8 February 2019

Sound check

Analysis of Audio Signals

ELEC-E5620 - Audio Signal Processing, Lecture #4

Vesa Välimäki
Acoustics Lab, Dept. Signal Processing and Acoustics, Aalto University

Course Schedule in 2019 (Periods III-VI)

0. General issues (Vesa & Benoit)	11.1.2019
1. History and future of audio DSP (Vesa)	18.1.2019
2. Digital filters in audio (Vesa)	25.1.2019
3. Audio filter design (Vesa)	1.2.2019
4. Analysis of audio signals (Vesa)	8.2.2019
5. Audio effects processing (Benoit)	15.2.2019
* No lecture (Evaluation week for Period III)	22.2.2019
6. Synthesis of audio signals (Fabian)	1.3.2019
7. Reverberation and 3-D sound (Benoit)	8.3.2019
8. Physics-based sound synthesis (Vesa)	15.3.2019
9. Sampling rate conversion (Vesa)	22.3.2019
10. Audio coding (Vesa)	29.3.2019

Outline

- Spectral analysis using DFT & FFT
- Short-time Fourier transform and sinusoidal modeling
- Feature extraction
  - Envelope detection, spectral centroid, pitch detection, noise estimation, beat tracking

Some figures on these slides have been scanned from the textbook:
Ken Steiglitz, *A Digital Signal Processing Primer with Applications to Digital Audio and Computer Music*, Addison-Wesley, 1996.

Current and Future Applications

- Signal analysis is required in many audio processing tasks
  - Feature analysis of audio signals
  - Model parameter estimation for sound synthesis/coding
  - Signal/noise detection
  - Source separation
  - Automatic transcription

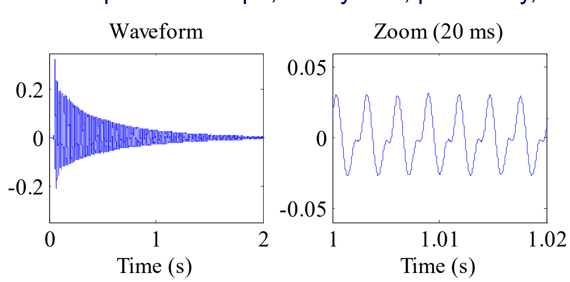
- Many current and future applications
  - Pitch correction
  - Audio restoration
  - Noise reduction
  - Automatic classification of audio (e.g. music/speech/commercial/silence)
  - Music understanding systems
    - Recognition of musical piece, style, composer, performer etc.

MPEG-7 Standard: Low-level Descriptors

- Standardized to enable the applications mentioned previously
- Basic
  – Instantaneous waveform, power, silence
- Basic Spectral
  – Power spectrum, spectral centroid, spectral spread, spectral flatness
- Signal Parameters
  – Fundamental frequency, harmonicity
- Timbral Temporal
  – Log attack time, temporal centroid of a monophonic sound
- Timbral Spectral
  – Features specific to the harmonic portions of signals (harmonic spectral centroid, spectral deviation, spectral spread, …)

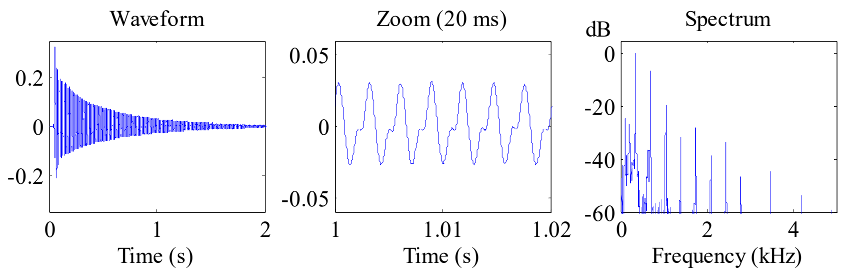
What Does the Waveform Tell?

- Signal waveform is the <u>lowest level signal representation</u>
  – Sampled pressure variations
- What can be seen from it?
  – Attack time and other temporal features of simple signals
  – Temporal envelope, decay rate, periodicity, smoothness?



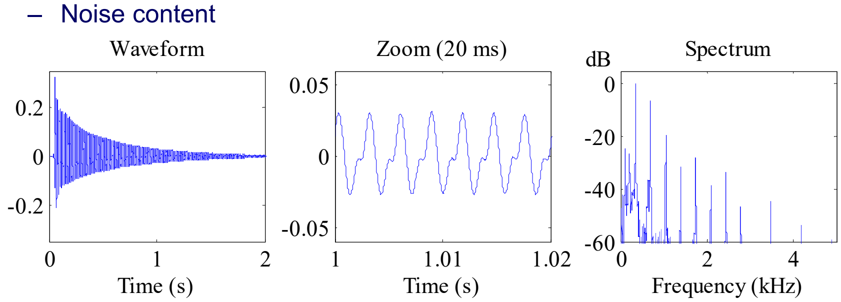
Spectral Analysis

- Spectrum is another useful representation
  - Human hearing works as a spectral analyzer
- For single tones, a one-shot spectrum is useful
  - Compute FFT of the whole signal



Spectral Analysis

- What can be seen in the spectrum?
  - Partials as peaks
  - Harmonicity / inharmonicity
  - Fundamental frequency?
  - Noise content



DFT

- The Discrete Fourier Transform
- A version of the Fourier transform for number sequences
  - Discrete in both time and frequency!
- Computes the frequency content at N discrete frequencies for an N-point sequence (sample index: n = 0, 1, 2, …, N – 1):

$$X_k = x_0 + x_1 e^{-\mathrm{j}k2\pi/N} + x_2 e^{-\mathrm{j}2k2\pi/N} + x_3 e^{-\mathrm{j}3k2\pi/N} + \ldots + x_{N-1} e^{-\mathrm{j}(N-1)k2\pi/N}$$

where k = 0, 1, 2, …, N – 1 is the frequency bin (discrete frequency index)

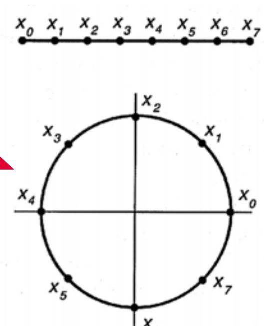
- DFT yields a complex-valued sequence
  - Absolute value $|X(k)|$ is the spectrum (magnitude spectrum)
  - Angle is the phase spectrum

DFT Example #1

- ***N* = 8**

Frequencies:

0, $f_s/8$, $2f_s/8$, $3f_s/8$, $4f_s/8$, $5f_s/8$, $6f_s/8$, $7f_s/8$



Nyquist limit

- When signal is real-valued…
  - ➢ bins 0, 1, 2, … $N/2$ contain unique info
  - 5 points out of 8 (X_0, …, X_4)
  - The rest are negative frequencies (X_5, X_6, X_7)
  - ➢ The spectrum is real-valued at 0 & Nyquist
  - No phase at 0 and at Nyquist! $e^{-\mathrm{j}0/N} = 1$
  - Symmetric and periodic weights $e^{-\mathrm{j}(N/2)n2\pi/N} = e^{-\mathrm{j}n\pi} = (-1)^n$

