Pattern Recognition

Part 4: Feature Extraction

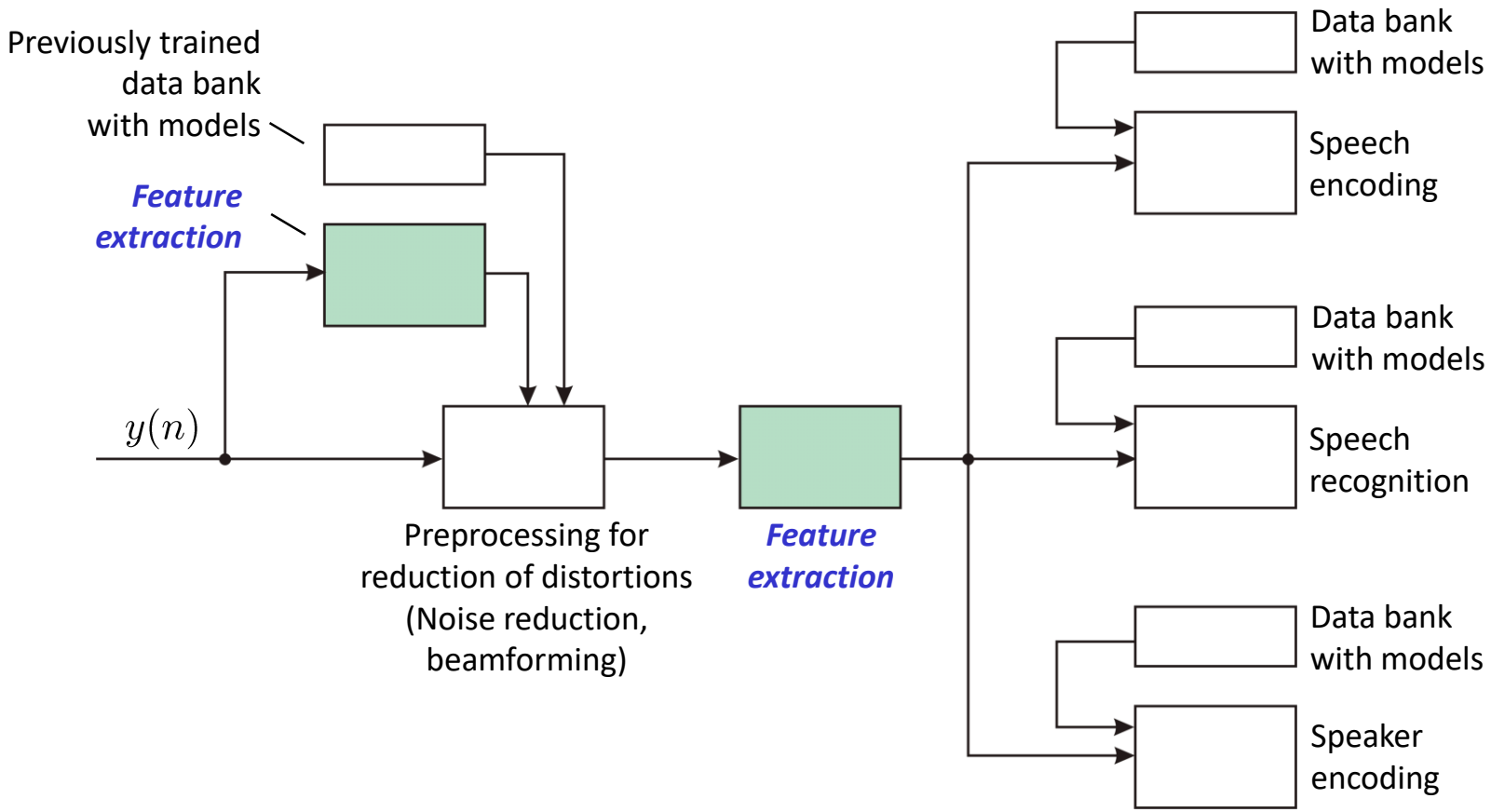
Gerhard Schmidt

Christian-Albrechts-Universität zu Kiel
Faculty of Engineering
Institute of Electrical and Information Engineering
Digital Signal Processing and System Theory

Contents

- Introduction
- Features for speech and speaker recognition
  - Fundamental frequency
  - Spectral envelope
- Representation of the spectral envelope
  - Predictor coefficients
  - Cepstral coefficients
  - Mel-filtered cepstral coefficients (MFCCs)

Introduction



Literature

Estimation of the fundamental frequency

- W. Hess: ***Pitch Determination of Speech Signals: Algorithms and Devices***, Springer, 1983

Prediction

- M. S. Hayes: ***Statistical Digital Signal Processing and Modeling*** – Chapter 4 and 5 (Signal Modeling, The Levinson Recursion), Wiley, 1996
- E. Hänsler, G. Schmidt: ***Acoustic Echo and Noise Control*** – Chapter 6 (Linear Prediction), Wiley, 2004

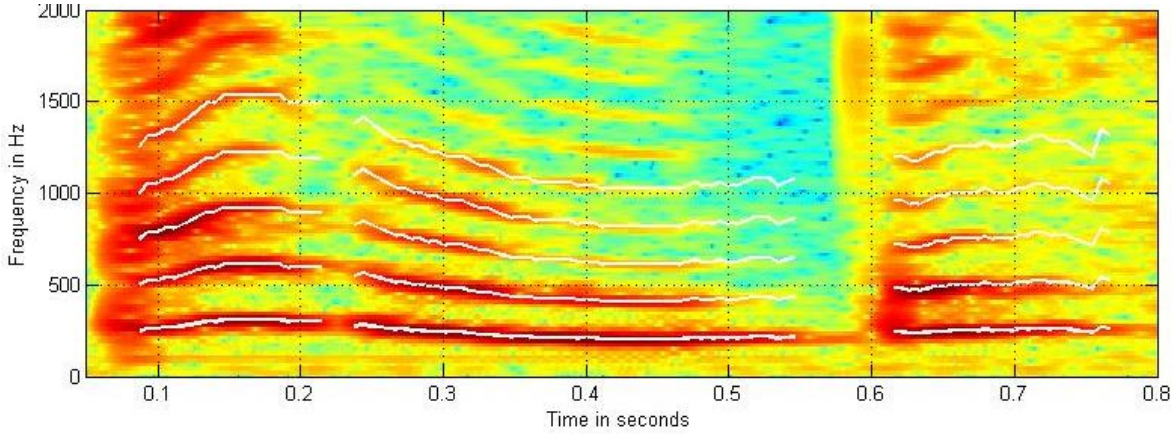
Mel-filtered cepstral coefficients

- E Schukat-Talamanzzini: ***Automatische Spracherkennung – Grundlagen, statistische Modelle und effiziente Algorithmen***, Vieweg, 1995 (in German)
- L. Rabiner, B.-H. Juang: ***Fundamentals of Speech Recognition***, Prentice-Hall, 1993

Features for Speech and Speaker Recognition – Fundamental Frequency

Fundamental frequency:

- Feature extraction mostly with ***autocorrelation based methods***.
- Used for (rough) ***discrimination between male, female, and children's speech***.
- The contour of the fundamental frequency be used for estimating ***accentuations in speech*** (helpful for recognizing questions, grouped phone numbers) or the ***emotional state of the speaker***.



