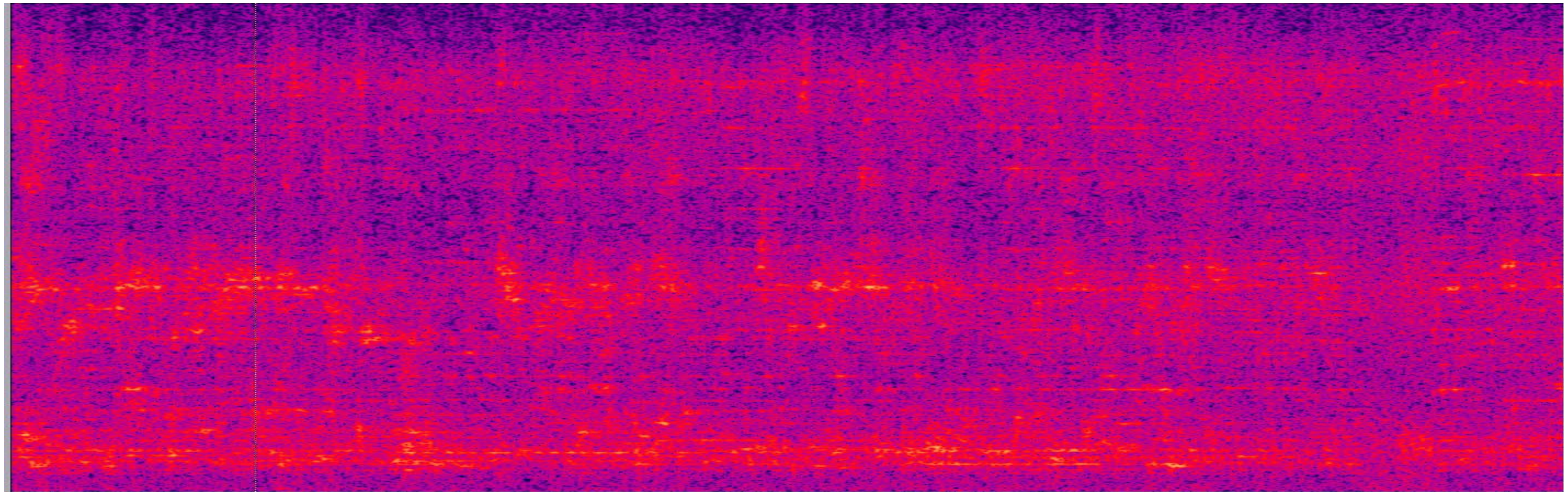


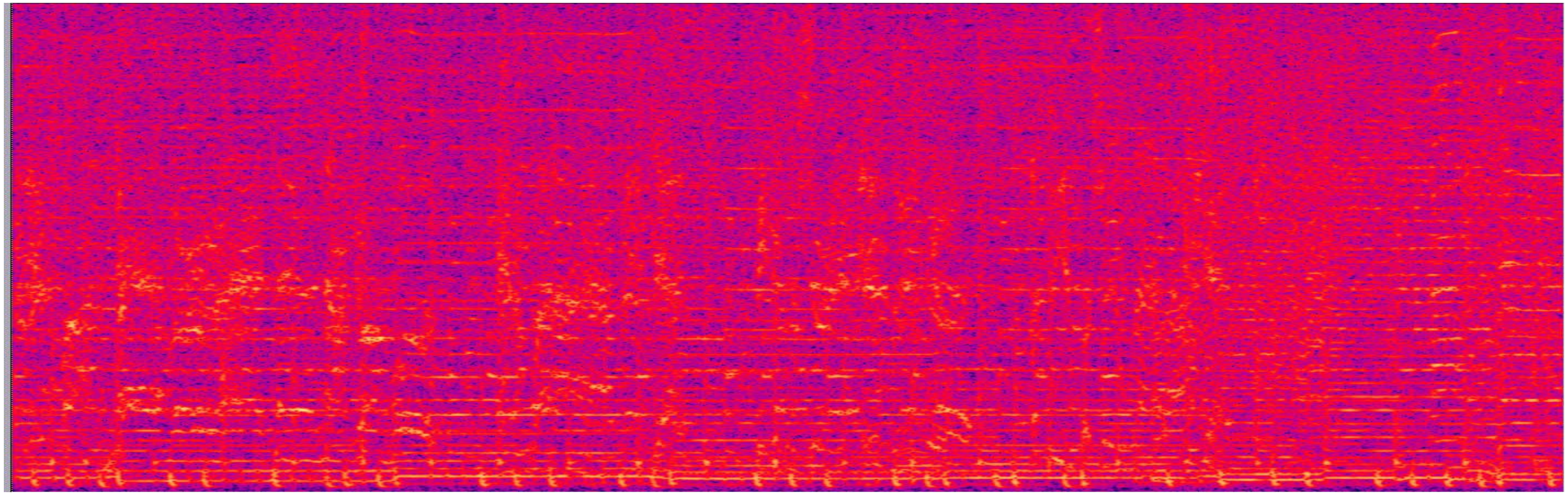
Recognition rate – GSM codec



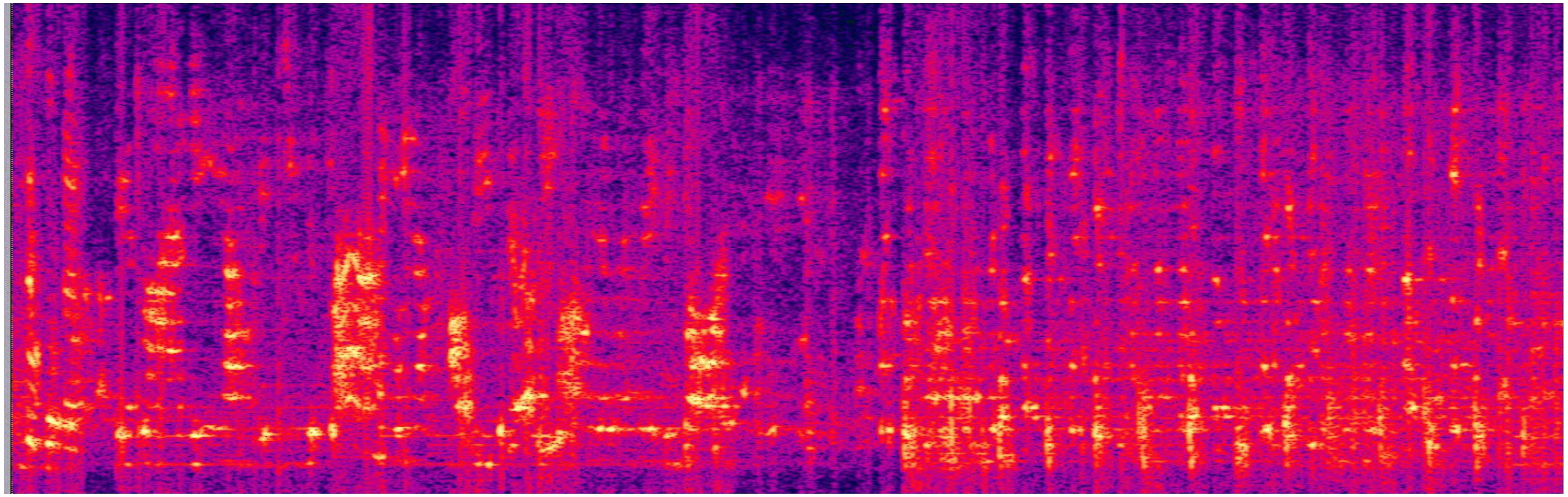
Sound Examples

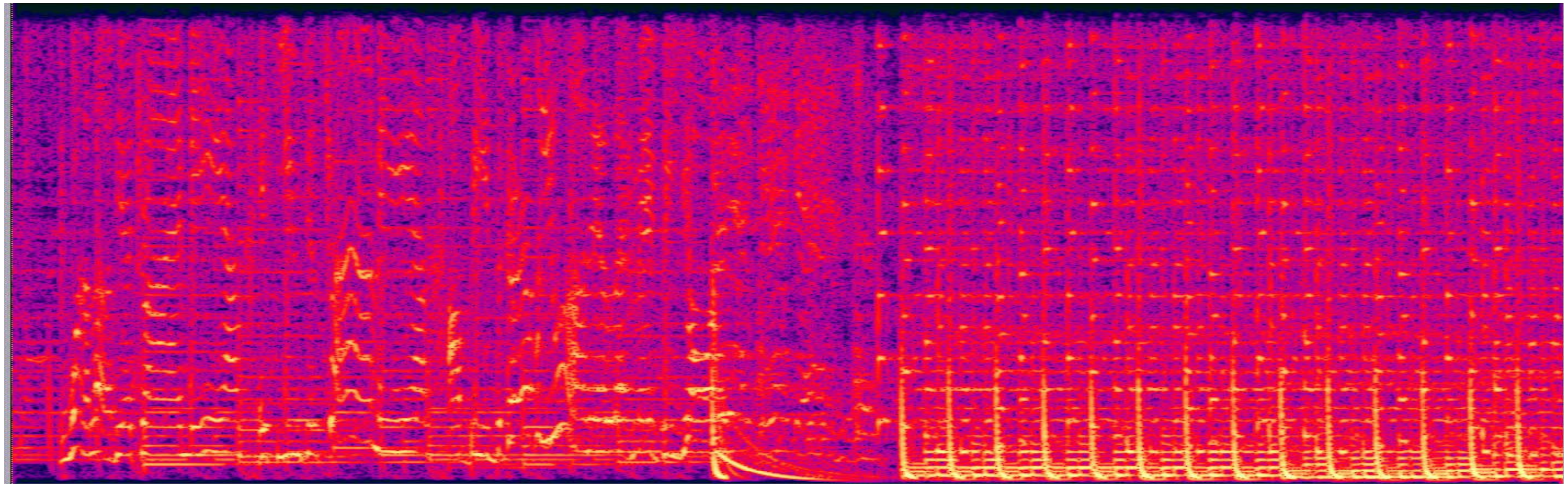