DFT Example #2

- ***N* = 1024, sampling rate 44100 Hz**
  - Spectrum computed at multiples of 44100/1024 Hz = 43.0664 Hz

	Bin	Frequency	
513 points	**0**	**0 Hz**	
	1	43.1 Hz	
	2	86.1 Hz	
	…	…	
	510	21964 Hz	
	511	22007 Hz	
	512	**22050 Hz**	**← Nyquist limit**
511 points	513	–22007 Hz	
	514	–21964 Hz	
	…	…	
	1022	– 86.1 Hz	
	1023	– 43.1 Hz	

FFT

- The Fast Fourier Transform algorithm was invented in 1960s (Cooley & Tukey, 1965)
  - And earlier by others, e.g. Carl Friedrich Gauss in 1800s
- Efficient computation of the discrete Fourier transform
  - Today it yields fast implementations for frequency-domain techniques
- Traditional Cooley-Tukey FFT is for lengths of power-of-2
  - E.g, 1024, 2048, 4096 etc.
  - Many other possibilities available (e.g., radix-3 FFT), but uncommon
- Number of multiplications is $O(N \log N)$ instead of $O(N^2)$
  - For a 1024-point DFT, the speedup factor is about 100
    (10,000 vs. 1,000,000)

Complexity of DFT

- DFT repeats the same operations and also multiplies by ±1

DFT: $e^{-j2\pi/N} = W_N \longrightarrow X_k = \sum_{n=0}^{N-1} x_n W_N^{nk}$

$$
O(N^2) \left\{ \begin{array}{l}
X_0 = x_0 W_N^0 + x_1 W_N^{0*1} + x_2 W_N^{0*2} + \ldots + x_{N-1} W_N^{0*(N-1)} \\
X_1 = x_0 W_N^0 + x_1 W_N^{1*1} + x_2 W_N^{1*2} + \ldots + x_{N-1} W_N^{1*(N-1)} \\
\ldots \\
X_{N-1} = x_0 W_N^0 + x_1 W_N^{(N-1)*1} + x_2 W_N^{(N-1)*2} + \ldots + x_{N-1} W_N^{(N-1)*(N-1)}
\end{array} \right.
$$

- For example, when N = 8, this leads to 8 x 8 = 64 complex multiplications

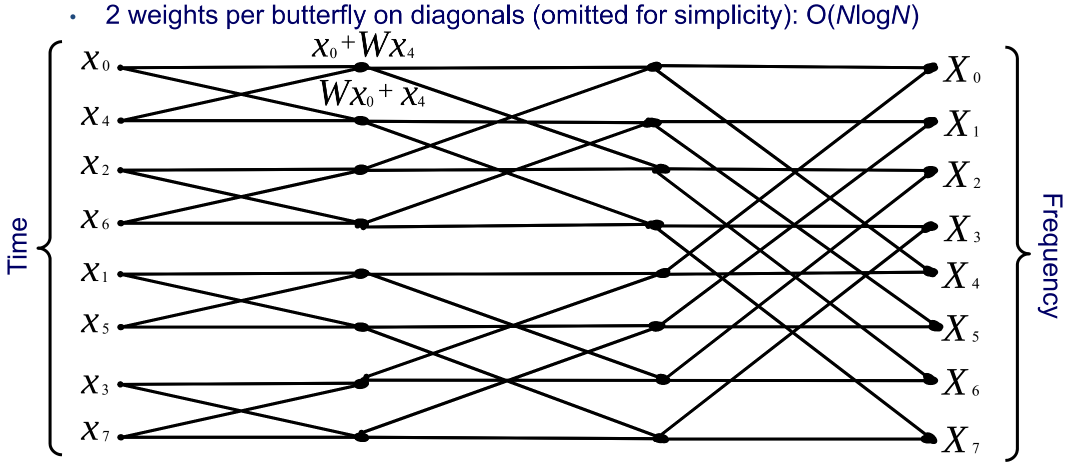
Trivial cases: $W_N^0 = 1$, $W_N^{N/2} = -1$

Symmetric: $W_N^{k+N/2} = -W_N^k$

Periodic: $W_{N/2}^{k+N/2} = W_{N/2}^k$

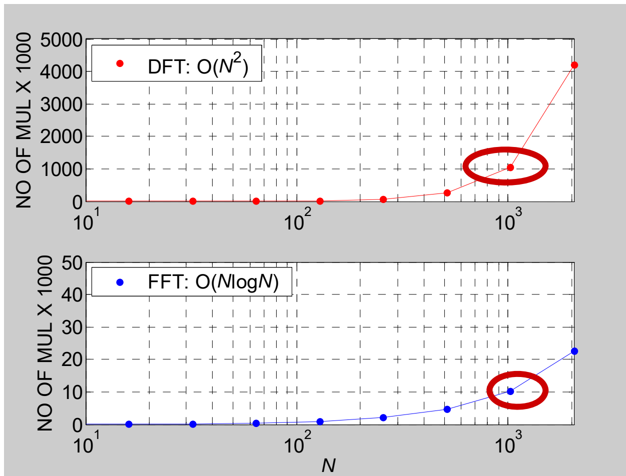
FFT: Divide and Conquer

- Butterfly diagram example with N = 8, leads to log(N) = 3 stages
- 2 weights per butterfly on diagonals (omitted for simplicity): O(Nlog$N$)



Example:
https://www.youtube.com/watch?v=EsJGuI7e_ZQ

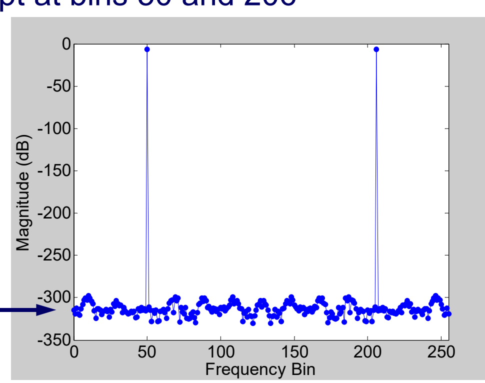
FFT: Huge Savings



Rounding Error in FFT

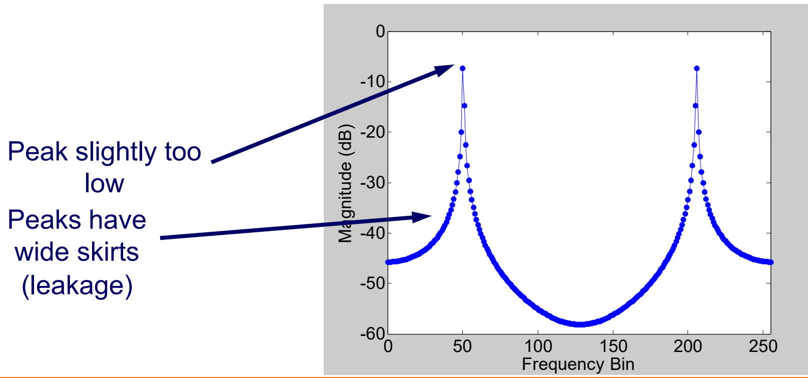
- N_{FFT} = 256
- Sinewave of freq. **50/256 X f_s**
- Spectrum should be 0 except at bins 50 and 206
  - Positive and negative freq.

