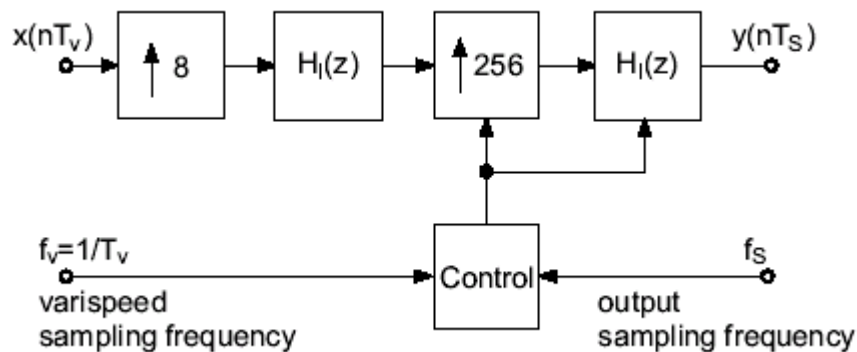
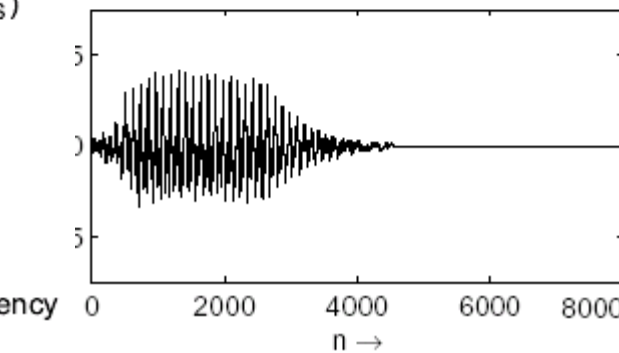
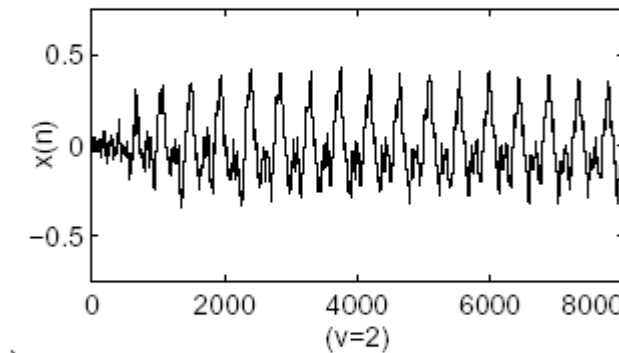
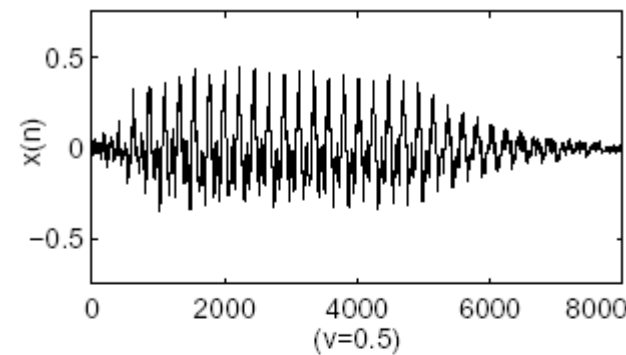
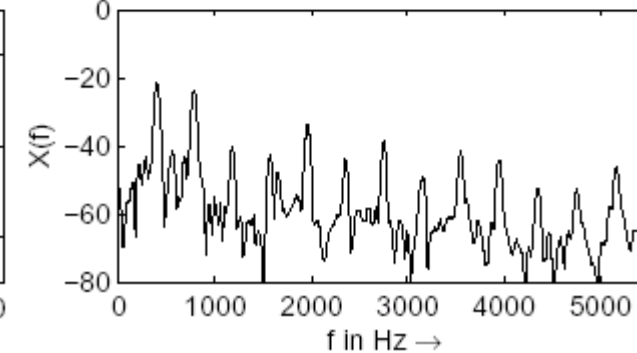
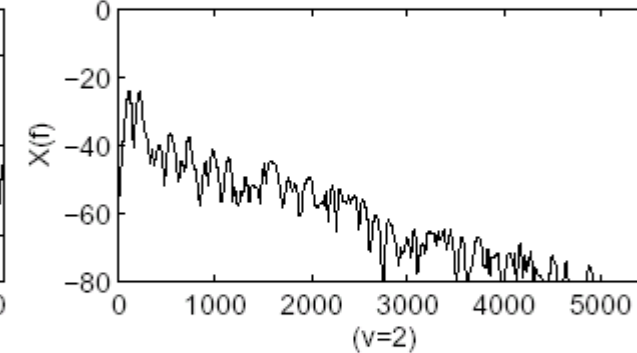
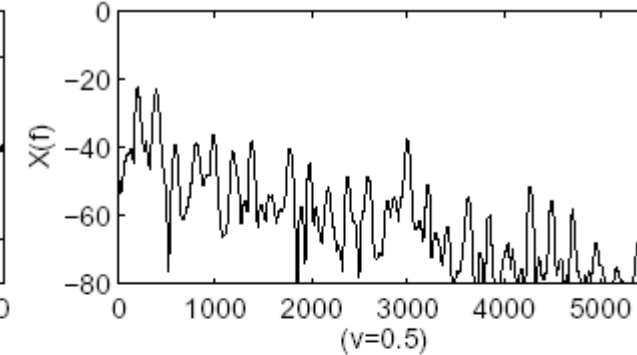


Allgemein

- Zeitdehnung
- Tonhöhenänderung
- Kurzzeit-Fourier-Transformation
 - Phasenvocoder
 - Filterbanksummation
 - Blockfiltertechnik
 - Zeit-Frequenz-Skalierung
 - Effekte

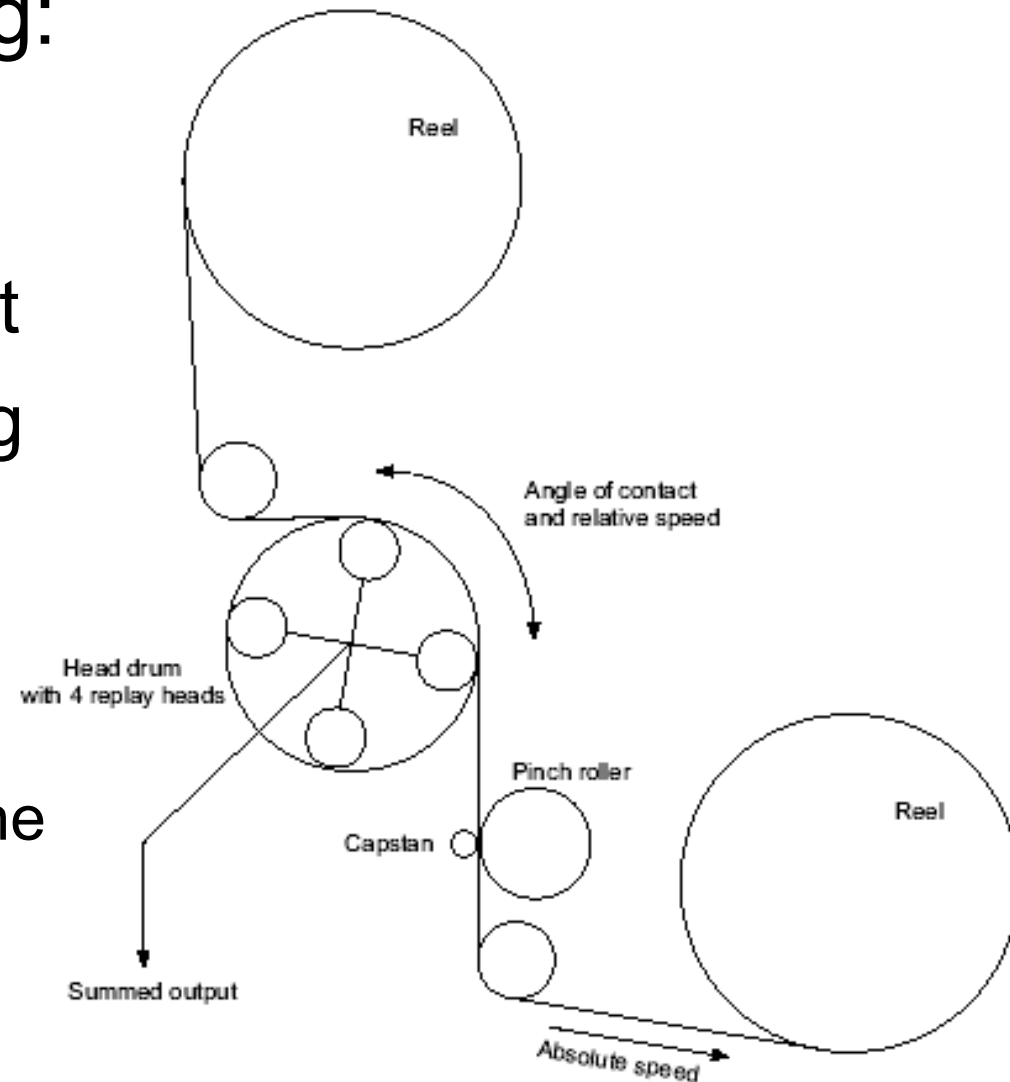
Zeitdehnung

- Veränderung der Abspielgeschwindigkeit
 - Zeitdehnung (time stretching)
 - Nachteil: Klangveränderung

Variable speed replay ($v=1$), time domain signalsVariable speed replay ($v=1$), spectra

