

ASSP Library/Tools

Spektralanalyse

Florian Schiel/Lasse Bombien

Seminar Werkzeuge der Sprachverarbeitung
(Sommersemester 2012)

Vortrag

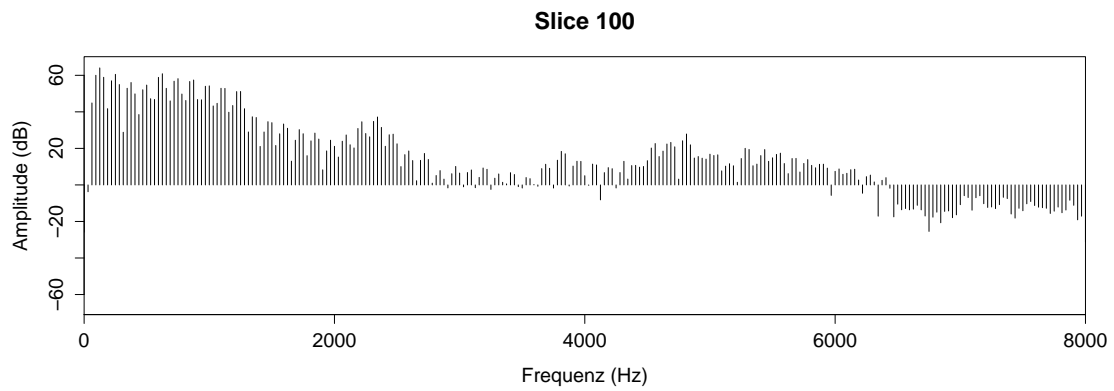
Spektrale Daten DFT

- Mit der Fourier Analyse kann man ein Signal in die einzelnen Sinuskomponenten zerlegen. Dabei ergibt die Summe der einzelnen Komponenten wiederum das ursprüngliche Signal.
- Im Digitalen Bereich verwendet man die Diskrete Fourier Transformation (DFT; oder die Fast Fourier Transformation, FFT) um die Energie/Amplitude in aufeinanderfolgenden Abschnitten des Frequenzspektrums zu ermitteln.
- Die ASSP Tools bieten unter anderen Spektralen Analyseverfahren auch die DFT an.

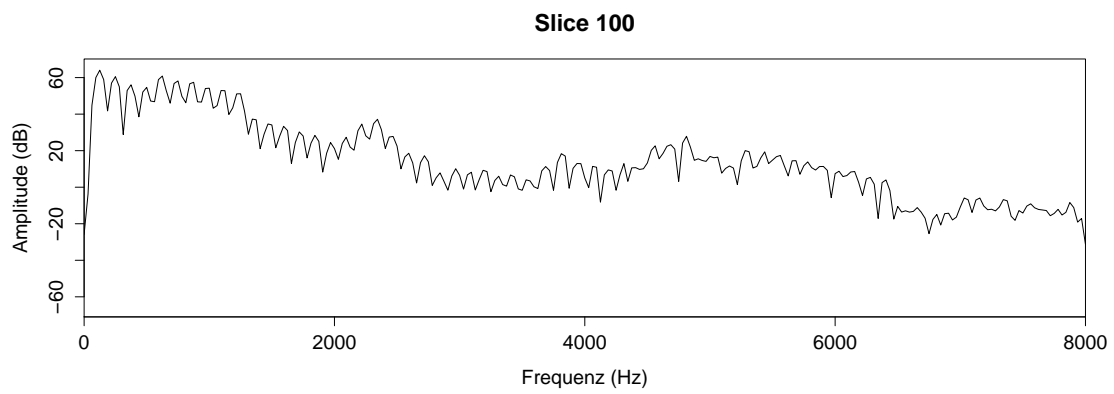
Wichtige Analyseparameter

- Abtastrate des Signals (f_s).
- FFT-Länge: Anzahl der Abtastpunkte im Analysefenster (N), 2er Potenz
- In ms entspricht die FFT-Länge: $1000 \frac{N}{f_s}$
 - $f_s = 16000 \text{ Hz}, N = 64: d = 1000 \frac{64}{16000 \text{ Hz}} = 4 \text{ ms}$
 - $f_s = 44100 \text{ Hz}, N = 64: d = 1000 \frac{64}{44100 \text{ Hz}} = 1.45 \text{ ms}$
- Frequenzauflösung (frequency resolution, f_{res}): $\frac{f_s}{N}$
 - $f_s = 16000, N = 256: f_{res} = \frac{16000 \text{ Hz}}{256} = 62.5 \text{ Hz: Schmalband}$
 - $f_s = 16000, N = 64: f_{res} = \frac{16000 \text{ Hz}}{64} = 250 \text{ Hz: Breitband}$
- Anzahl der Spektralwerte: $\frac{N}{2} + 1$