Noise caused by rounding errors



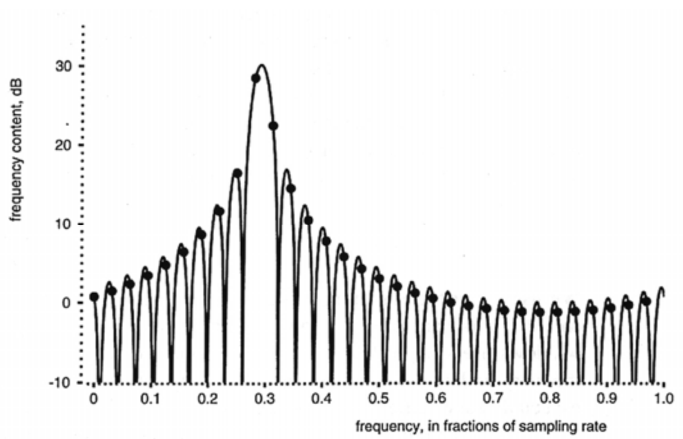
Frequency-Dependent Peak-Shape

- $N_{FFT} = 256$
- Sinewave of freq. **50.3/256 X f_s**



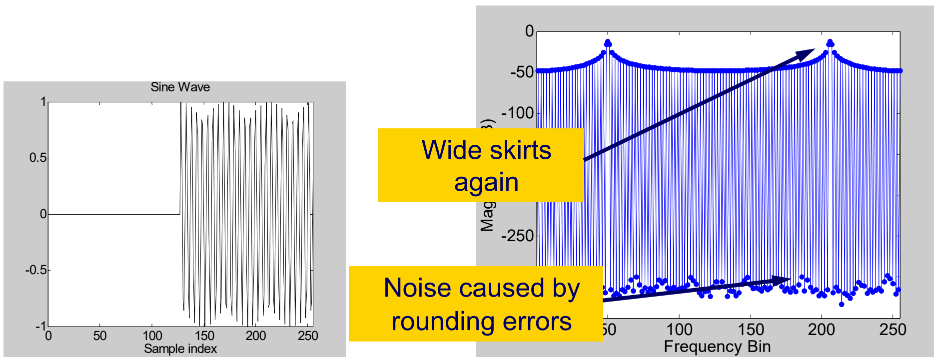
Spectral Leakage

- FFT samples the continuous spectrum at discrete points



Spectral Smearing

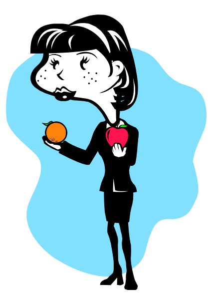
- N_{FFT} = 256
- Sinewave of freq. **50.3/256 X f_s**
- Sinewave starts at sample #128



FFT: Pros and Cons

☺ Much faster than DFT

☺ Accurate – no approximation



☹ No temporal information
- Signal onsets/offsets cause smearing

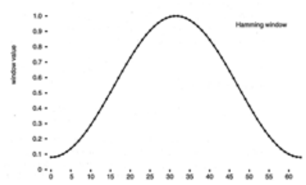
☹ Shape depends on frequency
- Wide main lobe
- Confusing side lobes
- Spectral leakage

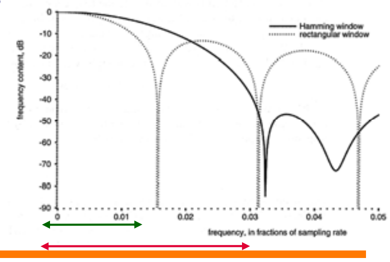
☹ Rounding errors look like additional noise

(Common to DFT and FFT)

Improvements for FFT

- Windowing
  - Smooth fade-in and fade-out of signal frames helps to suppress spectral leakage
  - A wide selection of window functions
    - Tradeoff between width of central lobe and suppression of side lobes
- Zero-padding
  - Extend signal frame with a sequence of zeros
  - Interpolates the discrete spectrum





Parabolic Interpolation

- Fit a parabola to a local maximum and its 2 neighbors
  - Obtain peak value at the peak of the parabola
  - Accuracy depends on true shape of the lobe (e.g. window function)

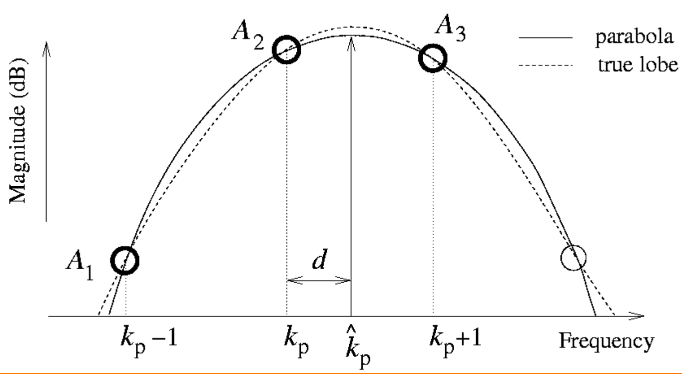
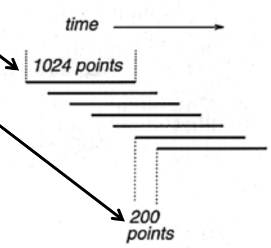


Figure by Dr. Paulo A. A. Esquef

Short-Time Fourier Transform (STFT)

- It makes sense to analyze sounds with a running spectrum
  - Human hearing analyzes sound in both time and frequency
- Sounds are stationary over time intervals of about 10 ms
  - 441 samples @ 44.1 kHz
- Short-time Fourier transform (STFT) is a sequence of FFTs
  - FFT length can be 128…1024 samples
  - Hop size is usually 10…512 samples
  - Overlap = (FFT length) – (hop size)
  - Use windows to crossfade frames



Computing the STFT

- Divide signal into frames
  - Decide frame length and hop size
- Window each signal frame
  - e.g., with a Hamming window
- Then compute the FFT

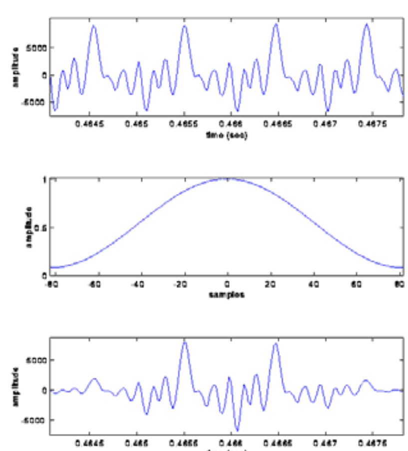
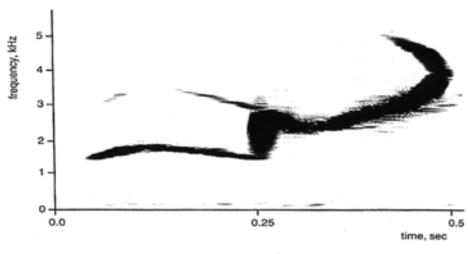