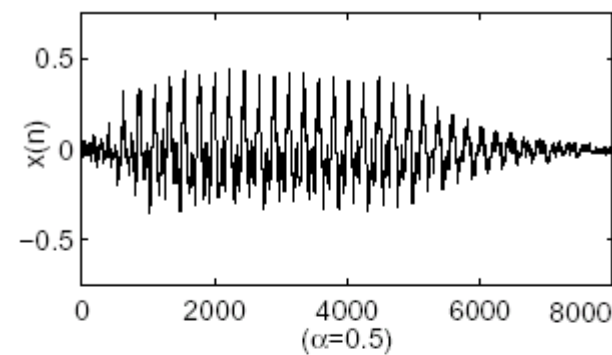
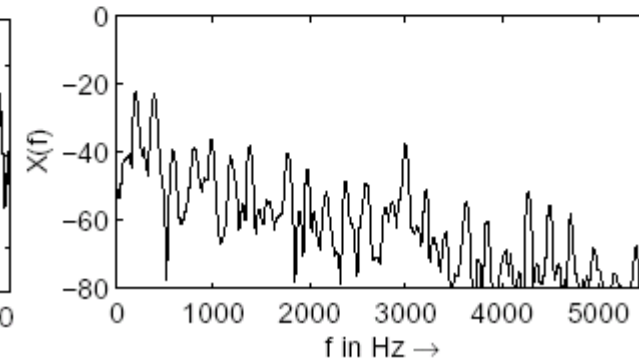
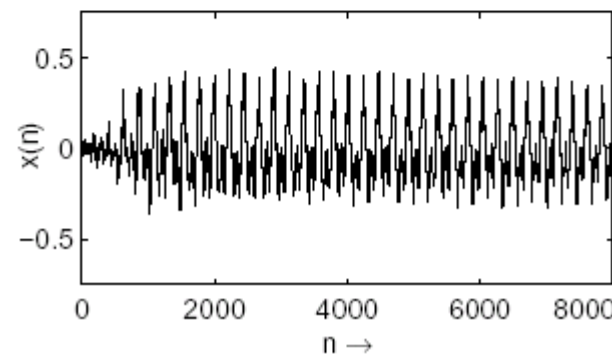
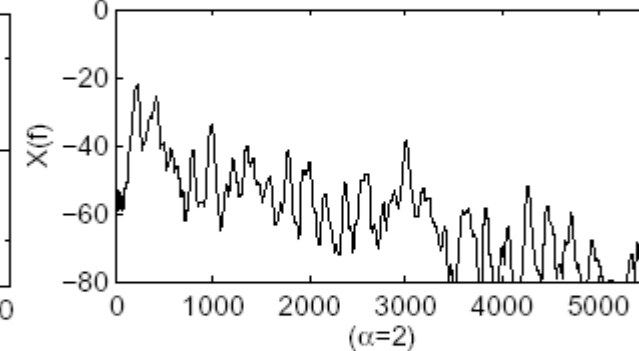
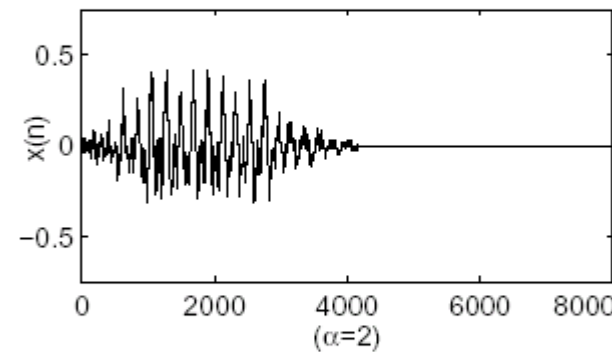
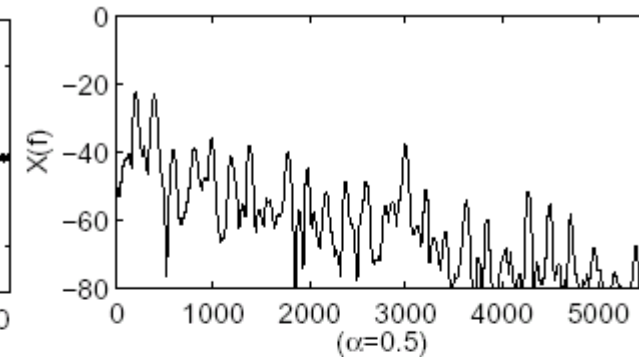
Zeitdehnung

- Elektroakustische Lösung:
 - Abspielgeschwindigkeit wurde durch die relative Geschwindigkeit bestimmt
 - Kontaktfläche: Fensterung eines Blocks
 - Mehrere Abtastköpfe:
 - Overlap-And-Add
 - Reduktion der Kontaktfläche



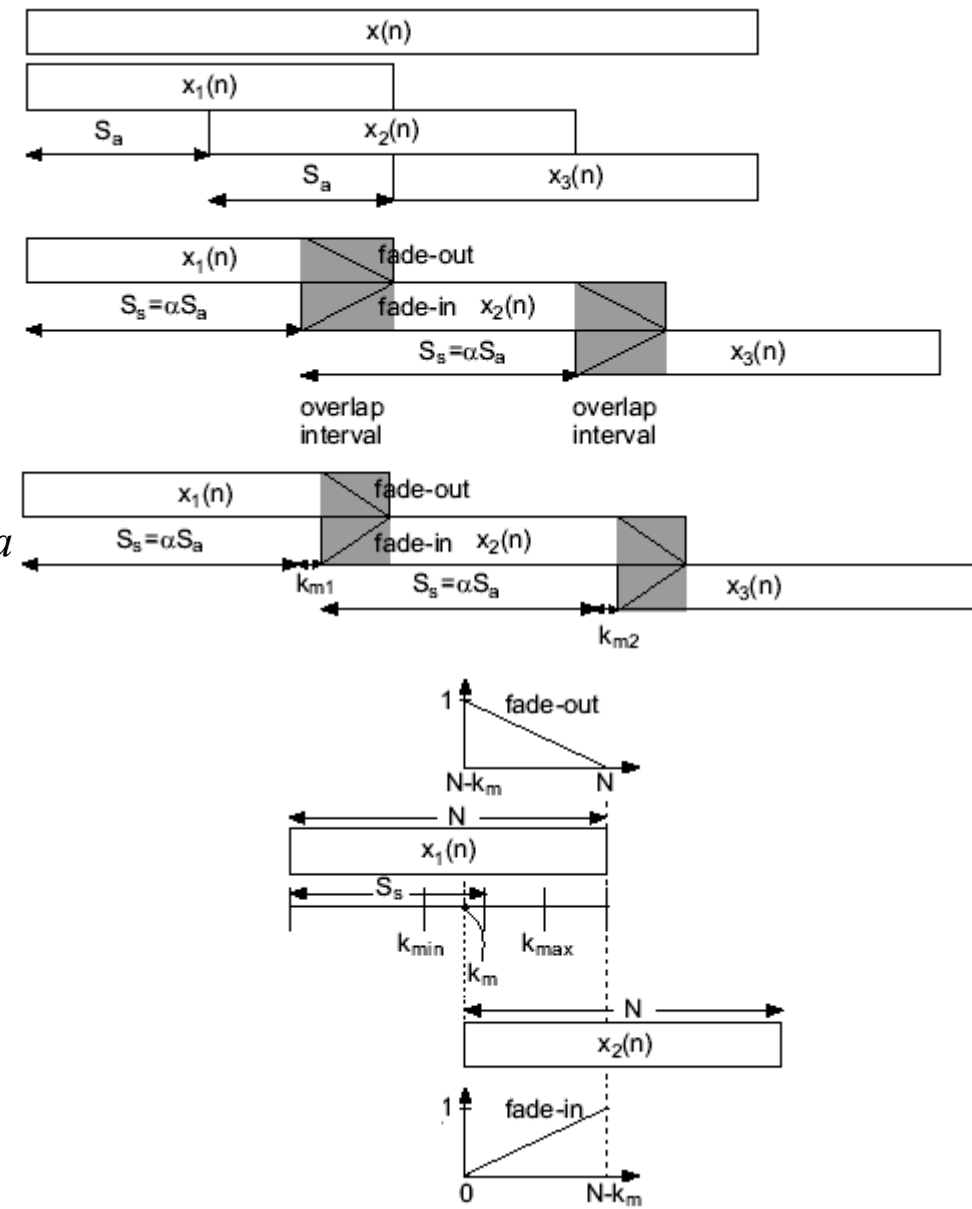
Zeitdehnung

- Ohne Tonhöhen-änderungen:
 - Kompression: Samples auslassen
 - Expansion: öfters abspielen

Time stretching ($\alpha=1$), time domain signalsTime stretching ($\alpha=1$), spectra