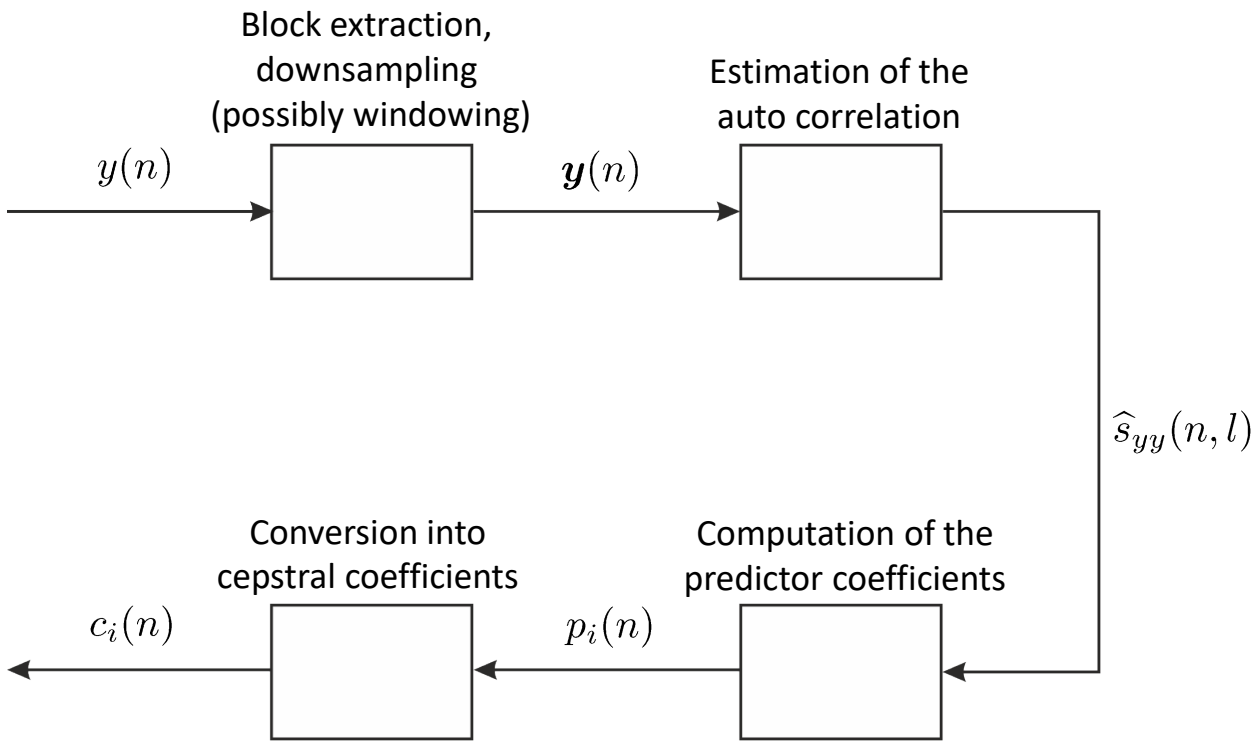
- Certain types of ***noise*** can be distinguished from speech by estimating the fundamental frequency (e.g. „GSM buzz")
- It can be of advantage to „***normalize***" the frequency axis to the average fundamental frequency of a speaker.

Features for Speech and Speaker Recognition – Spectral Envelope

Spectral envelope

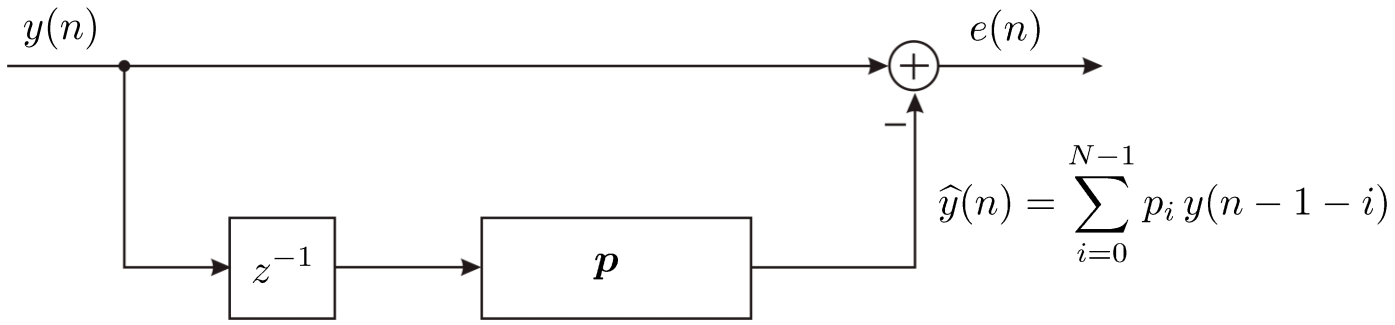
- The spectral envelope is currently the ***most important feature in speech and speaker recognition.***
- The spectral envelope is extracted ***every 10 to 20 ms*** and then used in subsequent algorithms such as speech recognition or coding.
- In order to reduce the computational complexity of the subsequent signal processing, the envelope should be computed **compact** (with a low number of relevant parameters) and in a form that a suitable for a cost function.
- Some signal processing techniques (e.g. bandwidth extension, speech reconstruction) need a representation of the spectral envelope that can also be ***used in the signal path***. Other methods (e.g. speech and speaker recognition) are not bound to this condition.
- Typically, either ***cepstral coefficients***, so called *mel-filtered cepstral coefficients* or *mel-frequency cepstral coefficients* (***MFCCs***) are used.

Representation of the Spectral Envelope Using Cepstral Coefficients



Predictor Error Filter – Part 1

Structure of a prediction error filter:



Cost function for optimizing the coefficients:

$$\mathrm{E}\Big\{ \big[\underbrace{y(n) - \widehat{y}\,(n)}_{e(n)} \big]^2 \Big\} \longrightarrow \min$$

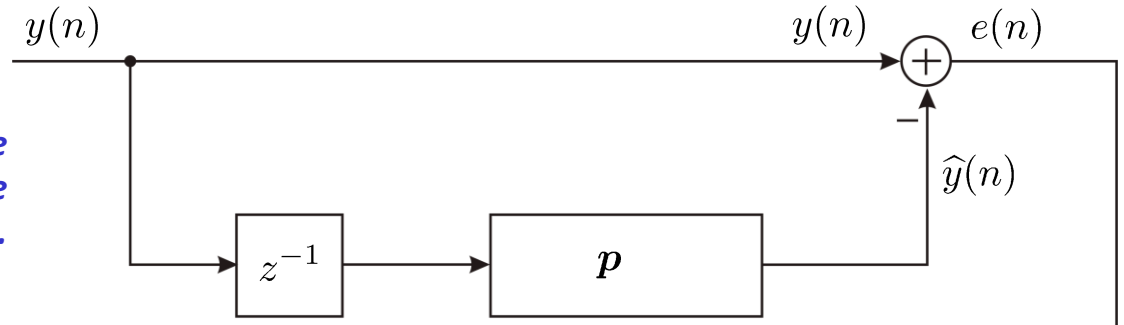
Frequency components with high signal power will be attenuated first (Parseval).

This causes spectral flattening (whitening) of the spectrum.

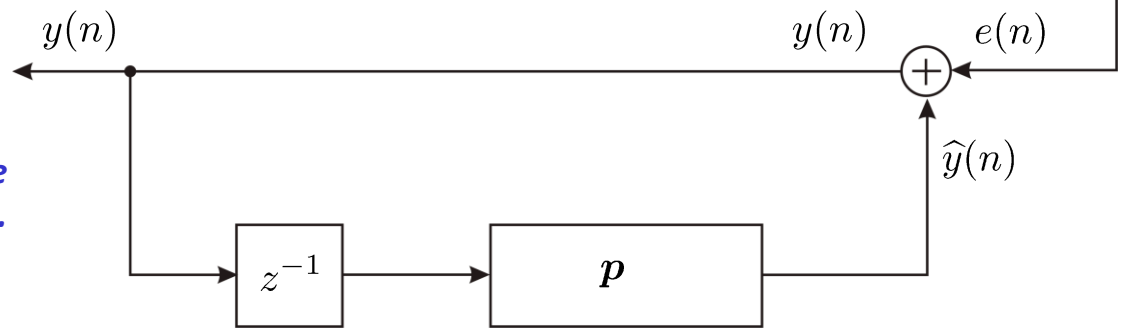
Predictor Error Filter – Part 2

Structure of a prediction error filter and an inverse filter:

The FIR version of the filter removes the spectral envelope.