Figure taken from (Serra, 1997)

Spectrogram and Waterfall

- Two alternatives to visualize the STFT



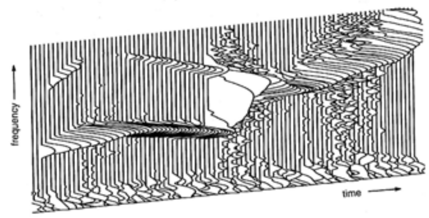
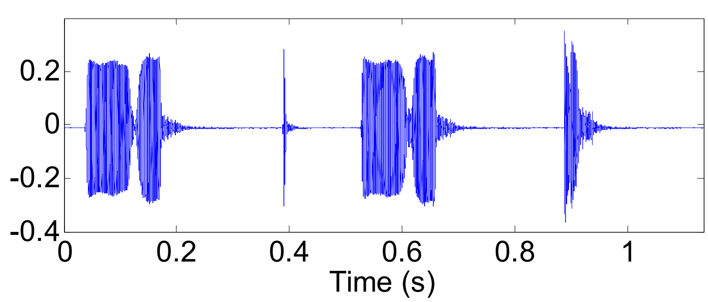


Fig. 7.3 Waterfall version of the spectrogram in the preceding figure, also using 1024-point Hamming windows.

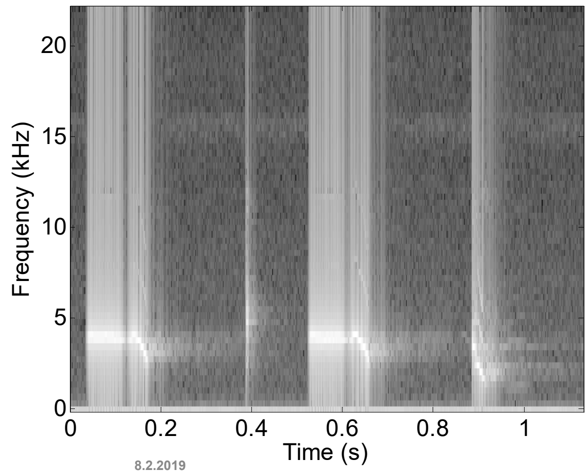
STFT Examples

- Example signal: bird singing (*nightingale*, 'satakieli' in Finnish)
  - Tonal components, fast variations in time and frequency, transients



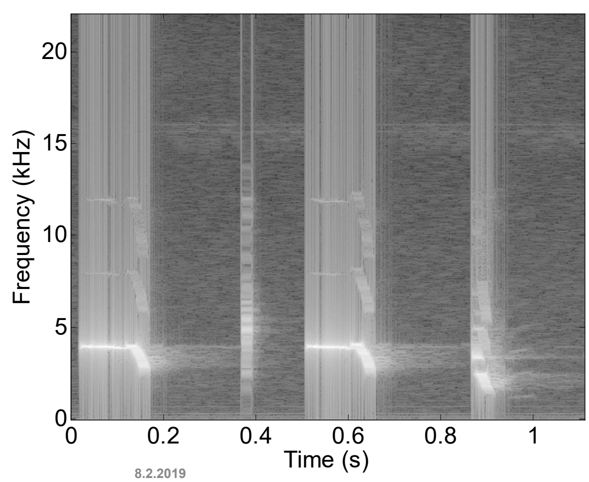
STFT Example #1

- N_{FFT} = **128,** Hop size = **64,** Window: **Rectangular**
  - **Fuzzy!**



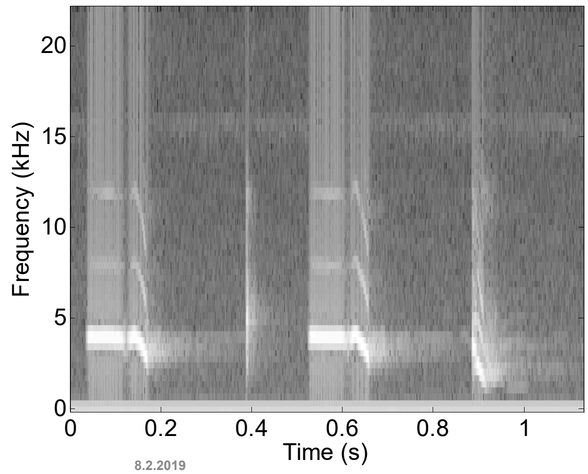
STFT Example #2

- N_{FFT} = **1024,** Hop size = **64,** Window: **Rectangular**
  - **Better but still fuzzy**



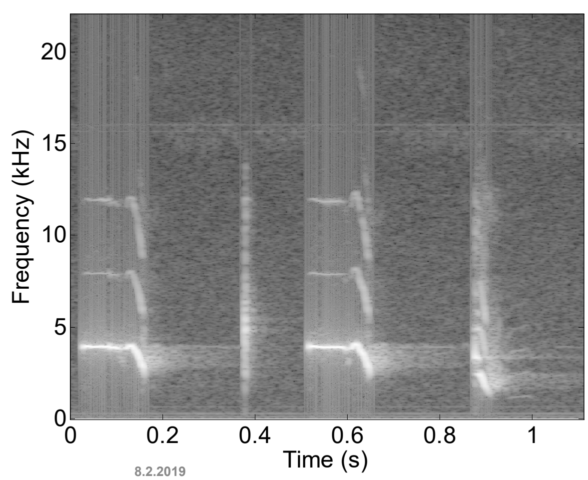
STFT Example #3

- N_{FFT} = **128,** Hop size = **64,** Window: **Hamming**
  - **Better but bad**



STFT Example #4

- N_{FFT} = **1024,** Hop size = **64,** Window: **Hamming**
  - **Pretty good**



Sinusoidal Modeling

- As an extension of STFT,
- sinusoidal components can
- be extracted
  - McAulay-Quatieri algorithm
  - Tracking phase vocoder
- Find peaks of $|X(k)|$
  - These correspond to sinusoidal components
- Phase can be looked up from the phase spectrum at peak frequencies
  - Requires zero-phase windowing

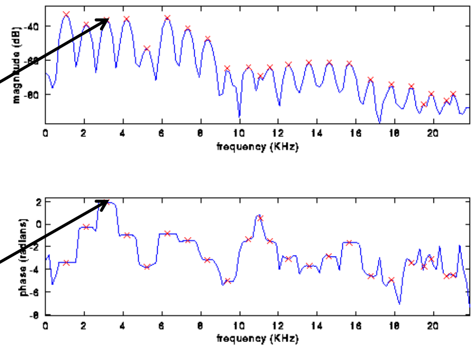


Figure taken from (Serra, 1997)

Zero-Phase Windowing of Frames

- FFT assumes circular time
- Split the windowed frame from the
- middle and switching the 2 parts
- (Smith and Serra, 1987)
  - If zero-padding is used, zeros go in the middle
- Magnitude spectrum is unchanged
- Phase spectrum becomes clean