Synchronous Overlap and Add

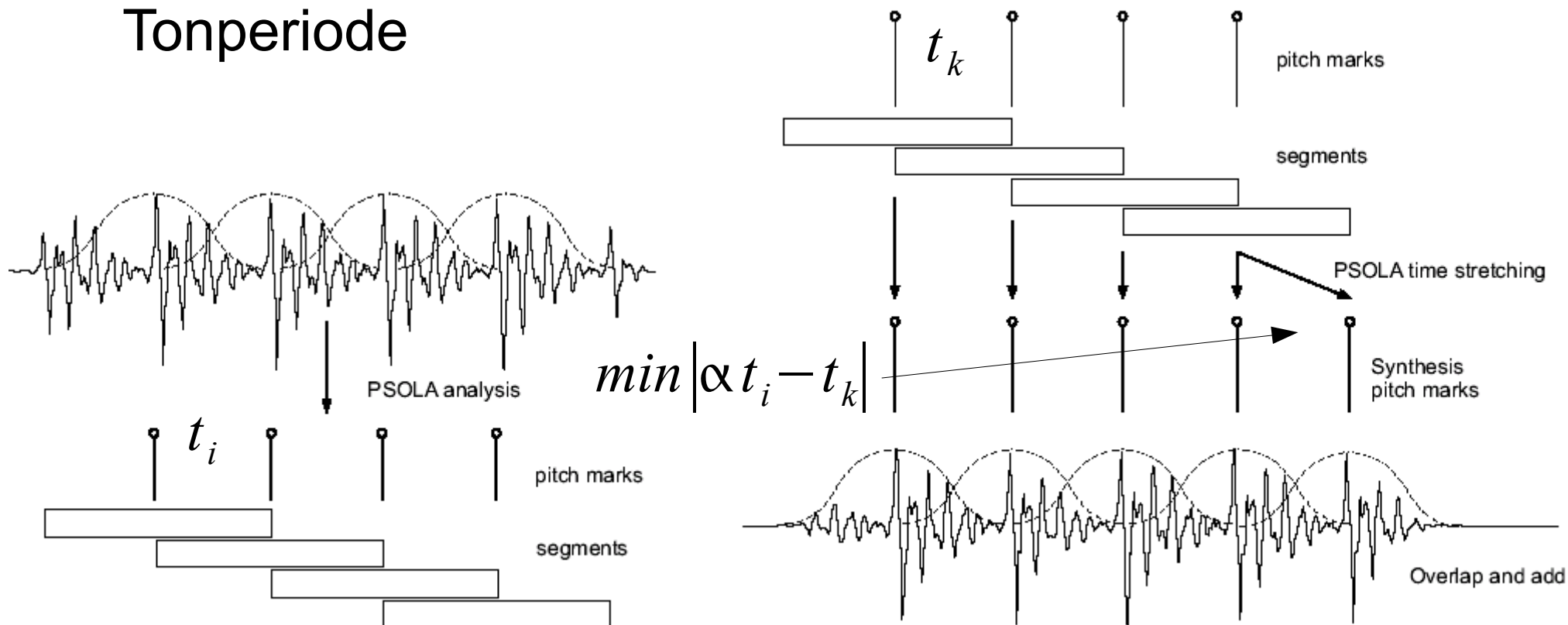
- SOLA:
 - Basiert auf Korrelation
 - Segmentierung des Signals in Blöcke S_a , Länge N
 - Verschiebung gemäß $S_s = \alpha \cdot S_a$
 - Kreuzkorrelation zw. den überlappenden Blöcken
 - Maximum der Korrelation: k_m
 - Fade in/out ab/bis k_m
 - Segmente überlappen



Pitch-synchronous Overlap and Add

- PSOLA:

- SOLA: Schwankungen in der Tonhöhe
- PSOLA: Synchronisation der Segmentierung auf die Tonperiode



Tonhöhenänderung

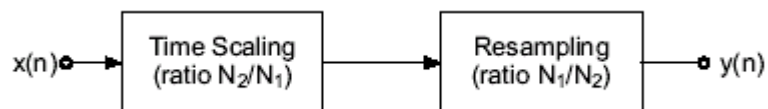
- Abspielgeschwindigkeit ändern: Pitch shifting

- Nachteil: Signallänge ändert sich

- Lösung: Resampling

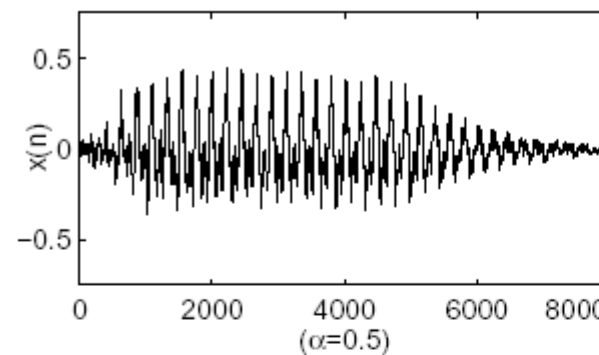
- Skalierung:

$$f_i^{new} = \alpha \cdot f_i^{old} \quad \alpha = \frac{N_2}{N_1}$$

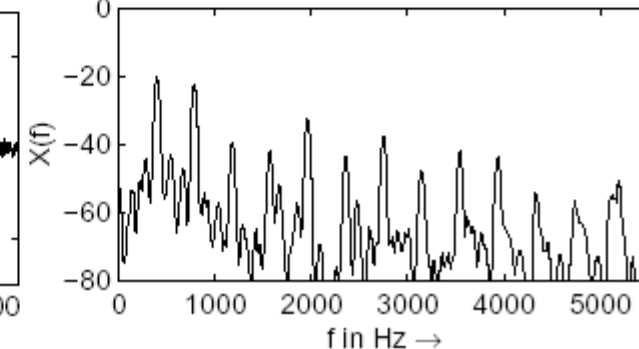
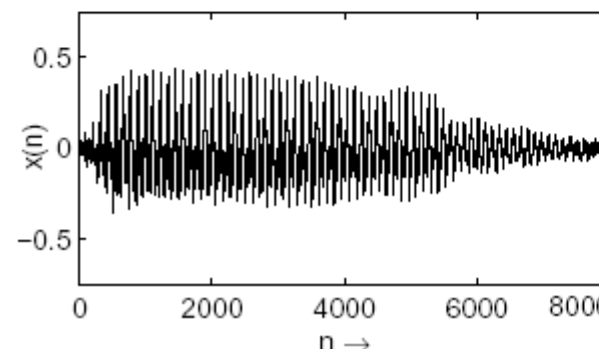
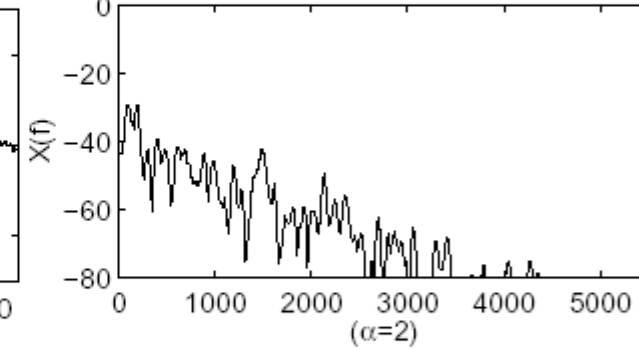
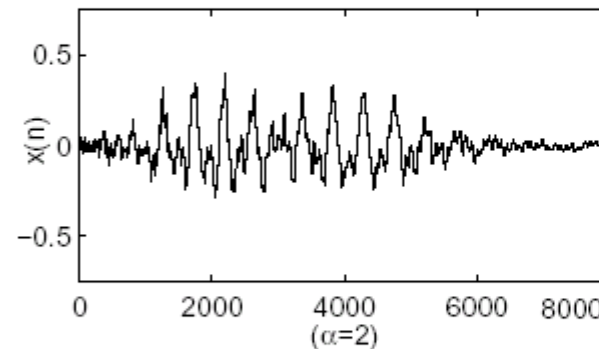
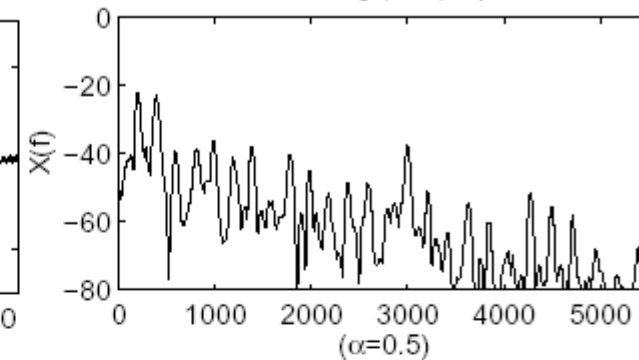