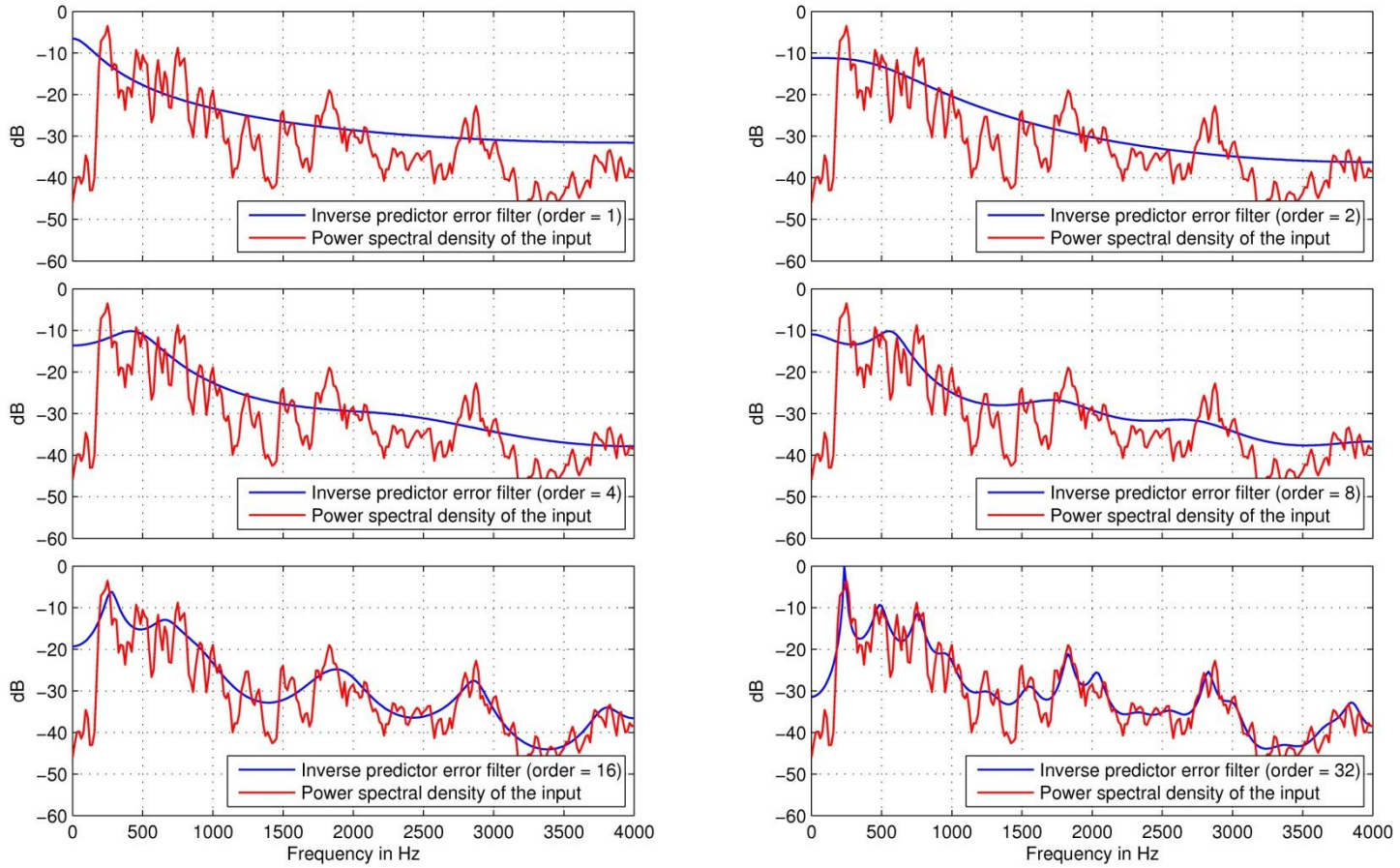


The IIR version of the filter reconstructs it.



Predictor Error Filter – Part 3

Frequency responses of inverse predictor error filters:



Typically, prediction orders between 10 and 20 are used for representing the spectral envelope.

Computation of the Predictor Coefficients – Part 1

Derivation:

- Cost function

$$\mathrm{E}\Big\{e^2(n)\Big\} = \mathrm{E}\Big\{\big[y(n) - \widehat{y}(n)\big]^2\Big\} \underset{\boldsymbol{p}=\boldsymbol{p}_{\mathrm{opt}}}{\longrightarrow} \min$$

- Error signal:

$$\begin{aligned} e(n) &= y(n) - \widehat{y}\,(n) \\ &= y(n) - \sum_{i=0}^{N-1} p_i\, y(n-i-1) \end{aligned}$$

- Differentiating the cost function:

$$\begin{aligned} \frac{\partial \mathrm{E}\{e^2(n)\}}{\partial p_i} &= \mathrm{E}\Big\{2\,e(n)\frac{\partial e(n)}{\partial p_i}\Big\} = \mathrm{E}\left\{2\,e(n)\,\frac{\partial\Big(y(n) - \sum_{j=0}^{N-1} y(n-1-j)\,p_j\Big)}{\partial p_i}\right\} \\ &= -2\,\mathrm{E}\Big\{e(n)\,y(n-1-i)\Big\} = -2\,\mathrm{E}\left\{\left(y(n) - \sum_{j=0}^{N-1} y(n-1-j)\,p_j\right) y(n-1-i)\right\} \end{aligned}$$

$$\text{for} \quad i \in \{0,\, ...,\, N-1\}$$

Computation of the Predictor Coefficients – Part 2

Derivation:

- Differentiating the cost function resulted in:

$$\frac{\partial \mathrm{E}\{e^2(n)\}}{\partial p_i} = -2\,\mathrm{E}\left\{\left(y(n) - \sum_{j=0}^{N-1} y(n-1-j)\,p_j\right) y(n-1-i)\right\}$$
$$\text{for} \quad i \in \{0, \ldots, N-1\}$$

- Setting the derivative to zero:

$$0 = \mathrm{E}\left\{\left(y(n) - \sum_{j=0}^{N-1} y(n-1-j)\,p_{\mathrm{opt},j}\right) y(n-1-i)\right\}$$
$$= s_{yy}(i+1) - \sum_{j=0}^{N-1} p_{\mathrm{opt},j}\, s_{yy}(i-j)$$
$$\text{for} \quad i \in \{0, \ldots, N-1\}$$

Computation of the Predictor Coefficients – Part 3

Derivation:

- Setting the derivative to zero resulted in:

$$0 = s_{yy}(i+1) - \sum_{j=0}^{N-1} p_{\mathrm{opt},j}\, s_{yy}(i-j)$$
$$\text{for} \quad i \in \{0, \ldots, N-1\}$$