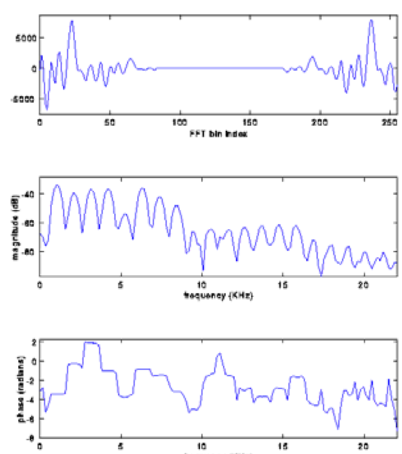


Figure taken from (Serra, 1997)

Peak Picking

- Usually it does not make sense to pick all peaks
  - Take *N* tallest peaks (e.g., 100) or all peaks above a threshold

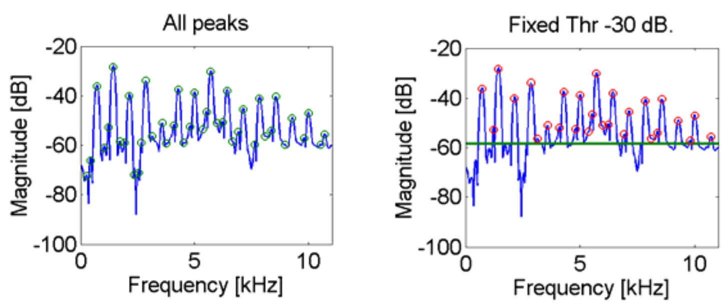


Figure by Paulo A. A. Esquef

Continuous Trajectories

- Search location of each spectral peak in the following frame
  - Yields the time-dependence of frequency and amplitude

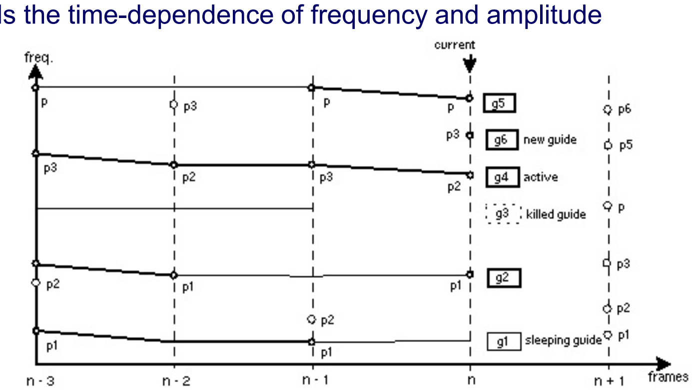
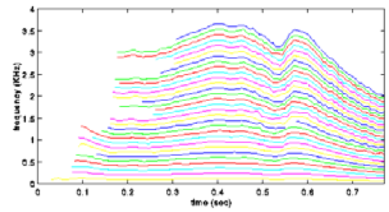


Figure taken from (Serra, 1997)

Frequency and Amplitude Tracks

- Frequency and amplitude of each sinusoidal component as a function of time
- This is called the deterministic part of the signal (Serra, 1997)
  - The remaining 'noise' is called 'stochastic part'



Original Deterministic Stochastic

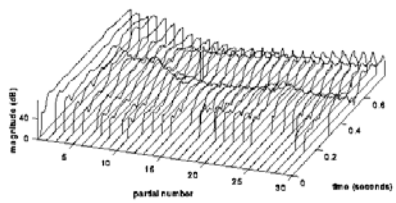
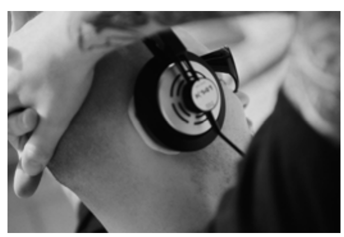


Figure taken from (Serra, 1997)

Aalto University
School of Electrical Engineering

35 8.2.2019

Features of Audio Signals



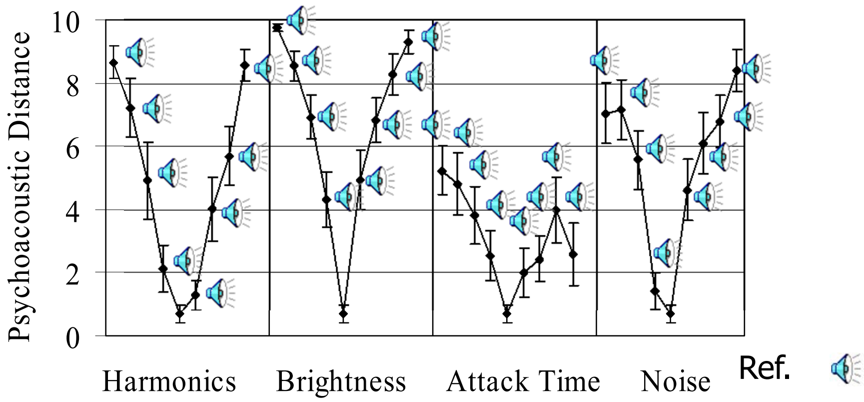
- Features relevant to humans
  - Duration
  - Loudness
  - Pitch
  - Timbre
    - A multi-dimensional feature
    - "The feature that enables people to separate two tones that have the same loudness, pitch, and duration, but that are still different"
    - Several factors affect "timbre", such as brightness, balance between even and odd harmonics, noise content ("noisiness"), temporal envelope (e.g., attack sharpness), and inharmonicity

Aalto University
School of Electrical Engineering

36 8.2.2019

Perception of Timbre-Related Features

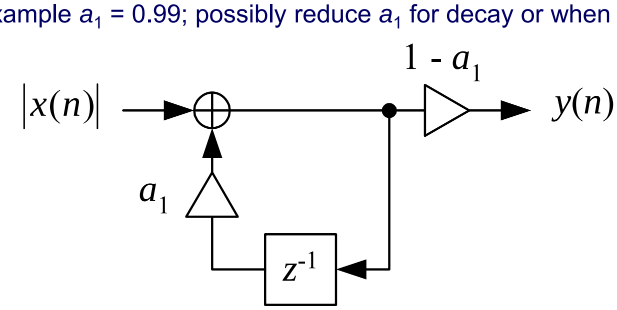
- Modification of 4 features of a cello tone for brain research



Ref: M. Ilmoniemi et al. 2004

Envelope Detection

- Full-wave rectification (abs) and temporal averaging
- Leaky integrator:
- $y(n) = (1 - a_1)\,|x(n)| + a_1\, y(n-1)$, where $a_1 = 1 - \varepsilon$
  - For example $a_1 = 0.99$; possibly reduce a_1 for decay or when $|x(n)| < y(n)$



Demo: Loudness Normalization

Rytis Stasiunas & Máté Szokolai

Aalto University
School of Electrical
Engineering

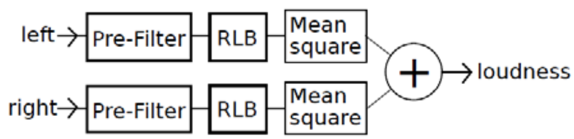
Loudness Estimation

- Running RMS value
  - Time-varying estimate proportional to instantaneous signal power
- Convert to decibels
  - 20 log[y(n)]
  - Human sensitivity to loudness follows approximately logarithmic relation
- An auditory model of loudness perception is needed in principle
  - Should account for frequency-dependent sensitivity of human hearing
  - For example brightness affects loudness perception