z.B.: SOLA

Pitch scaling ($\alpha=1$), time domain signals



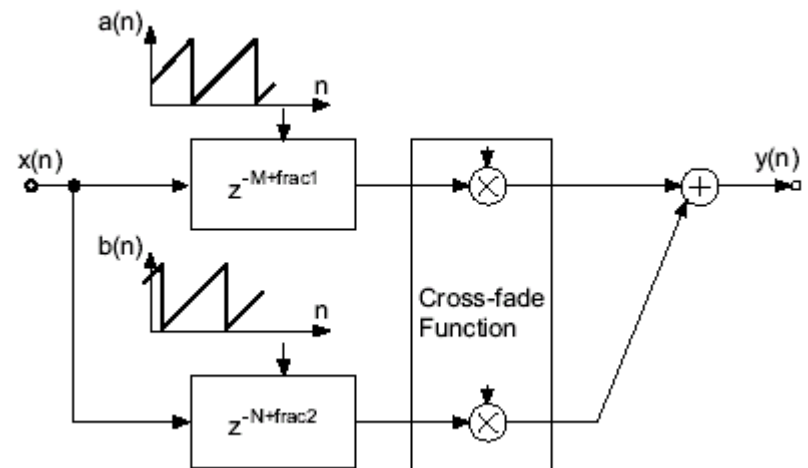
Pitch scaling ($\alpha=1$), spectra



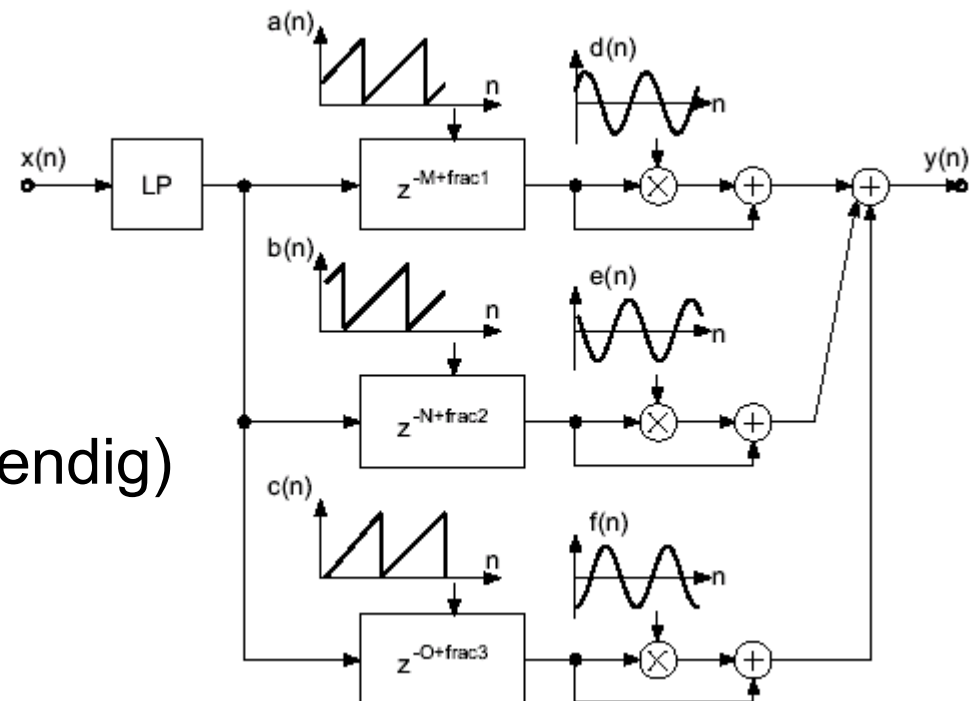
Tonhöhenänderung

- Delay Line Modulation:

Pitch detection für
Cross-fade notwendig:

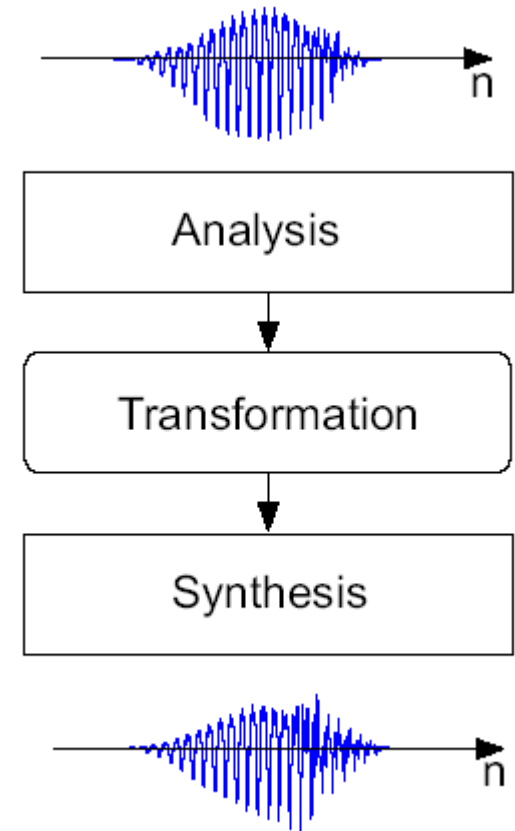


120°-Phasenversatz:
(Pitch detection nicht notwendig)



Time-Frequency-Processing

- Ziel Signaltransformation in der Zeit-Frequenebene
- Analyse:
 - Kurzzeit-Fourier-Transformation – Short Time Fourier Transformation (STFT)
- Synthese:
 - Inverse STFT



STFT

- Kontinuierlich:

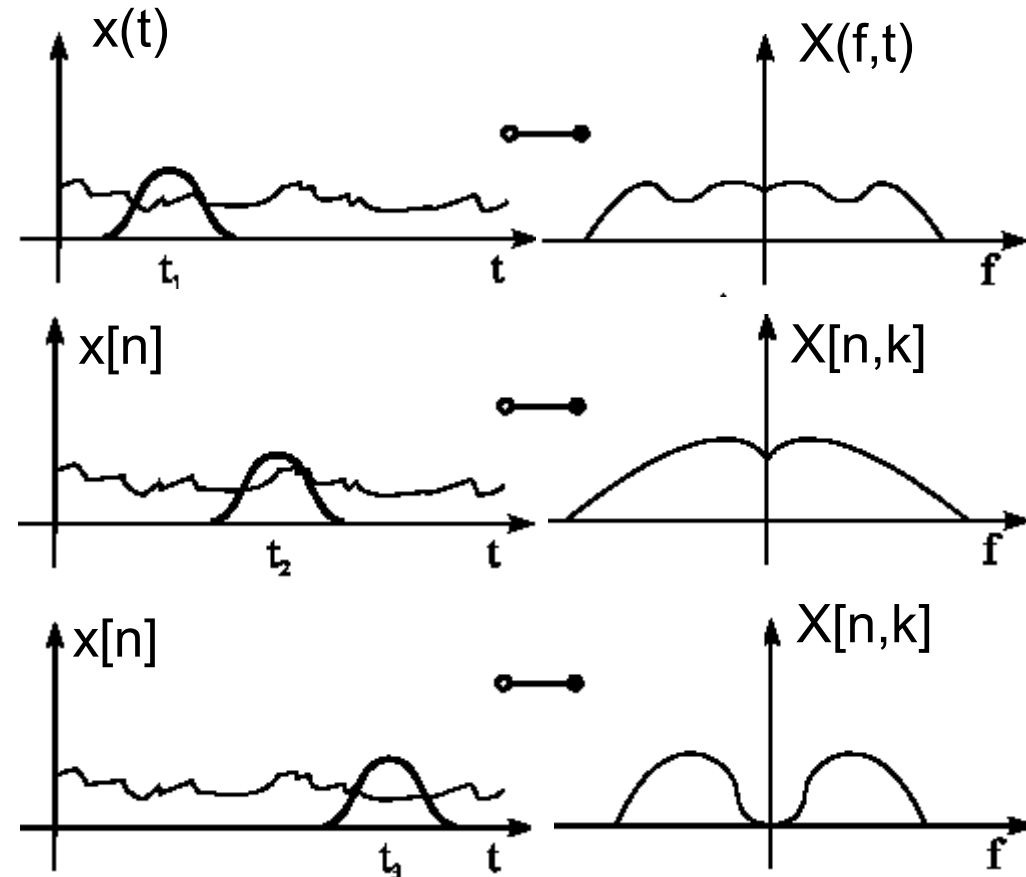
$$X(f, \tau) = \int_{-\infty}^{+\infty} x(t) w(t - \tau) \cdot e^{-2\pi j f t} dt$$

- Diskret:
Gabor-Transformation

$$X[k, n] = \sum_{m=0}^{M-1} x[m] w[m-n] \cdot e^{-2\pi j m k / M}$$

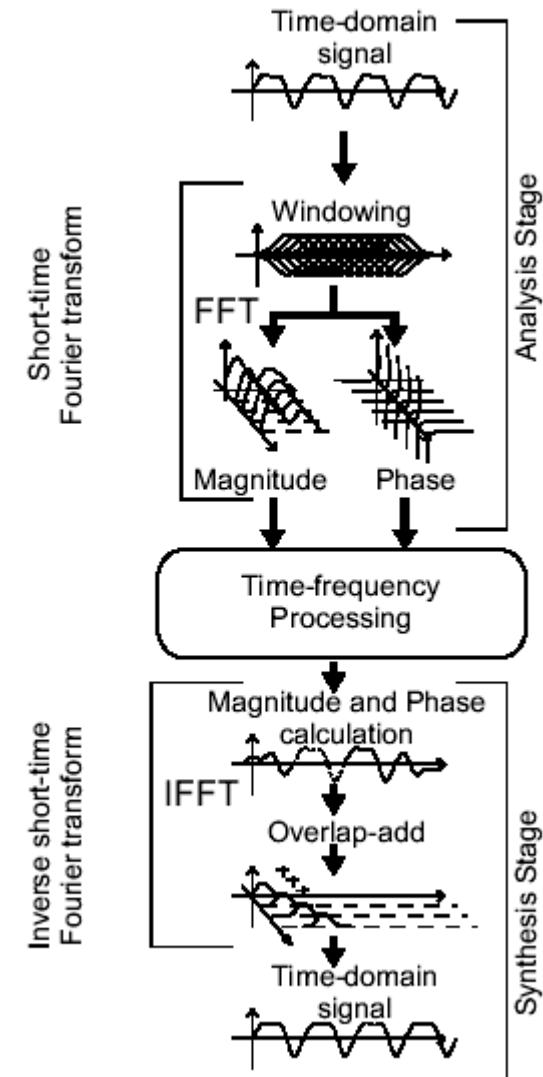
$$X_{n,k} = \sum_{m=0}^{M-1} x[m] w[m-n] \cdot W^k$$

mit: $W^k = e^{-\frac{2\pi j m}{M}}$, $j^2 = -1$



Phasenvocoder

- Analyse (STFT)
- Schätzung der Momentanfrequenzen aus der Phase
- Synthese (ISTFT)
- Ziel: Perfekte Rekonstruktion



Phasenvocoder

- Filterbank-Summationstechnik:

Analyse:

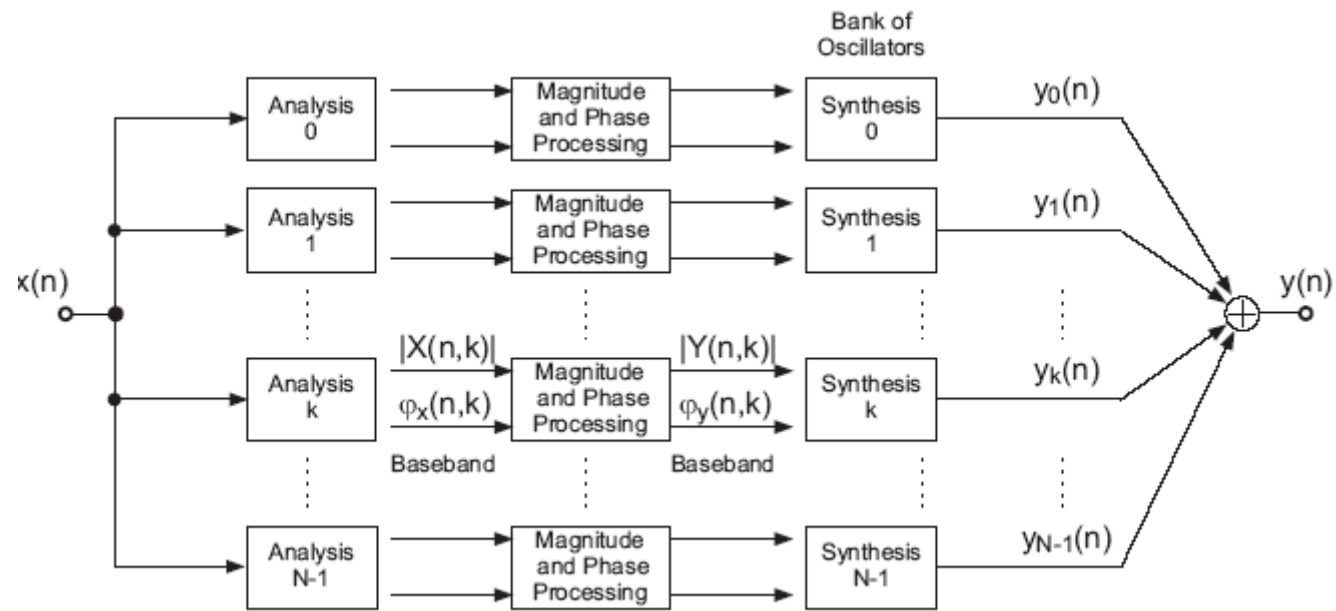
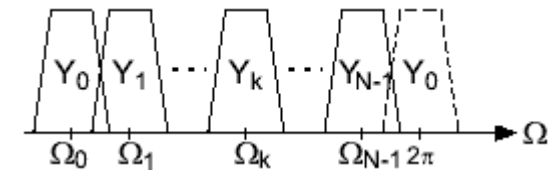
$$X_{n,k} = (h * x)[n] e^{-j\Omega_k n}$$

Transformation:

$$X_{n,k} \longrightarrow Y_{n,k}$$

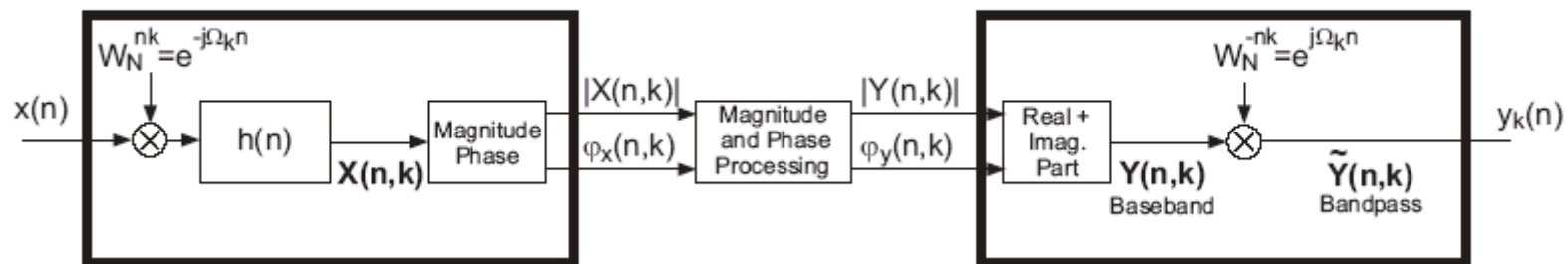
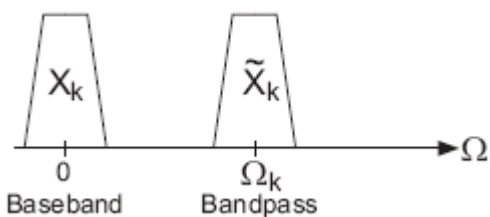
Synthese:

$$\tilde{Y}_{n,k} = Y_{n,k} e^{j\Omega_k n}$$



Synthesis:
Oscillator with variable magnitude and phase

Analysis

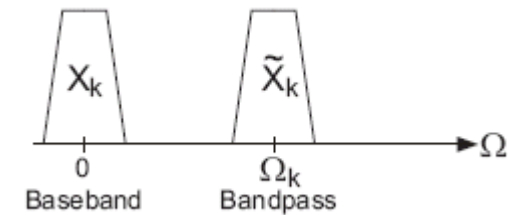


Phasenvocoder

- Phase:

- Baseband: $\varphi_k[n]$

- Bandpass: $\tilde{\varphi}_k[n] = \Omega_k \cdot n + \varphi_k[n]$



- Momentanfrequenz: $f_k[n] \longrightarrow \Omega_k[n] = \frac{2\pi}{f_s} f[n]$

- Signal als FM: $X[n, k] = \cos \tilde{\varphi}_k[n]$

mit $\Omega_k[n] = \frac{\partial \varphi_k[n]}{\partial n}$: $\tilde{\Omega}_k[n] = \Omega_k + \frac{d\varphi_k[n]}{dn}$

mit $dn=1$: $\tilde{\Omega}_k[n] = \Omega_k + \varphi_k[n] - \varphi_k[n-1]$

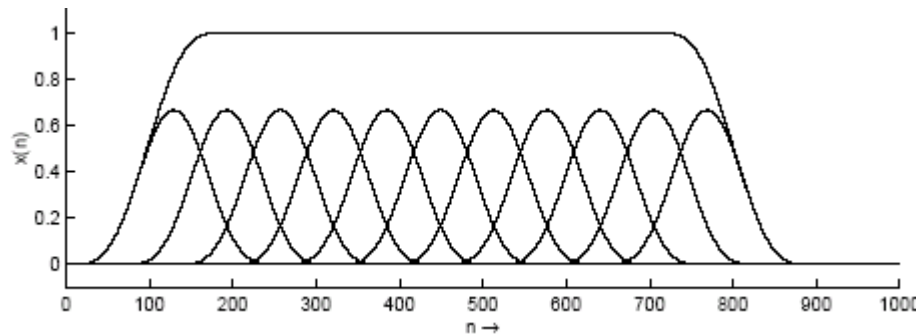
mit $\Omega_k = 2\pi \frac{k}{N}$: $f_k[n] = f_s \left(\frac{k}{N} + \frac{\varphi_k[n] - \varphi_k[n-1]}{2\pi} \right)$

Phasenvocoder

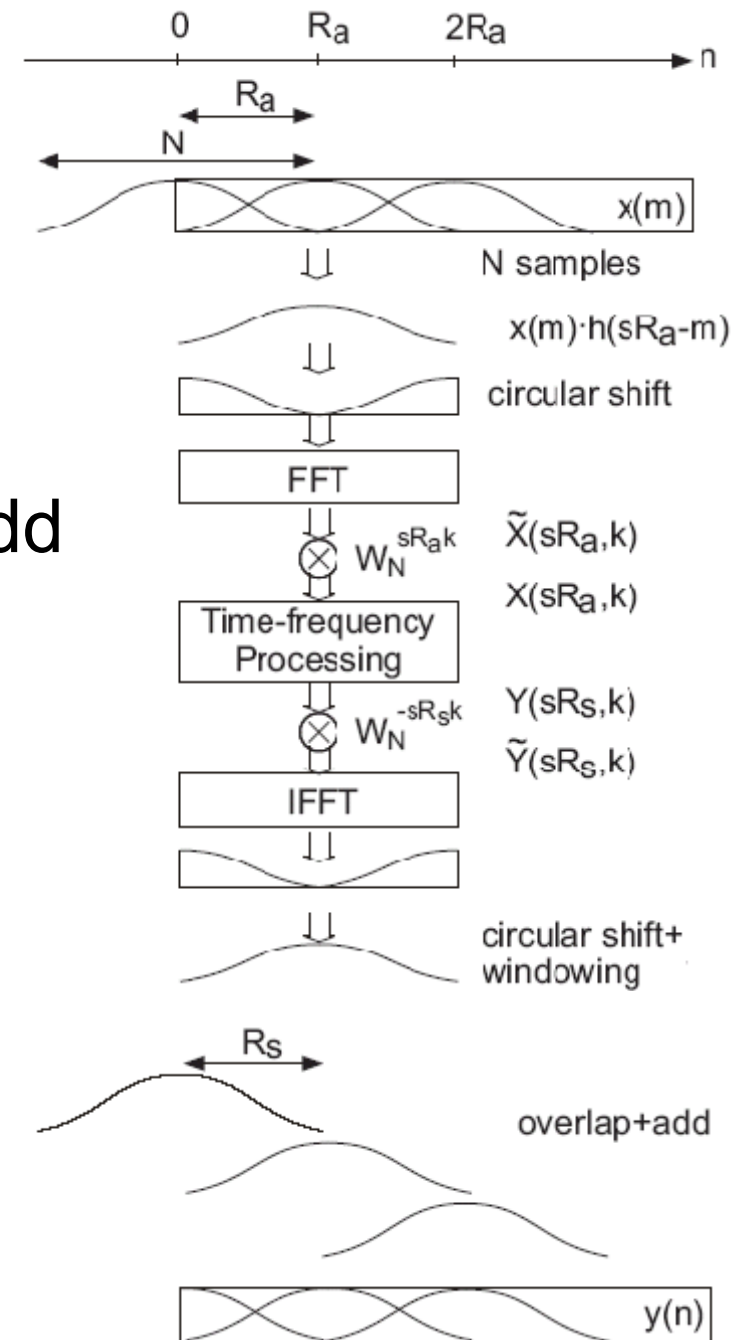
- Blockfiltertechnik

- Analyse: Fensterung, FT
- Synthese: FT, Fenster, Overlap+Add
- perfekte Rekonstruktion:

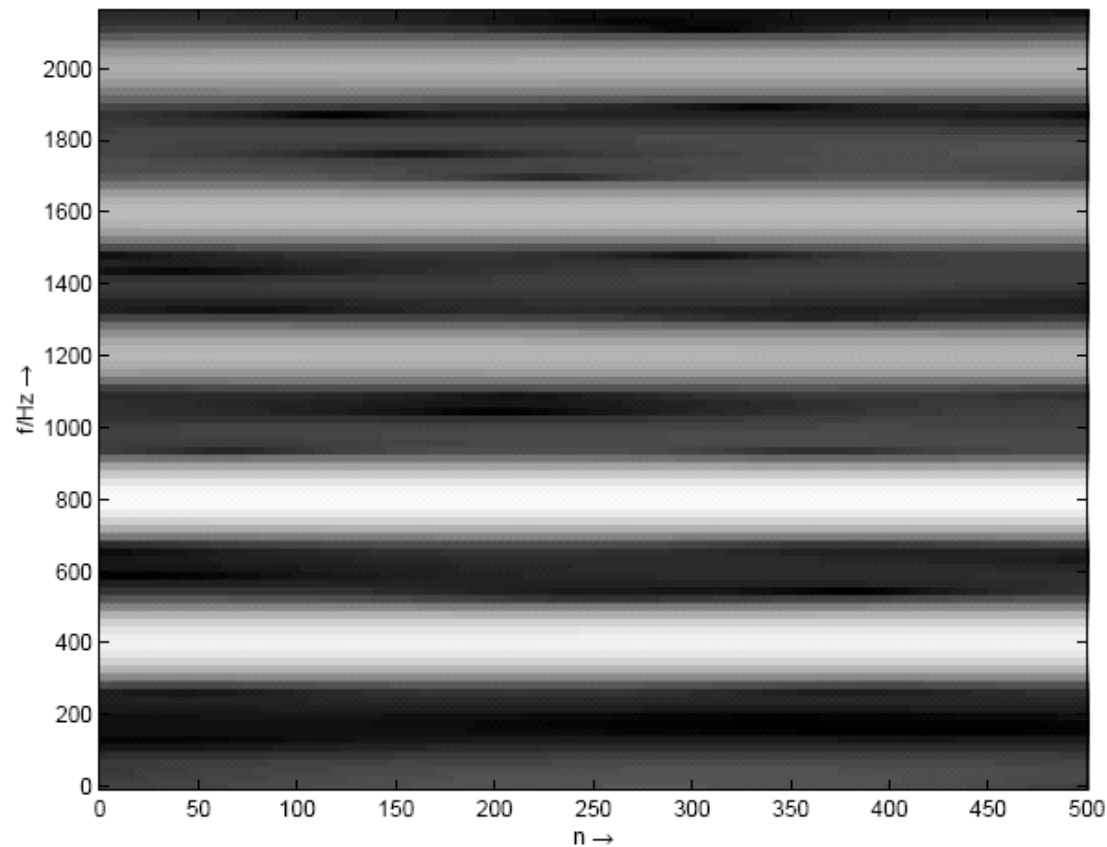
z.B.: Summe der Fenster = 1



$$X[sR_a, k] = \sum_{m=-\infty}^{\infty} x[m] h[sR_a - m] W^{mk}$$

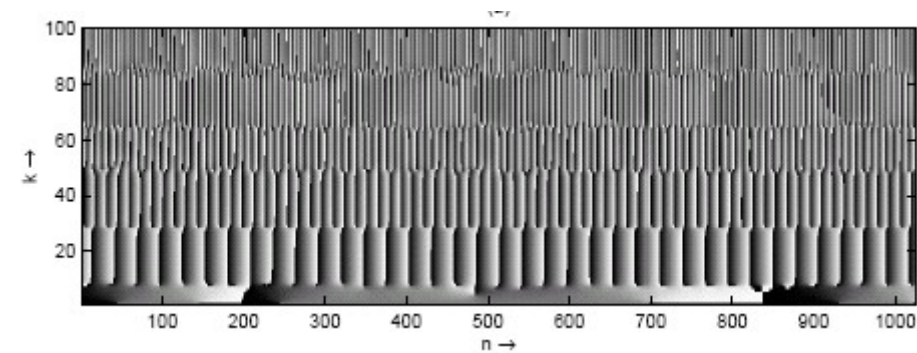


Spectrogram - Phasogram



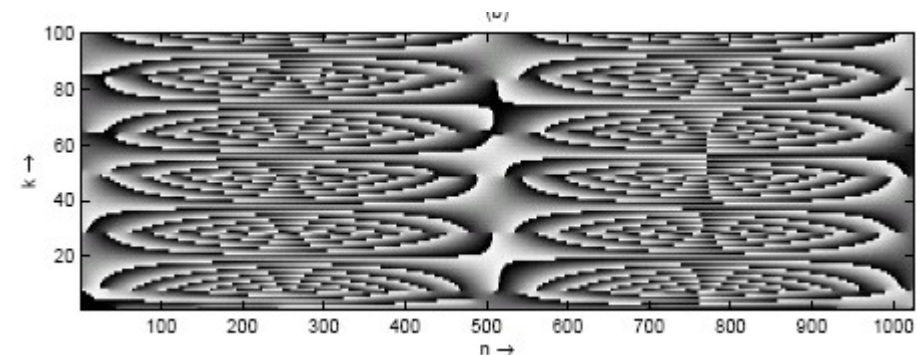
Signal: Flöte

Absolute Phase: $\tilde{\Phi}_{n,k}$



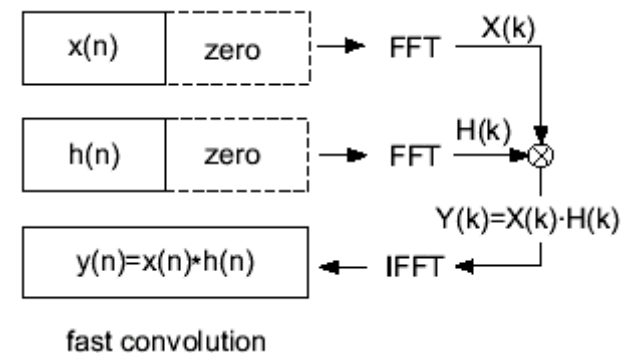
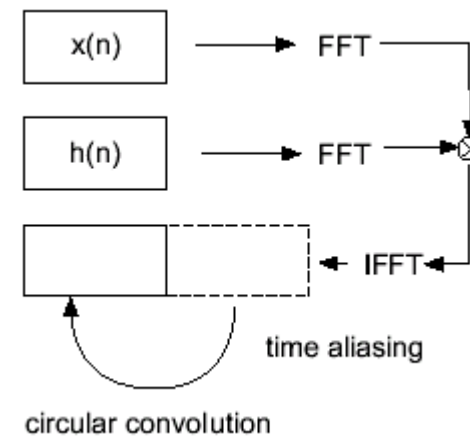
Demodulierte Phase:

$$\tilde{\Phi}_{n,k} - 2\pi m k / N$$



Transformationen

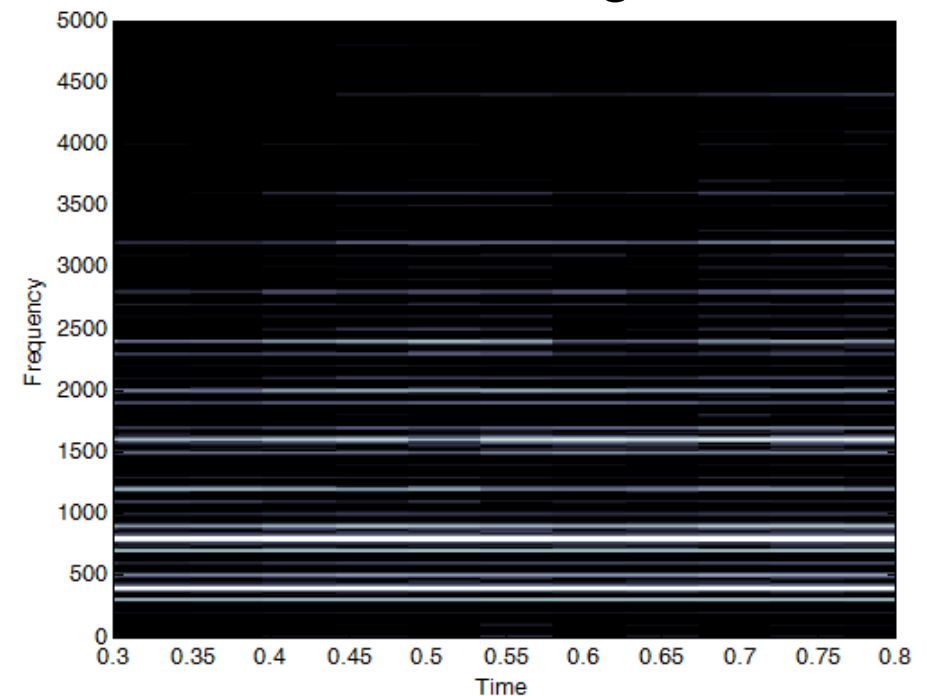
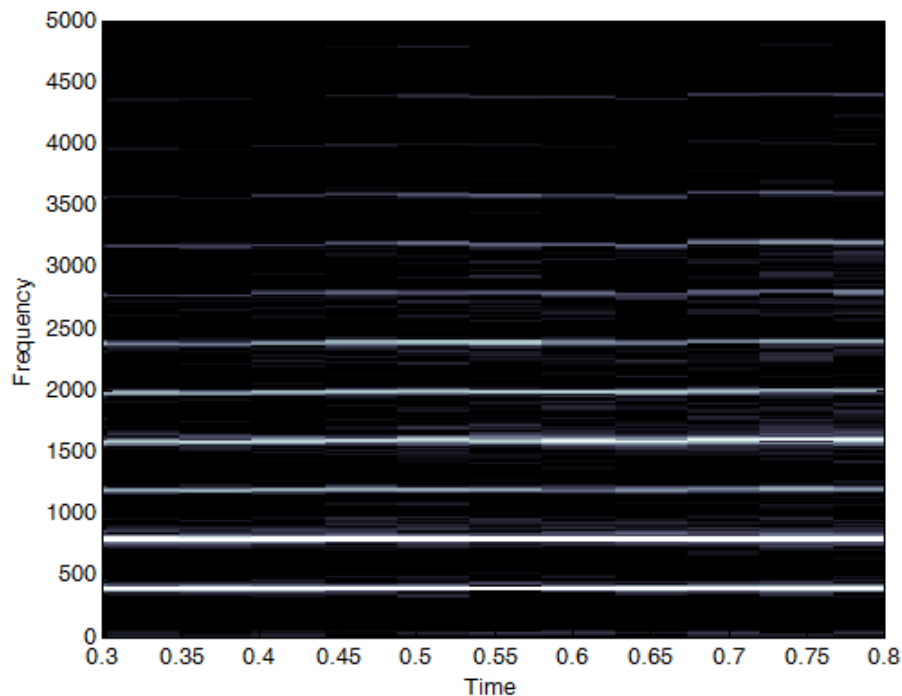
- Klangfarbe
- Dispersion
- Time stretching
- Pitch shifting
- Effekte:
 - Robotization
 - Mutation
 - Whisperization
 - Denoising



Robotization

- Phase auf 0 setzen:
 - fixe Tonhöhe in jedem Band
 - Gruppierung aller Frequenzen in einem Zeitpunkt
 - Kammfiltereffekte

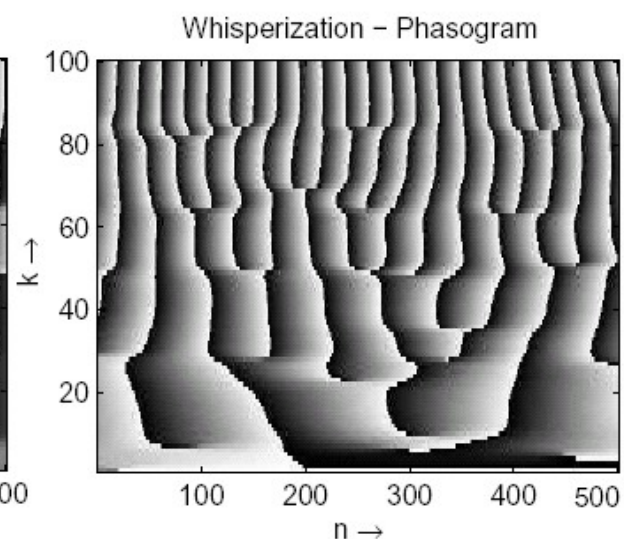
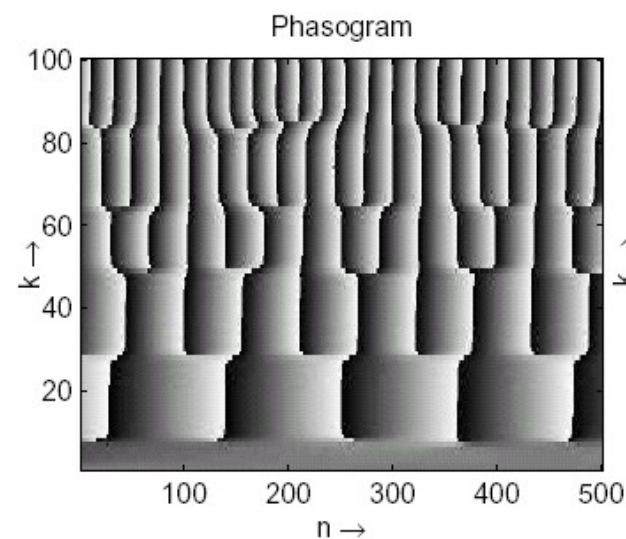
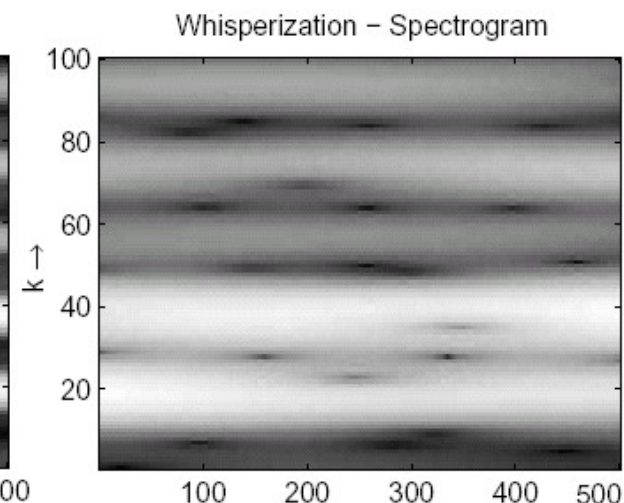
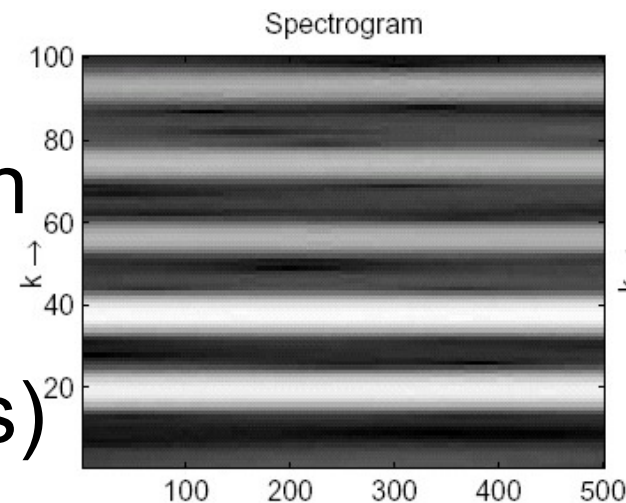
Signal: Flöte



Whisperization

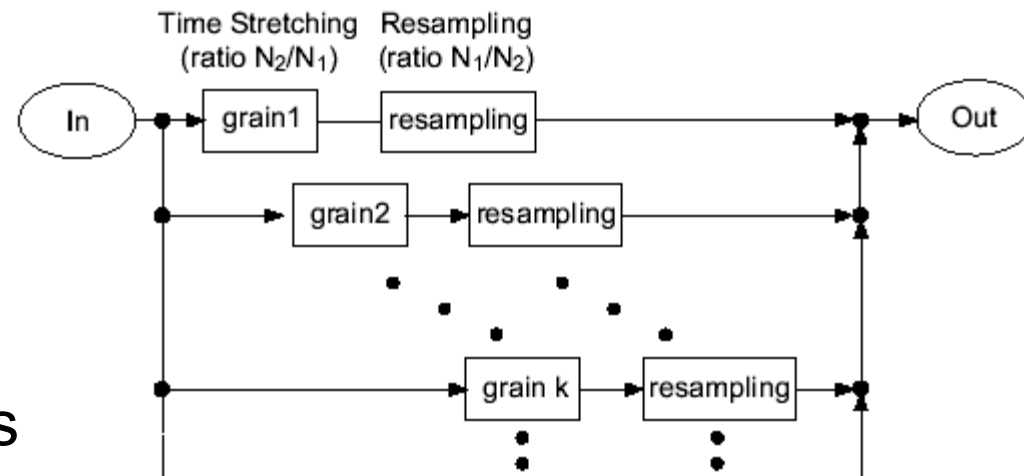
- Ziel: Sprachsignal als Geflüster
- Leichter Zufälligkeiten in der Phase
- kleine Hopsize ($< 2\text{ms}$)
 - sonst: Bewegung in den Harmonischen

Signal: Flöte



Pitch shifting

- Multiplikation jeder Frequenz mit α_f
- Filterbanktechnik: $\cos(\Omega_k n + \varphi_k[n]) \longrightarrow \cos(\alpha_f \cdot \Omega_k n + \varphi_k[n])$
- Blockfiltertechnik:
 - Frame-Resampling:
 - Vor der IFFT:
 - $\alpha_f > 1$: IFFT der ersten bins
 - $\alpha_f < 1$: Zero padding bis zu $f_s/2$