- Equation system with *N* equations:

$$\begin{aligned}
s_{yy}(1) &= p_0\, s_{yy}(0) + p_1\, s_{yy}(1) + \ldots + p_{N-1}\, s_{yy}(N-1) \\
s_{yy}(2) &= p_0\, s_{yy}(1) + p_1\, s_{yy}(0) + \ldots + p_{N-1}\, s_{yy}(N-2) \\
s_{yy}(3) &= p_0\, s_{yy}(2) + p_1\, s_{yy}(1) + \ldots + p_{N-1}\, s_{yy}(N-3) \\
&\vdots \qquad\quad \vdots \qquad\quad \vdots \qquad\qquad\quad \vdots \\
s_{yy}(N) &= p_0\, s_{yy}(N-1) + p_1\, s_{yy}(N-2) + \ldots + p_{N-1}\, s_{yy}(0)
\end{aligned}$$
$$\text{for } p_i = p_{\mathrm{opt},i}$$

Computation of the Predictor Coefficients – Part 4

Derivation:

- Matrix-vector notation:

$$\underbrace{\begin{bmatrix} s_{yy}(1) \\ s_{yy}(2) \\ \vdots \\ s_{yy}(N) \end{bmatrix}}_{\boldsymbol{s}_{yy}(1)} = \underbrace{\begin{bmatrix} s_{yy}(0) & s_{yy}(1) & \dots & s_{yy}(N-1) \\ s_{yy}(1) & s_{yy}(0) & \dots & s_{yy}(N-2) \\ \vdots & \vdots & \ddots & \vdots \\ s_{yy}(N-1) & s_{yy}(N-2) & \dots & s_{yy}(0) \end{bmatrix}}_{\boldsymbol{S}_{yy}} \underbrace{\begin{bmatrix} p_{\mathrm{opt},0} \\ p_{\mathrm{opt},1} \\ \vdots \\ p_{\mathrm{opt},N-1} \end{bmatrix}}_{\boldsymbol{p}_{\mathrm{opt}}}$$

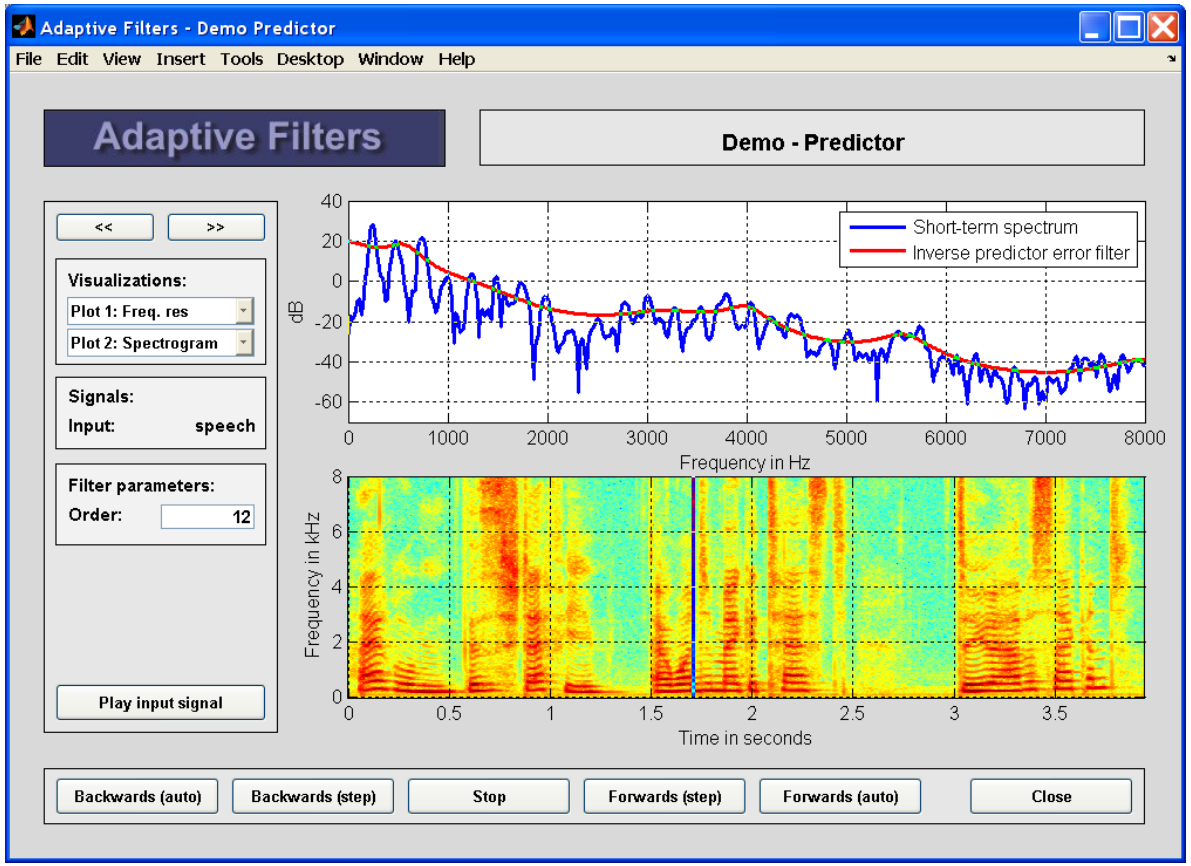
- Compact notation:

$$\boldsymbol{p}_{\mathrm{opt}} = \boldsymbol{S}_{yy}^{-1}\, \boldsymbol{s}_{yy}(1)$$

Computationally efficient and robust solution of the equation system e.g. using Levinson-Durbin-Recursion.

Computation of the Predictor Coefficients – Part 5

Matlab example:



Representation of the Spectral Envelope Using Cepstral Coefficients – Part 1

Requirements:

- A cost function should capture „***distances***“ between spectral envelopes. Similar envelopes should cause a small distance, envelopes that differ a lot should lead to large distances, and identical envelopes should cause a distance of zero.
- The cost function should be ***invariant to variations in the recording level/gain*** of the input signal.
- The cost function should be „easy“ to compute.
- The cost function should be similar to the ***human perception*** of sound (e.g. regarding the logarithmic loudness perception).

Ansatz:

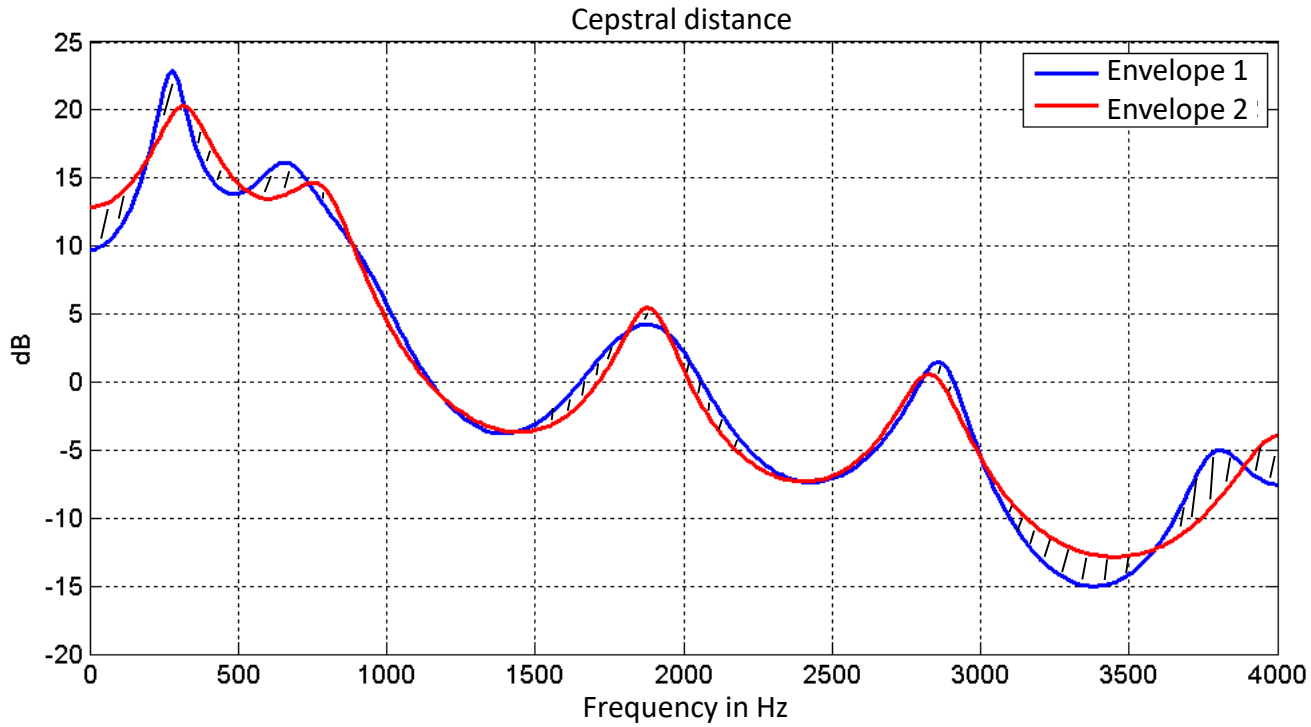
$$d_{\text{ceps}}(\ldots,\ldots) = \int_{\Omega=0}^{2\pi} \left| \ln\left\{P_{\text{inv},1}\left(e^{j\Omega}\right)\right\} - \ln\left\{P_{\text{inv},2}\left(e^{j\Omega}\right)\right\} \right|^2 d\Omega$$