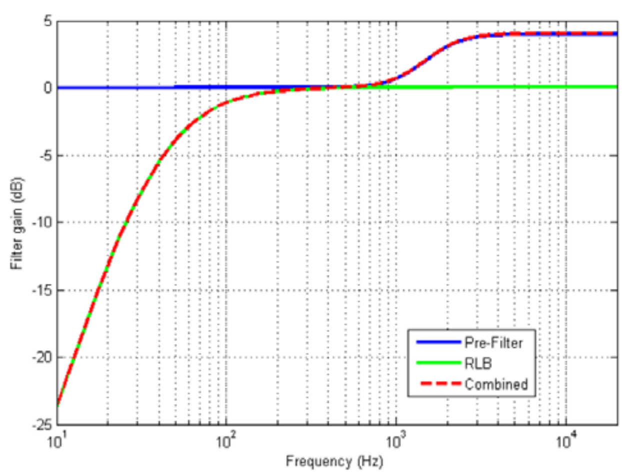
Aalto University
School of Electrical
Engineering
40
8.2.2019

Loudness Estimation using RLB

- A recent recommendation ITU-R BS.1770 uses a simple approximation of frequency-dependent sensitivity
  - A head-related shelving filtering (pre-filter)
  - Revised Low-Frequency B weighting (RLB)
- Can be used for adjusting the levels of music files in a playlist



Loudness Estimation using RLB



Demo: Pitch Detection

Seyoung Park

Aalto University
School of Electrical
Engineering

Pitch

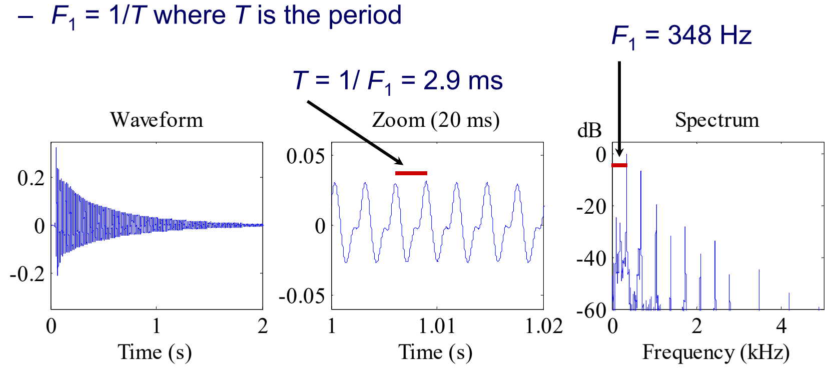
- Pitch is the perceived fundamental frequency
  - F0 is a physical quantity – pitch is a subjective attribute
  - Pitch is the frequency that (musical) humans would sing, whistle when asked about the height of a musical tone
  - Alternatively, test subject can adjust the frequency of a sine wave to match a test tone
- For sine waves: pitch = F0
- Humans perceive pitch clearly for very complex tones
  - Pitch of complex harmonic and even inharmonic tones (e.g., bells)
  - Also ”missing fundamental” is strongly perceived (e.g., on the phone)
  - The auditory system tries to assign a pitch to all sounds

Aalto University
School of Electrical
Engineering

44 8.2.2019

Pitch of Harmonic Tones

- For harmonic complex tones: pitch = F_1
  - F_1 is frequency of the lowest common factor of harmonic frequencies
  - $F_1 = 1/T$ where T is the period



Pitch Extraction

- Pitch estimation methods were first developed for speech
  - Today hundreds of estimation methods available
- Methods can be classified into two classes
  - 1) Time-domain methods: periodicity, T
  - 2) Frequency-domain methods: fundamental frequency, F_1
- Problematic algorithms
  - Large errors are usually octave errors (one octave up or down)
  - Pre- or post-processing may reduce errors
    - For example, compression or spectral whitening of input signal, median filtering of a sequence of F0 estimates
- The newest ’good’ method is YIN (de Cheveigné and Kawahara, 2002)

Autocorrelation Method

- A classical method used in speech processing for decades
- Compute correlation of the signal *x*(*n*) with itself in short frames (Rabiner, 1977)

$$r_l(m) = \frac{1}{N} \sum_{k=0}^{N-1-m} x'(l+k)x'(l+k+m), \quad 0 \le m \le M-1$$

where $x'(n)$ is the windowed signal of length N

(but $N - m$ samples used), m is the lag,

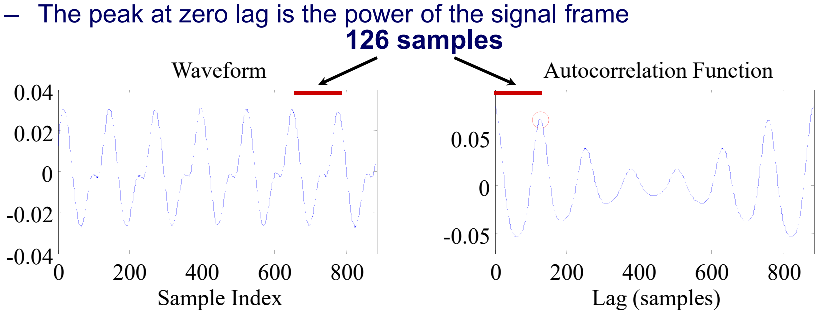
M is the number of autocorrelation points computed, and

l is the starting sample of the frame

- Select the second maximum as the estimate for period

Autocorrelation Example

- Compute as IFFT{|$X(k)$|^2}
  - Autocorrelation function is inverse Fourier transform of power spectrum
- Autocorrelation peaks at the fundamental period
  - The peak at zero lag is the power of the signal frame



  - Frame length = 884 samples; Blackman window

Running Autocorrelation

- Just as STFT or centroid, the autocorrelation function can be executed for short frame with hops
  - Even every sample
  - Usually frames overlap considerably
  - Can perform well on musical sounds with clear harmonic nature
  - Gives easily errors with rapidly changing signals
  - It helps if the range of F0 can be restricted

Speech (male)

Pop singing (female)

Demo by Hannu Pulakka and Kari Valde, TKK, 2003

Aalto University
School of Electrical Engineering

49

8.2.2019

Pitch Detection Applications

- Auto-Tune by Antares Audio Technologies (2007-) (Hildebrand, US Patent, 1999)
  - ➢ Detect pitch and modify (autocorr, interpolate, resynth)
- Singing computer games are based on pitch detection
  - ➢ Hedgehog game by Elmorex, Finland, 2000 (Hämäläinen *et al.*, 2004)
  - ➢ Staraoke (MTV3 Junior 2003-2009)
  - ➢ Sony SingStar (2004-)