Time stretching

- Filterbanktechnik: $\cos(\Omega_k n + \varphi_k[n]) \longrightarrow \cos(\Omega_k n + \alpha_t \cdot \varphi_k[n])$

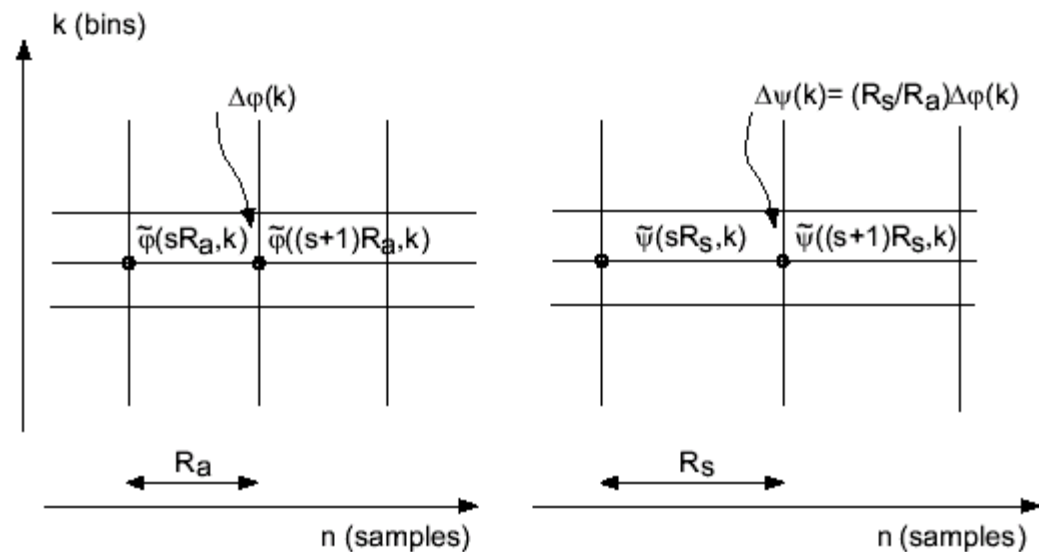
- Vorgehensweise:

$$X = |X_k| \cdot e^{j\varphi_k}$$

$$\alpha_t = R_s / R_a$$

$$\psi_k = \tilde{\varphi}_k \cdot \alpha_t$$

$$\longrightarrow Y = |X_k| \cdot e^{j\psi_k}$$

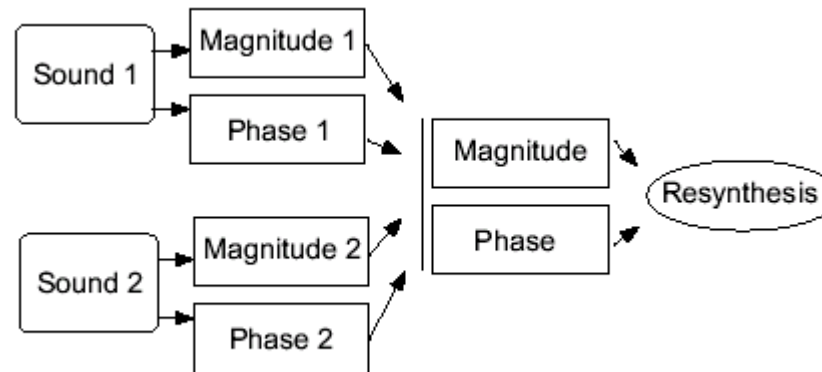


- Unwrapping der Phase
- Allpass-Charakteristik
- Fenster für perfekte Rekonstruktion

R_a ... analysis hop size
 ψ ... synthesis phase
 φ_k ... analysis phase

Mutation

- Idee: 2 Signale verschmelzen:

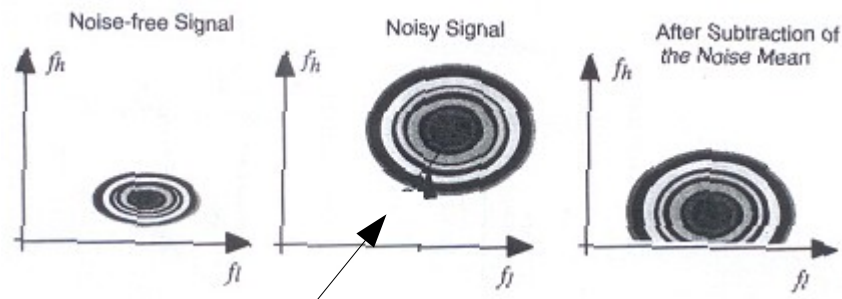


- Kein Morphing (=Veränderung der Merkmale)!
- Möglichkeiten:
 - Überlagerung der Amplituden (kein Abmischen!)
 - Maskierung der Amplituden
 - Variation der Phasen (Pitchinformation! Unwrapping!)

Denoising

- Ziel: Rauschen N aus einem verrauschten Signal X zu entfernen: $X[k] = S[k] + N[k]$

- spektrale Subtraktion: $|S[k]|^b = |X[k]|^b - \alpha |\bar{N}[k]|^b$
 - Phase bleibt gleich $b = 1, 2 \dots$ Betrachtungsweise
 - Rauschen an “signalfreien” Stellen ermitteln $\alpha \dots \begin{cases} 1 & \dots \text{Vollständige "Subtraktion"} \\ > 1 & \dots \text{Übersubtraktion} \end{cases}$
 - negative Amplituden vermeiden: $|X_{min}[k]|^b$ einhalten



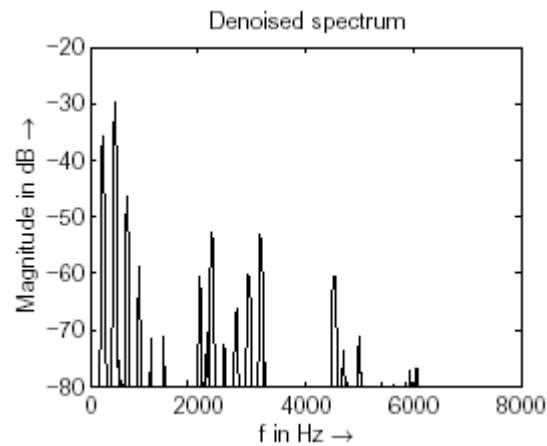
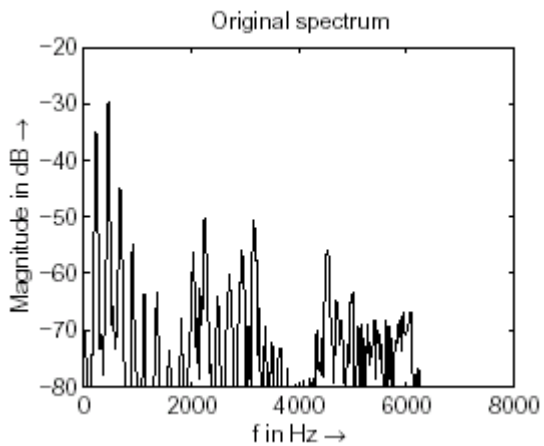
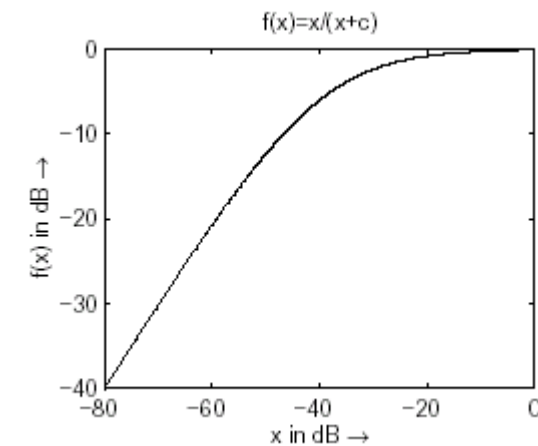
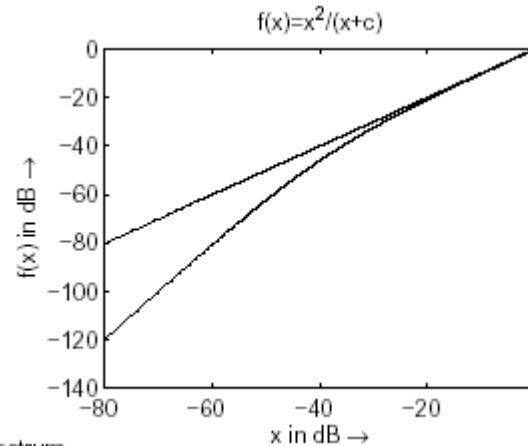
Varianzerhöhung aufgrund von Rauschen

Denoising

- Spektrales Noise Gating:
 - Nur Amplituden über einer Schwelle durchlassen
 - kleine Amplituden stärker abschwächen:

$$Y_{n,k} = f[|X_{n,k}|] \cdot X_{n,k}$$

$$f(x) = \frac{x}{(x+c)}$$



Alternative: $f(x) = f(SNR)$

Weiterführende Literatur

- M. Dolson (1987). The Phase Vocoder: A Tutorial. Computer Music Journal 1987 10(4):14-27
- H. G. Feichtinger and T. Strohmer, Gabor analysis and algorithms (Birkhäuser, Boston, 1998)
- H. G. Feichtinger and T. Strohmer, Advances in Gabor analysis (Birkhäuser, Basel, 2003)
- P. Balazs, H. G. Feichtinger, M. Hampejs, and G. Kracher, Double preconditioning for gabor frames. IEEE Trans Sig Proc, vol. 54, pp. 4597-4610, 2006.