Example 1



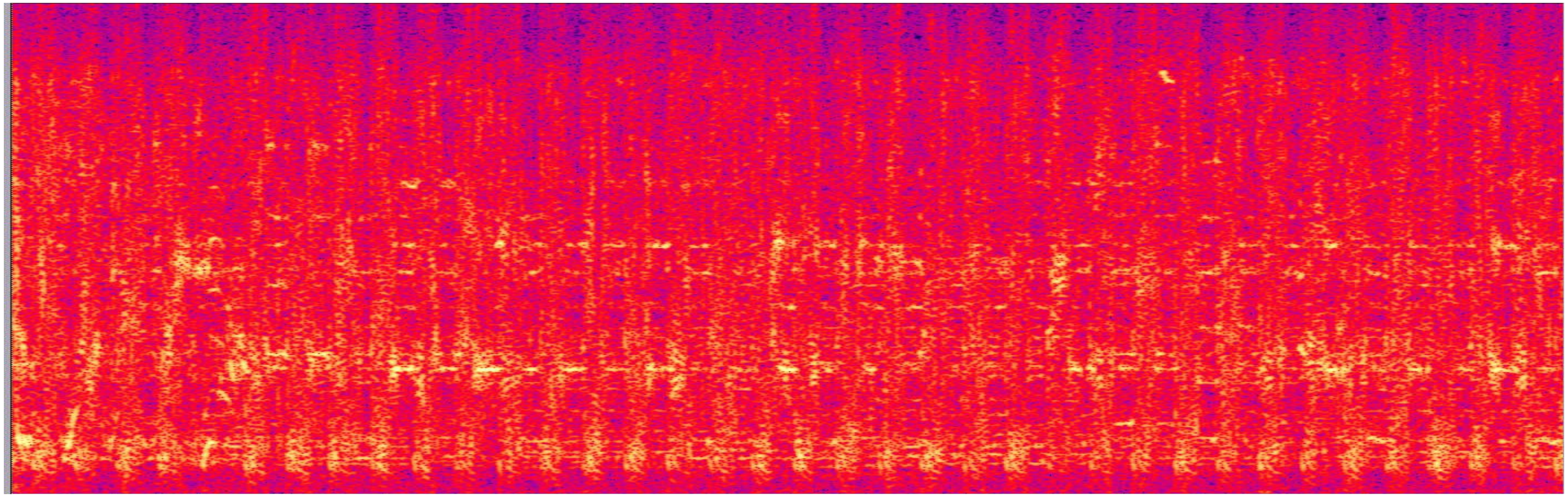


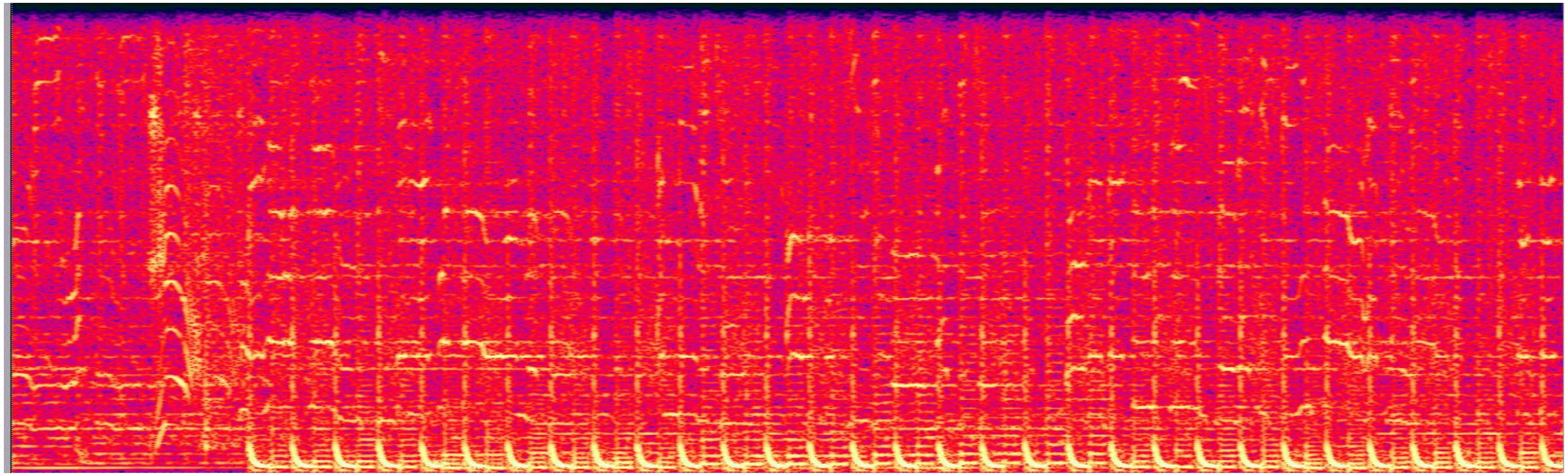
Example 2



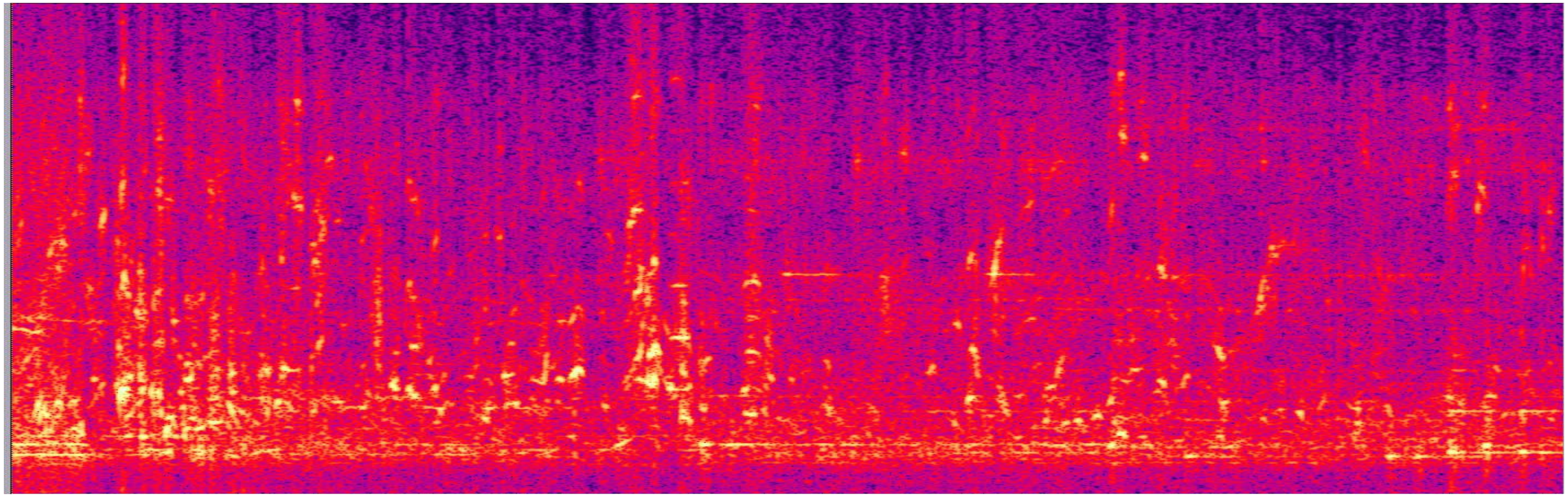


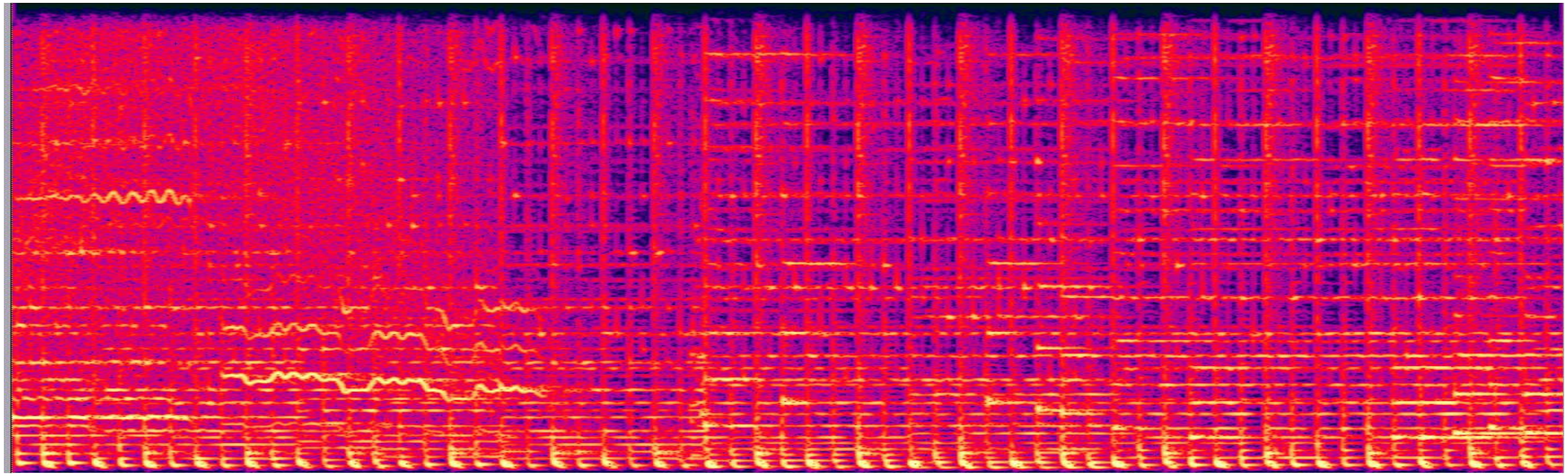