Aalto University
School of Electrical Engineering

50

8.2.2019

Multi-Pitch Estimation

- Practical applications require estimation of multiple F0s simultaneously
  - Music usually consist of several instruments playing together and chords
  - The F0 of all or at least some of the most prominent tones is needed
- Iterative F0 estimation and cancellation
  - First find the most prominent tone and its F0
  - Cancel it from the mixture (e.g. inverse comb filtering or subtraction of harmonics)
  - Take the next tone and so on
  - E.g. A. Klapuri
    (IEEE Trans. SAP, Nov. 2003 and Proc. ISMIR'2006)

Spectrum of Several Tones Sounding Together

- Interpretation of the spectrum gets difficult with more than 1 tone

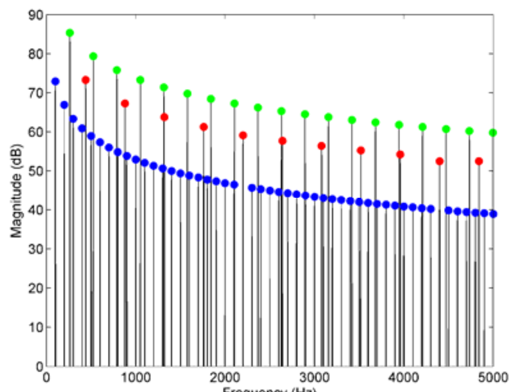


Figure by Jukka Rauhala

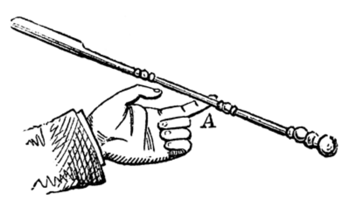
Multi-Pitch Estimation

- A very difficult challenge – research continues
- However, in products multi-pitch estimation actually works!
  - Melodyne uses DNA (Direct Note Access) technology for manipulation of notes in chords (2009-) (Neubäcker, US patent 2011)
  - In March 2013, Ableton Live announced the use of Audio2Note algorithm developed at IRCAM: polyphonic pitch processing!

Spectral Centroid

- Brightness of an audio signal can be described by the center of gravity of its magnitude spectrum

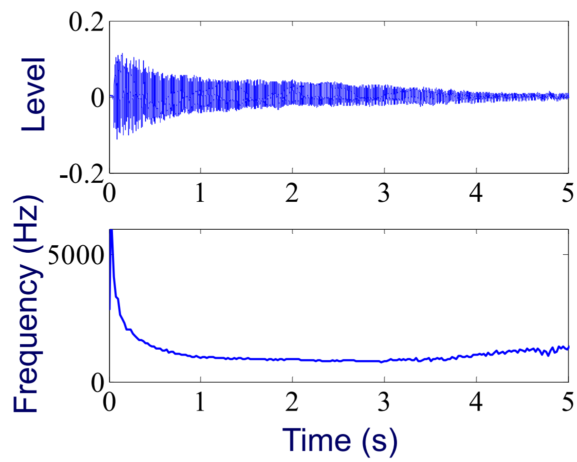
$$c = \frac{f_s}{N} \frac{\sum_{k=0}^{N/2} k\left|X(k)\right|}{\sum_{k=0}^{N/2} \left|X(k)\right|}$$



  - Magnitude spectrum of signal's derivative divided by magnitude spectrum
  - Note normalization by sampling rate f_s and FFT length N
- Alternatively, squared magnitude spectra $|X(k)|^2$ can be used or magnitude of harmonics only

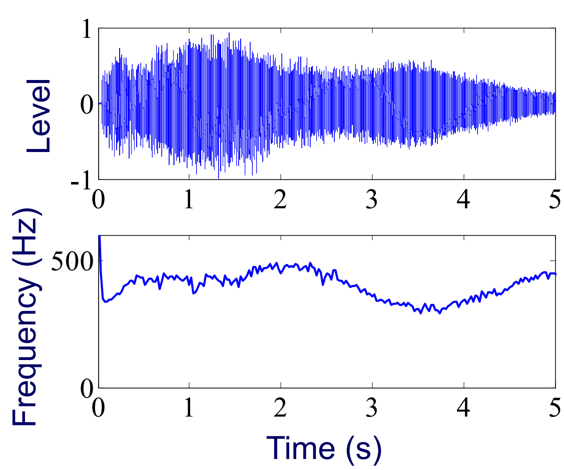
Centroid Example #1

- Harpsichord tone
  - B3, F0 = 58 Hz
- 2048-point FFT
  - 46 ms at 44.1 kHz
- Hop size: 1024 samples
  - 23 ms at 44.1 kHz
  - Average: 2029 Hz
  - Standard deviation: 999 Hz



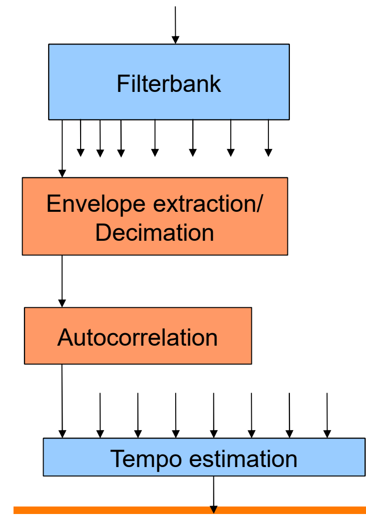
Centroid Example #2

- A ’tam-tam’ signal
- 2048-point FFT
  - 46 ms at 44.1 kHz
- Hop size: 1024 samples
  - 23 ms at 44.1 kHz
  - Average: 362 Hz
  - Standard deviation: 91.1 Hz



http://www.sfu.ca/sonic-studio/handbook/Sound/Inharmonic_Tamtam.aiff

Example 1: Beat detection

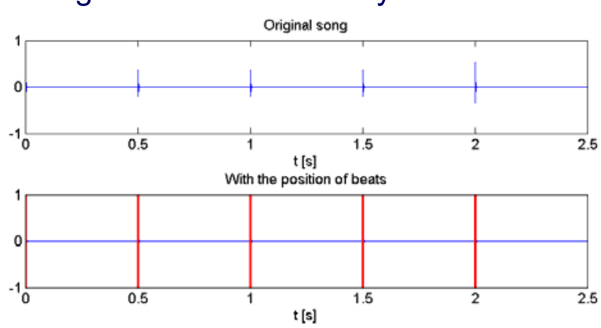


- Frequency division into 8 log-spaced frequency bands
- Extraction of temporal envelope and decimation to a lower sample rate
- Periodicity detection using autocorrelation
- Comparison between periodicity results across frequency bands and collection of the most prominent peaks

Demo by Politis and Brabec, 2010

Example 1: Beat detection

- Rhythmic structure detection
- Periodicity detection
- Higher level music analysis