Cepstral distance

Representation of the Spectral Envelope Using Cepstral Coefficients – Part 2

Ansatz:

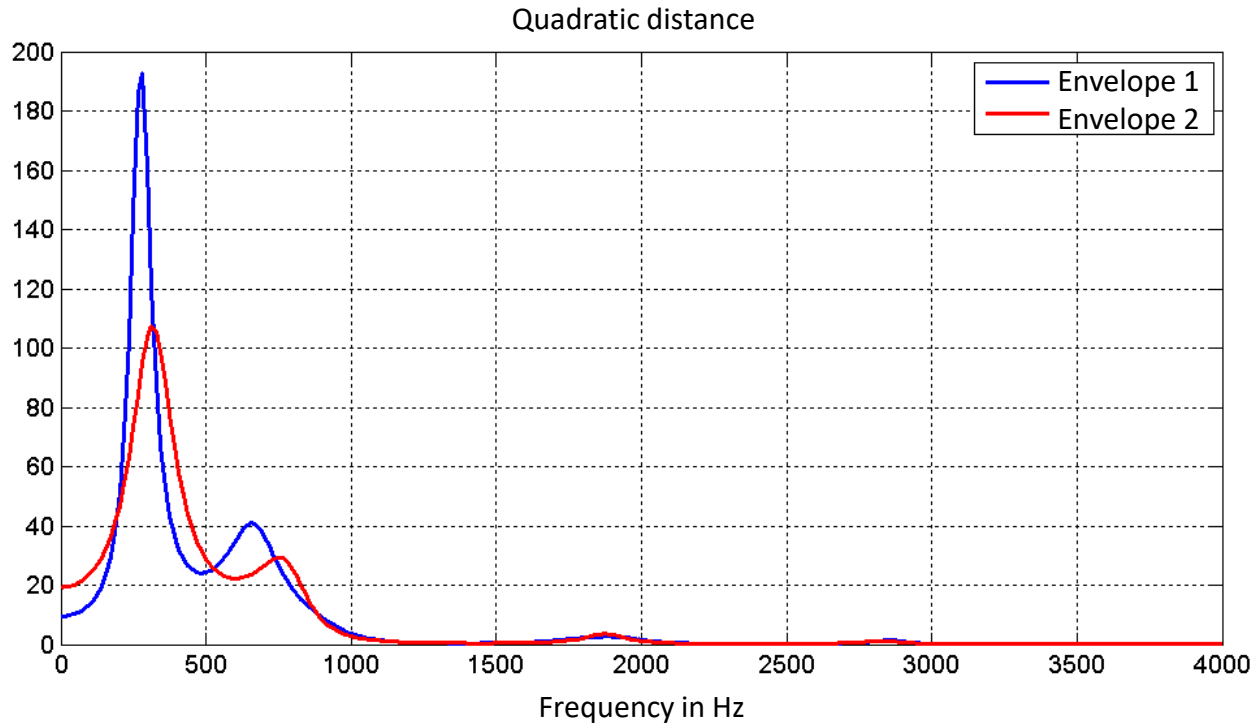
$$d_{\text{ceps}}(\ldots,\ldots) = \int_{\Omega=0}^{2\pi} \left| \ln\left\{P_{\text{inv},1}\left(e^{j\Omega}\right)\right\} - \ln\left\{P_{\text{inv},2}\left(e^{j\Omega}\right)\right\} \right|^2 d\Omega$$



Representation of the Spectral Envelope Using Cepstral Coefficients – Part 3

A well-known alternative – the quadratic distance:

$$d_{\text{quad}}(\ldots,\ldots) = \int_{\Omega=0}^{2\pi} \left| P_{\text{inv},1}\left(e^{j\Omega}\right) - P_{\text{inv},2}\left(e^{j\Omega}\right) \right|^2 d\Omega$$



Representation of the Spectral Envelope Using Cepstral Coefficients – Part 4

Cepstral distance:

$$d_{\text{ceps}}(\ldots,\ldots) = \int_{\Omega=0}^{2\pi} \left| \ln\left\{P_{\text{inv},1}\left(e^{j\Omega}\right)\right\} - \ln\left\{P_{\text{inv},2}\left(e^{j\Omega}\right)\right\} \right|^2 d\Omega$$

Parseval

$$d_{\text{ceps}}(\ldots,\ldots) = \sum_{i=-\infty}^{\infty} \left(c_{i,1} - c_{i,2}\right)^2$$

$$\approx \sum_{i=1}^{3/2\,N} \left(c_{i,1} - c_{i,2}\right)^2$$

$$c_i = \frac{1}{2\pi} \int_{\Omega=0}^{2\pi} \ln\left\{P_{\text{inv}}\left(e^{j\Omega}\right)\right\} e^{j\Omega i}\, d\Omega$$

$$\ln\{z\} = \ln|z| + j\arg\{z\}$$

Representation of the Spectral Envelope Using Cepstral Coefficients – Part 5

Computationally efficient transformation from prediction to cepstral coefficients:

- Definition

$$c_i = \frac{1}{2\pi} \int_{\Omega=0}^{2\pi} \ln\left\{P_{\text{inv}}\left(e^{j\Omega}\right)\right\} e^{j\Omega i}\, d\Omega$$

- Fourier-Transform for time-discrete signals and systems

$$\sum_{i=-\infty}^{\infty} c_i\, e^{-j\Omega i} = \ln\left\{P_{\text{inv}}\left(e^{j\Omega}\right)\right\}$$

- Replacing $e^{j\Omega}$ by z

$$\sum_{i=-\infty}^{\infty} c_i\, z^{-i}\Big|_{z=e^{j\Omega}} = \ln\left\{P_{\text{inv}}(z)\right\}\Big|_{z=e^{j\Omega}}$$

Representation of the Spectral Envelope Using Cepstral Coefficients – Part 6

Computationally efficient transformation from prediction to cepstral coefficients:

- Result so far

$$\sum_{i=-\infty}^{\infty} c_i \, z^{-i} \;=\; \ln\left\{P_{\mathrm{inv}}(z)\right\}$$