Example 3





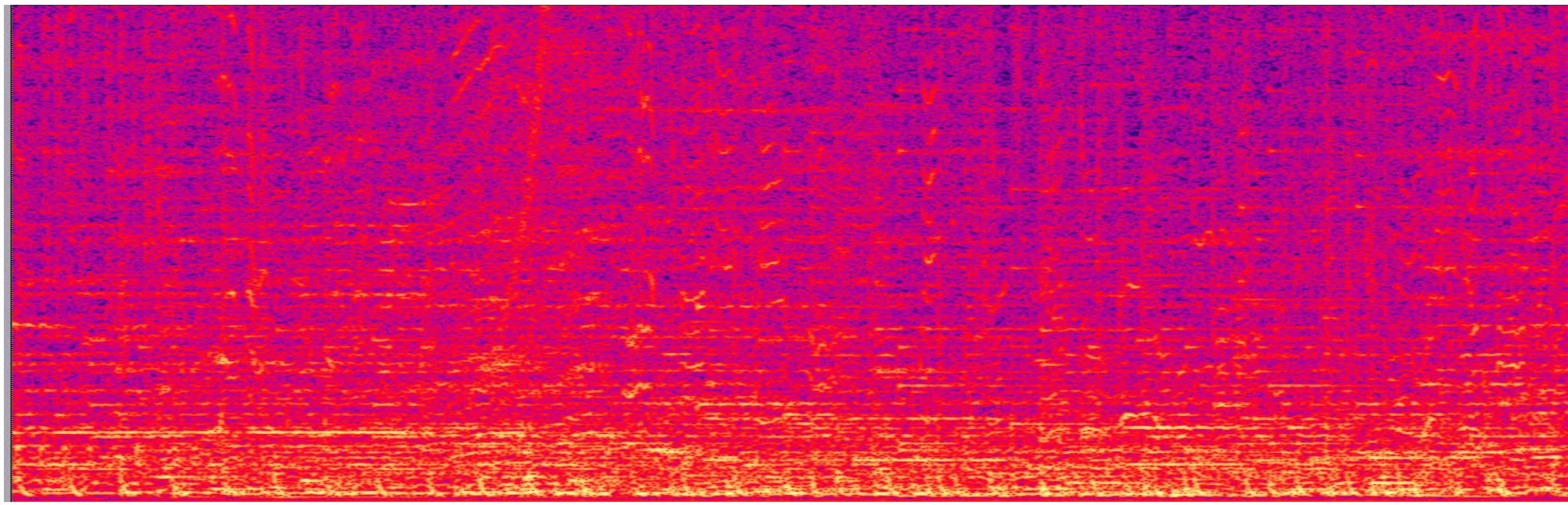
Example 4





Kajagoogoo and Limahl, *Never Ending Story*

Simultaneous Mix Example



1. Wim Mertens, ***Struggle for pleasure***
2. Brahms, ***Concerto for violin and Cello, A minor. Op. 102, allegro***
3. Ravel, ***Bolero*** (Dallas Symphony Orchestra)
4. Ravel, ***Bolero*** (London Symphony Orchestra)
5. Buena Vista Social Club, ***Chan Chan***
6. Robert Miles, ***Freedom***
7. M-People, ***One Night in Heaven***

Name that tune!

Live Example

Other Applications

- Radio monitoring
- Ad tracking
- P2P fileshare monitoring
- Library music identification
- Cueing and alignment
- Audio Google (query by example)
- Etc.

Conclusions

- Non symbolic
- Non-generalizing “exact matches”
- Highly noise resistant
- Highly scalable
- Very fast

Q&A