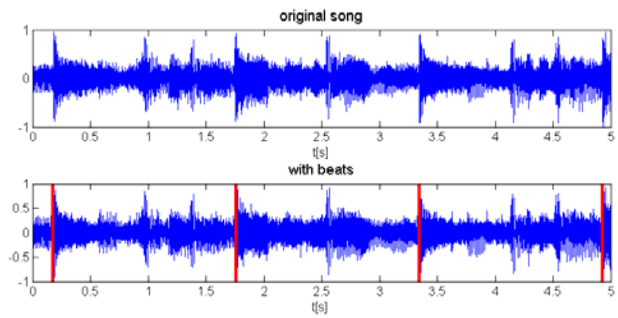


original

mixed

Demo by Politis and Brabec, 2010

Example 1: Beat detection

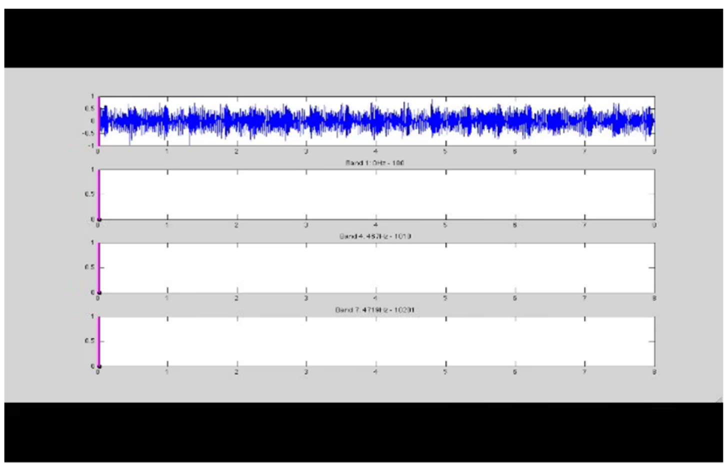



original


mixed

Demo by Politis and Brabec, 2010

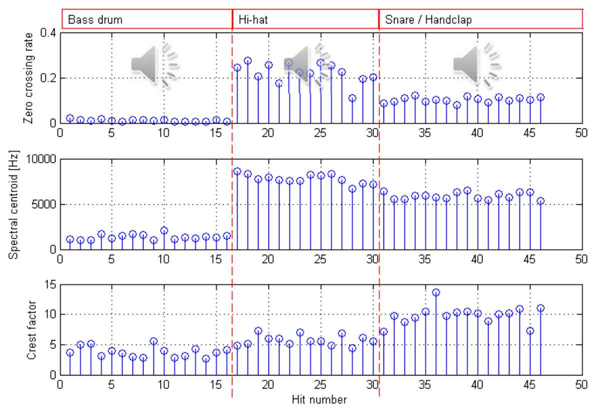
Example 1: Beat detection



Demo by Politis and Brabec, 2010

Example 2: Beatbox replacement

- Beatboxing sounds classified based on 3 audio features



$$Z_{cr} = \frac{1}{2N}\sum_{n=1}^{N-1}\left|\mathrm{sgn}(x(n+1)) - \mathrm{sgn}(x(n))\right|$$

$$Centroid = \frac{F_s}{N}\frac{\sum_{n=0}^{N/2}|X(n)|\cdot n}{\sum_{n=0}^{N/2}|X(n)|}$$

$$Crest = \frac{|x|_{peak}}{x_{rms}}$$

Demo by Shaohong Li and Antti Pakarinen, 2012

Example 2: Beatbox replacement

- Antti's beatboxing replaced with Roland TR808 samples

Photo from http://en.wikipedia.org/wiki/Roland_TR-808

Demo by Shaohong Li and Antti Pakarinen, 2012

Conclusions

- Time-varying spectral features are useful
  - Short-time Fourier spectrum (spectrogram)
  - Use an appropriate window to avoid leakage
  - Features as function of time (pitch, centroid…)
- Audio content analysis has great applications
  - Music recognition, smart effects, remixing, karaoke, beat tracking, music transcription…
- Ultimate goal: human-like understanding of musical sounds by computer
  - Capabilities of musically educated people in pitch and tempo tracking and content recognition

References

- J. B. Allen, "Short term spectral analysis, synthesis, and modification by discrete Fourier transform," *IEEE Trans. Acoustics, Speech, and Signal Processing*, vol. 25, no. 3, pp. 235–238. June 1977.
- D. Arfib, F. Keiler, and U. Zölzer, "Source-filter processing," in U. Zölzer (ed.), *DAFX – Digital Audio Effects*, Wiley, 2002, pp. 299-372.
- A. de Cheveigne and H. Kawahara, "YIN, a fundamental frequency estimator for speech and music," *Journal of the Acoustical Society of America*, vol. 111, no. 4, pp. 1917–1930, April 2002.
- H. A. Hildebrand, "Pitch detection and intonation correction apparatus and method," *US Patent 5,973,252*, Issue date: Oct. 26, 1999.
- P. Hämäläinen, T. Mäki-Patola, V. Pulkki, and M. Airas, "Musical computer games played by singing." In *Proc. 7th Int. Conf. Digital Audio Effects*, pages 367–371. Naples, Italy, Oct. 2004.

References (page 2)

- M. Ilmoniemi, V. Välimäki, and M. Huotilainen, "Subjective evaluation of musical instrument timbre modifications," in *Proceedings of Baltic-Nordic Acoustics Meeting* (BNAM2004), Mariehamn, Aland, June 8–10, 2004. http://www.acoustics.hut.fi/asf/bnam04/
- A. P. Klapuri, "Multiple fundamental frequency estimation based on harmonicity and spectral smoothness," *IEEE Transactions on Speech and Audio Processing*, vol. 11, no. 6, pp. 804–816, Nov. 2003.
- A.P. Klapuri, A.J. Eronen, and J.T. Astola, "Analysis of the meter of acoustic musical signals," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 14, no. 1, pp. 342–355, Jan. 2006.
- A.P. Klapuri, "Multiple fundamental frequency estimation by summing harmonic amplitudes," in *Proc. 7th International Conference on Music Information Retrieval*, Victoria, Canada, 2006.
- P. Neubäcker, "Sound-object oriented analysis and note-object oriented processing of polyphonic sound recordings," *U.S. Patent 8,022,286*, Issue date: Sep 20, 2011.

References (page 3)

- L. R. Rabiner, "On the use of autocorrelation analysis for pitch detection," *IEEE Trans. Acoustics, Speech, and Signal Processing*, vol. 25, no. 1, pp. 24–33. Feb. 1977.
- E. Scheirer, "Tempo and beat analysis of acoustic musical signals," *Journal of the Acoustical Society of America*, vol. 103, no. 1, pp. 588–601, 1998.
- X. Serra, "Musical sound modeling with sinusoids plus noise," in C. Roads *et al.* (eds.), *Musical Signal Processing.* Swets & Zeitlinger, 1997. Available on-line: http://www.iua.upf.es/~xserra/articles/msm/
- T. Tolonen and M. Karjalainen, "A computationally efficient multi-pitch analysis model," *IEEE Trans. Speech and Audio Processing*, vol. 8, no. 6, pp. 708–716, Nov. 2000.
- S. Boll, "Suppression of acoustic noise in speech using spectral subtraction," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol.27, no.2, pp.113–120, Apr 1979.
- R. Martin, "Noise power spectral density estimation based on optimal smoothing and minimum statistics," *IEEE Transactions on Speech and Audio Processing*, vol. 9, no. 5, pp. 504–512, July 2001.