- Inserting the structure of the inverse prediction error filter

$$\begin{aligned}\sum_{i=-\infty}^{\infty} c_i \, z^{-i} &= \ln\left\{\frac{1}{1-\sum_{i=1}^{N} p_{i-1}\, z^{-i}}\right\} \\ &= -\ln\left\{1-\sum_{i=1}^{N} p_{i-1}\, z^{-i}\right\}\end{aligned}$$

Representation of the Spectral Envelope Using Cepstral Coefficients – Part 7

Computationally efficient transformation from prediction to cepstral coefficients:

- Result so far

$$\sum_{i=-\infty}^{\infty} c_i \, z^{-i} \;=\; -\ln\left\{1 - \sum_{i=1}^{N} p_{i-1} \, z^{-i}\right\}$$

- Computation of the coefficients with non-negative indices

$$\ln\left\{1 - \sum_{i=1}^{N} p_{i-1} \, z^{-i}\right\} \;=\; \ln\left\{\prod_{i=0}^{N} \left(1 - b_i \, z^{-1}\right)\right\}$$

$$= \sum_{i=0}^{N} \ln\left\{1 - b_i \, z^{-1}\right\}$$

Insert

- Using the series

$$\ln\left\{1 - b \, z^{-1}\right\} \;=\; -\sum_{k=1}^{\infty} \frac{b^k}{k} \, z^{-k}, \qquad \text{für } |z| > |b|$$

Representation of the Spectral Envelope Using Cepstral Coefficients – Part 8

Computationally efficient transformation from prediction to cepstral coefficients:

- Computation of the coefficients with non-negative indices:
  - Result after inserting the series:

$$\ln\left\{1-\sum_{i=1}^{N} p_{i-1}\, z^{-i}\right\} \;=\; -\sum_{i=0}^{N}\sum_{k=1}^{\infty} \frac{b_i^k}{k}\, z^{-k}$$

  - This results in

$$\sum_{i=1}^{\infty} c_i\, z^{-i} \;=\; -\ln\left\{1-\sum_{i=1}^{N} p_{i-1}\, z^{-i}\right\}$$

All coefficients with non-negative indices are zero.

Representation of the Spectral Envelope Using Cepstral Coefficients – Part 9

Computationally efficient transformation from prediction to cepstral coefficients:

- Result so far

$$\sum_{i=1}^{\infty} c_i \, z^{-i} \;=\; -\ln\left\{1 - \sum_{i=1}^{N} p_{i-1} \, z^{-i}\right\}$$

- Take the derivative

$$\frac{d}{dz}\left[\sum_{i=1}^{\infty} c_i \, z^{-i}\right] \;=\; -\frac{d}{dz}\left[\ln\left\{1 - \sum_{i=1}^{N} p_{i-1} \, z^{-i}\right\}\right]$$

$$-\sum_{i=1}^{\infty} i \, c_i \, z^{-i-1} \;=\; -\sum_{i=1}^{N} i \, p_{i-1} \, z^{-i-1} \left[1 - \sum_{i=1}^{N} p_{i-1} \, z^{-i}\right]^{-1}$$

- Multiply both sides with […]

$$\sum_{i=1}^{\infty} i \, c_i \, z^{-i-1} \;-\; \sum_{k=1}^{\infty} \sum_{i=1}^{N} k \, c_k \, p_{i-1} \, z^{-k-i-1} \;=\; \sum_{i=1}^{N} i \, p_{i-1} \, z^{-i-1}$$

Representation of the Spectral Envelope Using Cepstral Coefficients – Part 10

Computationally efficient transformation from prediction to cepstral coefficients:

- Result so far

$$\sum_{i=1}^{\infty} i\, c_i\, z^{-i-1} \;-\; \sum_{k=1}^{\infty} \sum_{i=1}^{N} k\, c_k\, p_{i-1}\, z^{-k-i-1} \;=\; \sum_{i=1}^{N} i\, p_{i-1}\, z^{-i-1}$$

- Comparing the coefficients for $i \in \{1, \ldots, N\}$

$$i\, c_i \;-\; \sum_{k=1}^{i-1} k\, c_k\, p_{i-k-1} \;=\; i\, p_{i-1}$$

- Comparing the coefficients for $i > N$

$$i\, c_i \;-\; \sum_{k=1}^{i-1} k\, c_k\, p_{i-k-1} \;=\; 0$$

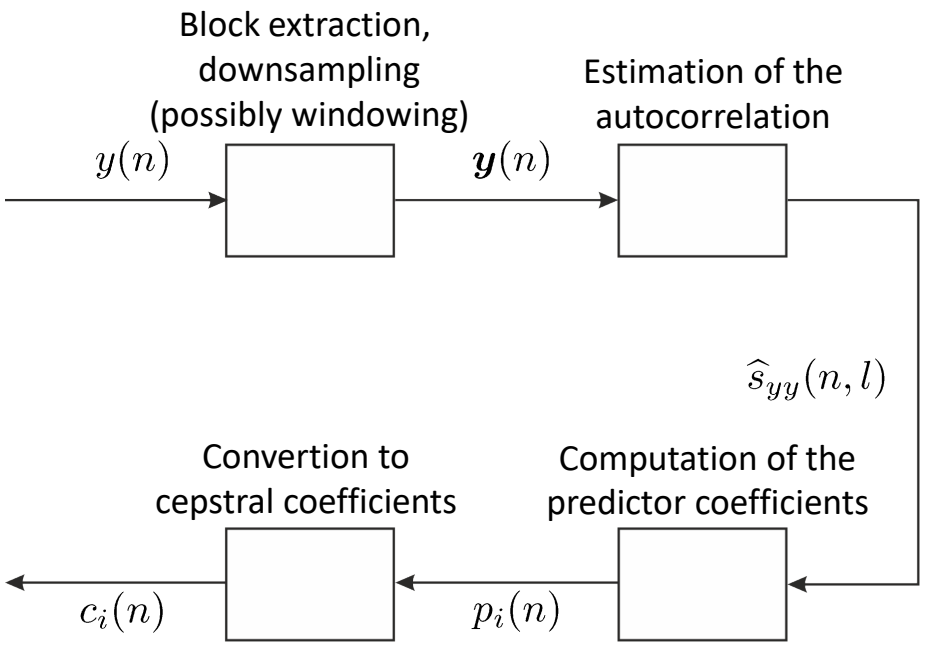
Representation of the Spectral Envelope Using Cepstral Coefficients – Part 11

Computationally efficient transformation from prediction to cepstral coefficients:

$$c_i = \begin{cases} 0, & \text{if } i < 1, \\ p_{i-1} + \frac{1}{i} \sum_{k=1}^{i-1} k\, c_k\, p_{i-k-1}, & \text{if } 1 \le i \le N, \\ \frac{1}{i} \sum_{k=1}^{i-1} k\, c_k\, p_{i-k-1}, & \text{else.} \end{cases}$$

Recursive computation with very low complexity. The summation can be stopped with low error after 3/2 N because cepstral coefficients with a higher index contribute only very little to the underlying cost function.