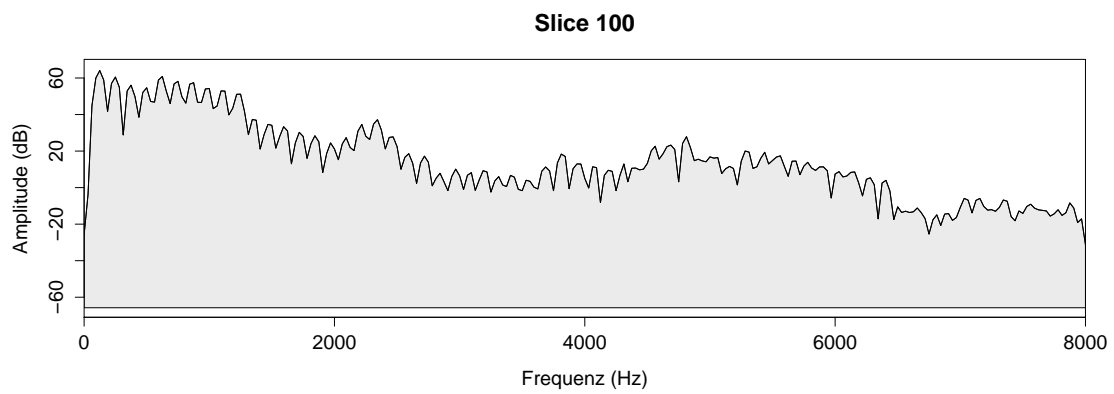
Linienspektrum, Schmalband



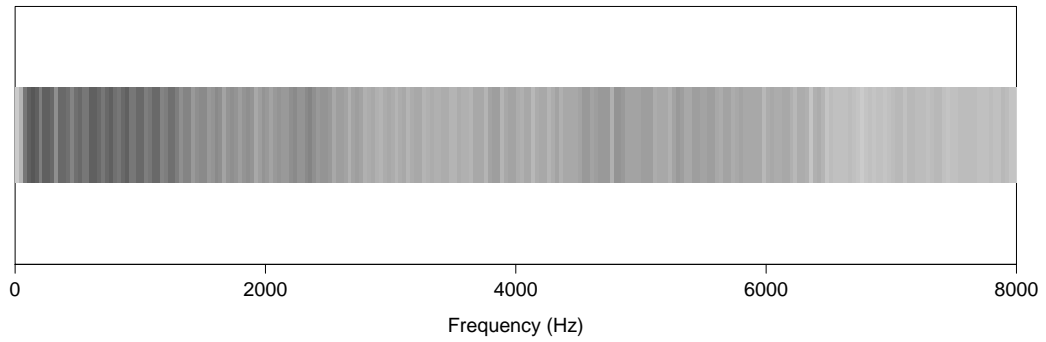
Spektrum, Schmalband



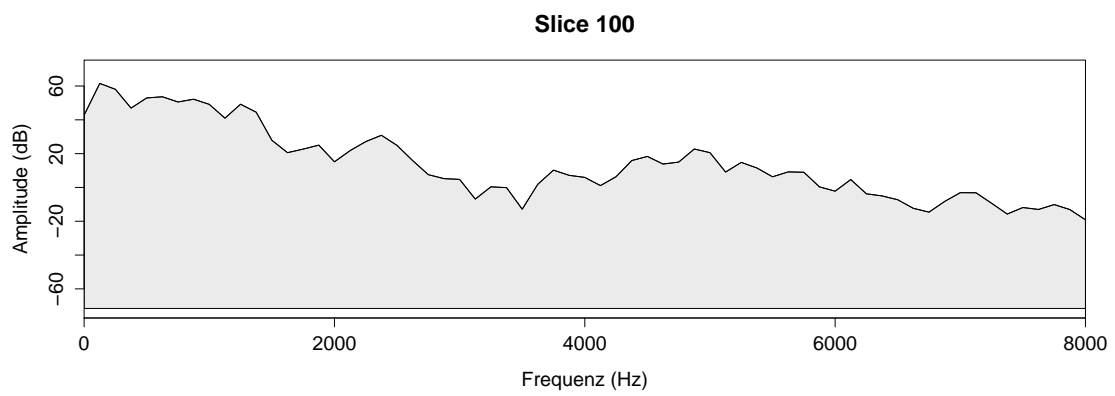
Spektrum, Schmalband, gefüllt



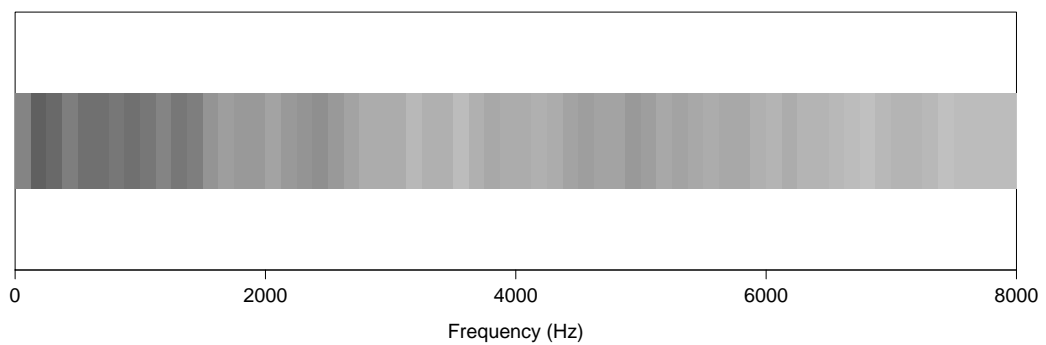
Stellen wir uns vor, es handelt sich hierbei um eine Scheibe aus grau getöntem Plexiglas, mehrere cm dick (nach "hinten" raus). Wie sieht sie von Oben betrachtet aus?



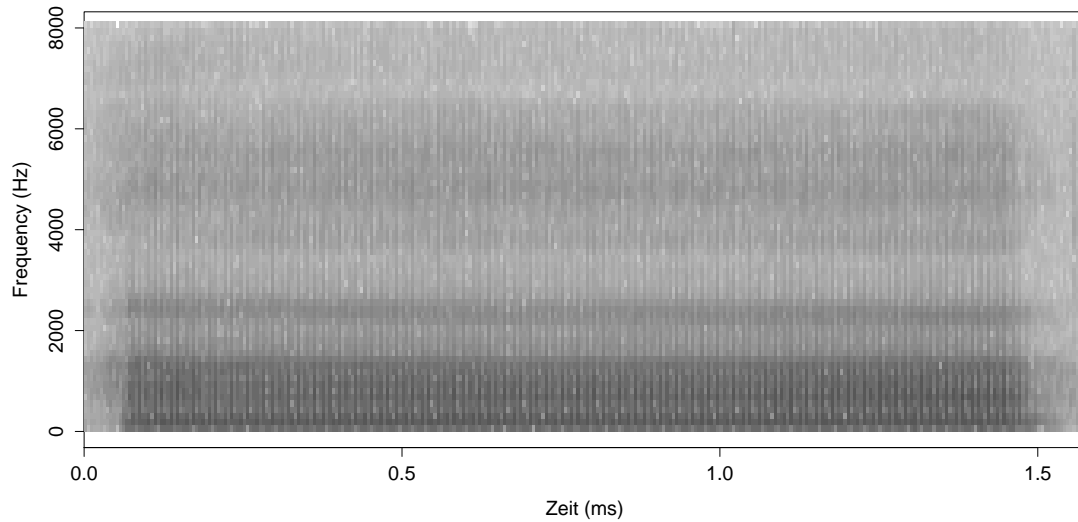
Spektrum, Breitband, gefüllt



Breitband



Spektrogramm



Übung

Führen Sie die folgenden Befehle zur Berechnung von spektralen Daten aus:

- `spectrum -t=DFT -oA -of=affaere.dft affaere.wav`
- `spectrum -t=CSS -oA -of=affaere.cep affaere.wav`

Starten Sie R (z.B. `rstudio &`) und führen Sie folgende Befehle aus:

```
dft <- read.table("affaere.dft")
DFT <- t(dft[,2:514])
plot(DFT[,500], type="l")
cep <- read.table("affaere.cep")
CEP <- t(cep[,2:514])
plot(CEP[,500], type="l")
q()
```