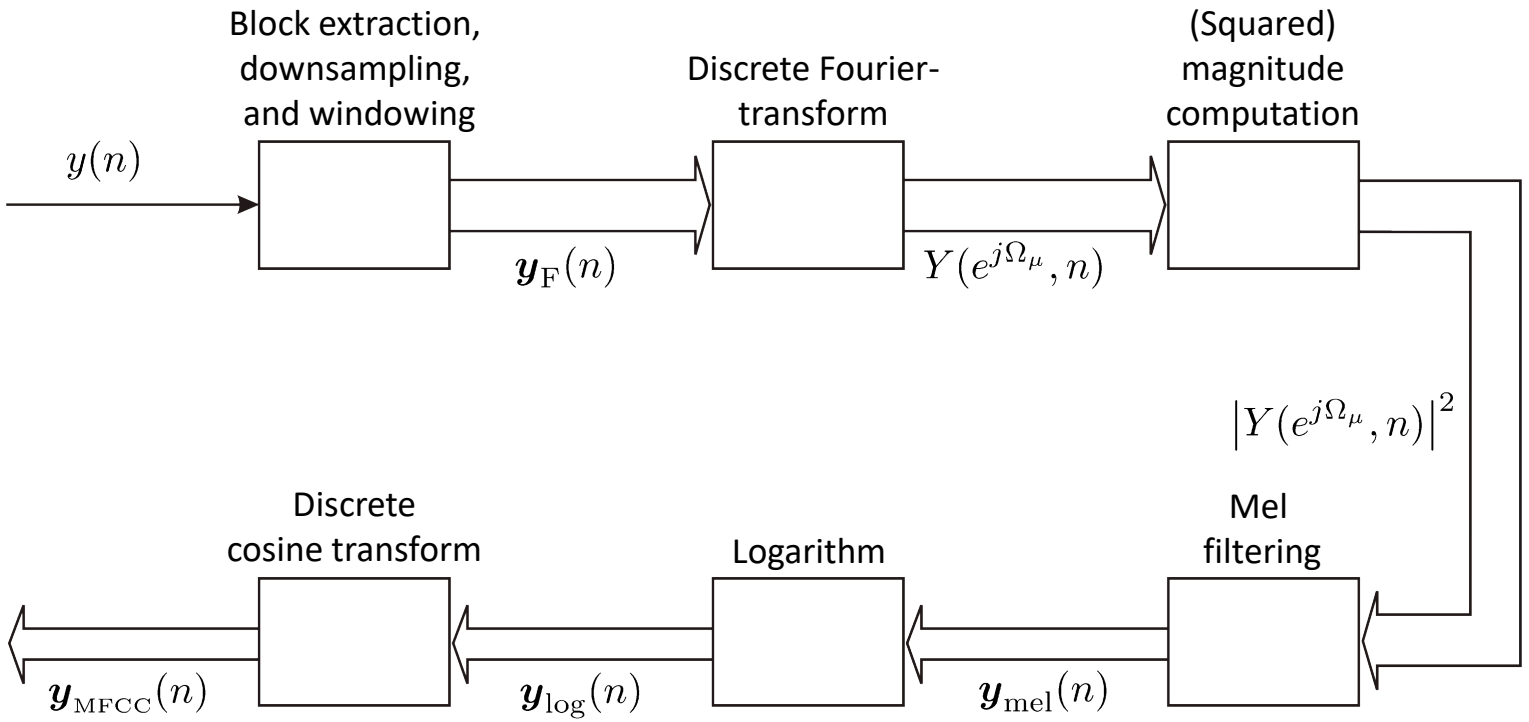
Representation of the Spectral Envelope Using Cepstral Coefficients – Part 12



- Typically, every 5 to 20 ms 15 to 30 cepstral coefficients are computed.
- Therefore, 10 to 20 predictor coefficients are computed.
- The autocorrelation values that are needed therefore are computed on an estimation basis of 20 to 50 ms of signal.
- This type of feature is commonly used when both spectral envelope and prediction error signal are used (coding, bandwidth extension, speech reconstruction).

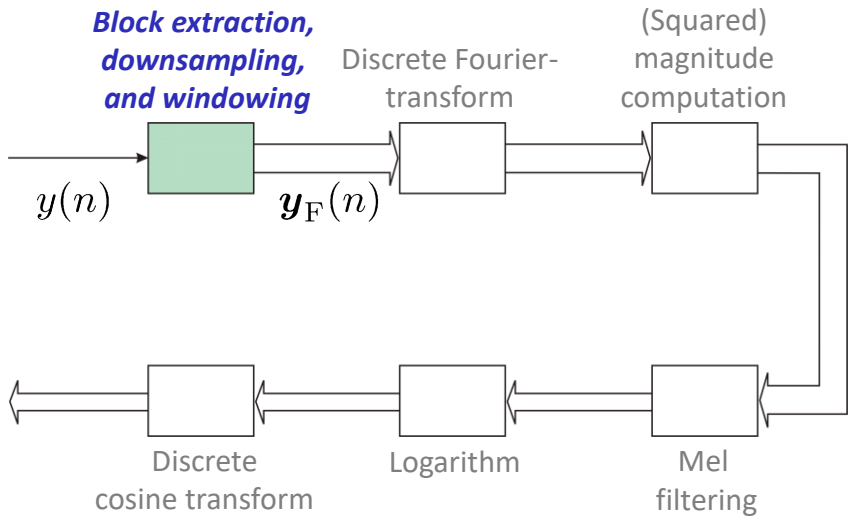
Mel-Filtered Cepstral Coefficients (MFCCs) – Part 1

Overview:



Mel-Filtered Cepstral Coefficients (MFCCs) – Part 2

Block extraction, downsampling, and windowing:



- Block extraction:

$$\tilde{\boldsymbol{y}}(n) = \left[y(n),\, y(n-1),\, \ldots y(n-N+1)\right]^{\mathrm{T}}$$

- Downsampling

$$\boldsymbol{y}(n) = \tilde{\boldsymbol{y}}(nr)$$

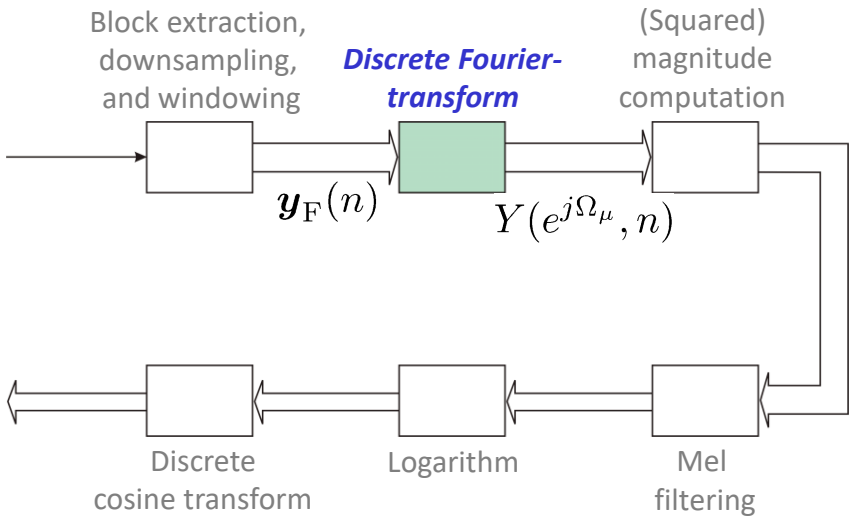
- Windowing:

$$\boldsymbol{y}_{\mathrm{F}}(n) = \boldsymbol{H}\,\boldsymbol{y}(n)$$

$$\boldsymbol{H} = \begin{bmatrix} h_0 & 0 & \ldots & 0 \\ 0 & h_1 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \ldots & 0 & h_{N-1} \end{bmatrix}$$

Mel-Filtered Cepstral Coefficients (MFCCs) – Part 3

Discrete Fourier-transform :



- Discrete Fourier transform:

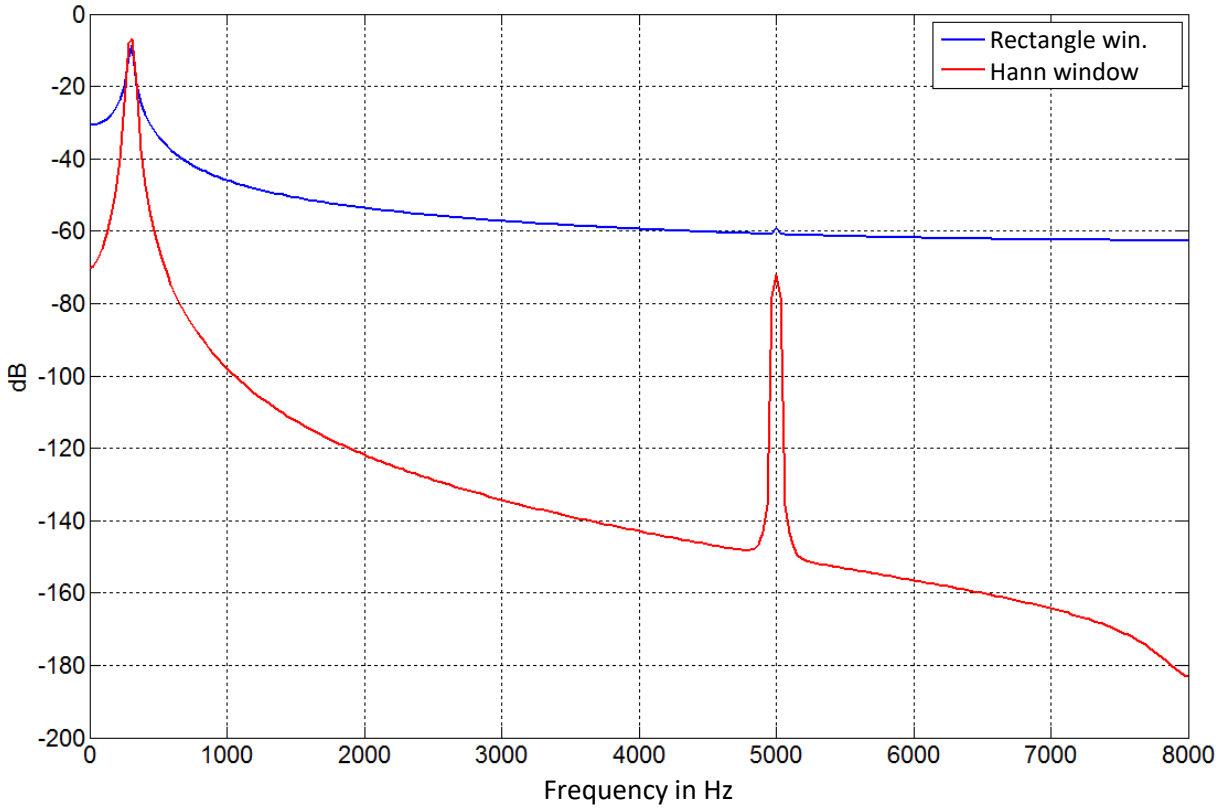
$$Y(e^{j\Omega_\mu}, n) = \sum_{k=0}^{N-1} y(nr-k)\, h_k\, e^{-j\frac{2\pi}{N}k\mu}$$

- In Matrix-vector notation:

$$\boldsymbol{y}(e^{j\Omega}, n) = \left[Y(e^{j\Omega_0}, n), \ldots, Y(e^{j\Omega_{N-1}}, n)\right]^{\mathrm{T}} = \boldsymbol{T}_N\, \boldsymbol{H}\, \boldsymbol{y}(n)$$

Mel-Filtered Cepstral Coefficients (MFCCs) – Part 4

Influence of the window function:

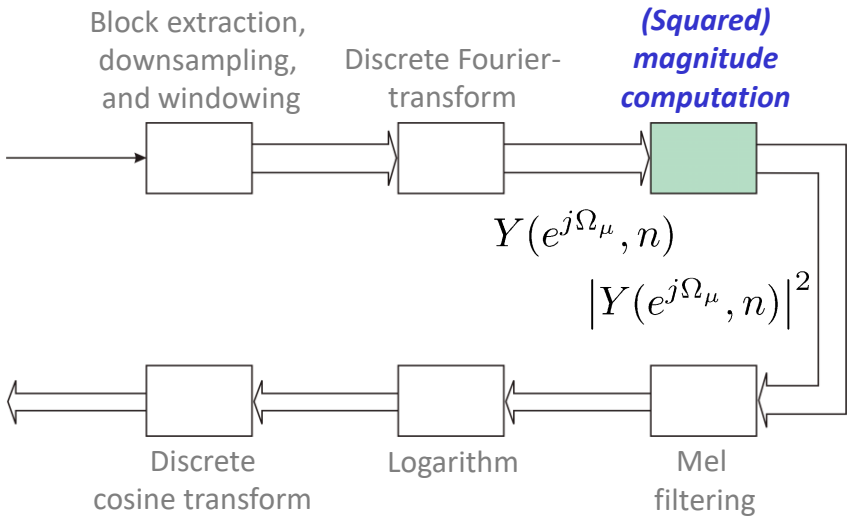


Input signal: two sinusoids with frequencies 300 Hz and 5000 Hz, amplitude ratio 66 dB

FFT-order and window length: 512

Mel-Filtered Cepstral Coefficients (MFCCs) – Part 5

(Squared) magnitude computation:



- Squared magnitude:

$$|Y(e^{j\Omega_\mu}, n)|^2 = \mathrm{Re}^2\{Y(e^{j\Omega_\mu}, n)\} + \mathrm{Im}^2\{Y(e^{j\Omega_\mu}, n)\}$$

- Approximation of the magnitude (reduced dynamic, reduced computational load):

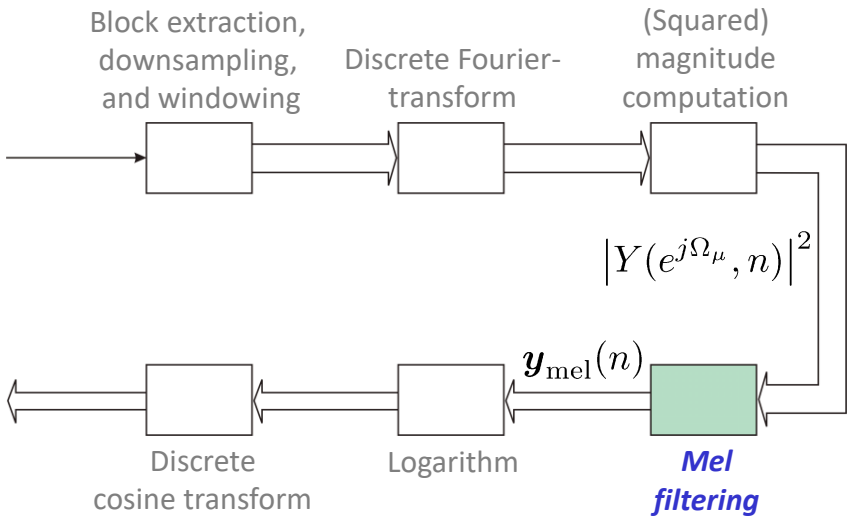
$$|Y(e^{j\Omega_\mu}, n)| \approx K\,|\mathrm{Re}\{Y(e^{j\Omega_\mu}, n)\}| + K\,|\mathrm{Im}\{Y(e^{j\Omega_\mu}, n)\}|$$

- In matrix-vector-notation:

$$\boldsymbol{y}_{\mathrm{abs}}(n) = \left[|Y(e^{j\Omega_0}, n)|, \ldots, |Y(e^{j\Omega_{N-1}}, n)|\right]^{\mathrm{T}}$$

Mel-Filtered Cepstral Coefficients (MFCCs) – Part 6

Mel filtering – part 1:



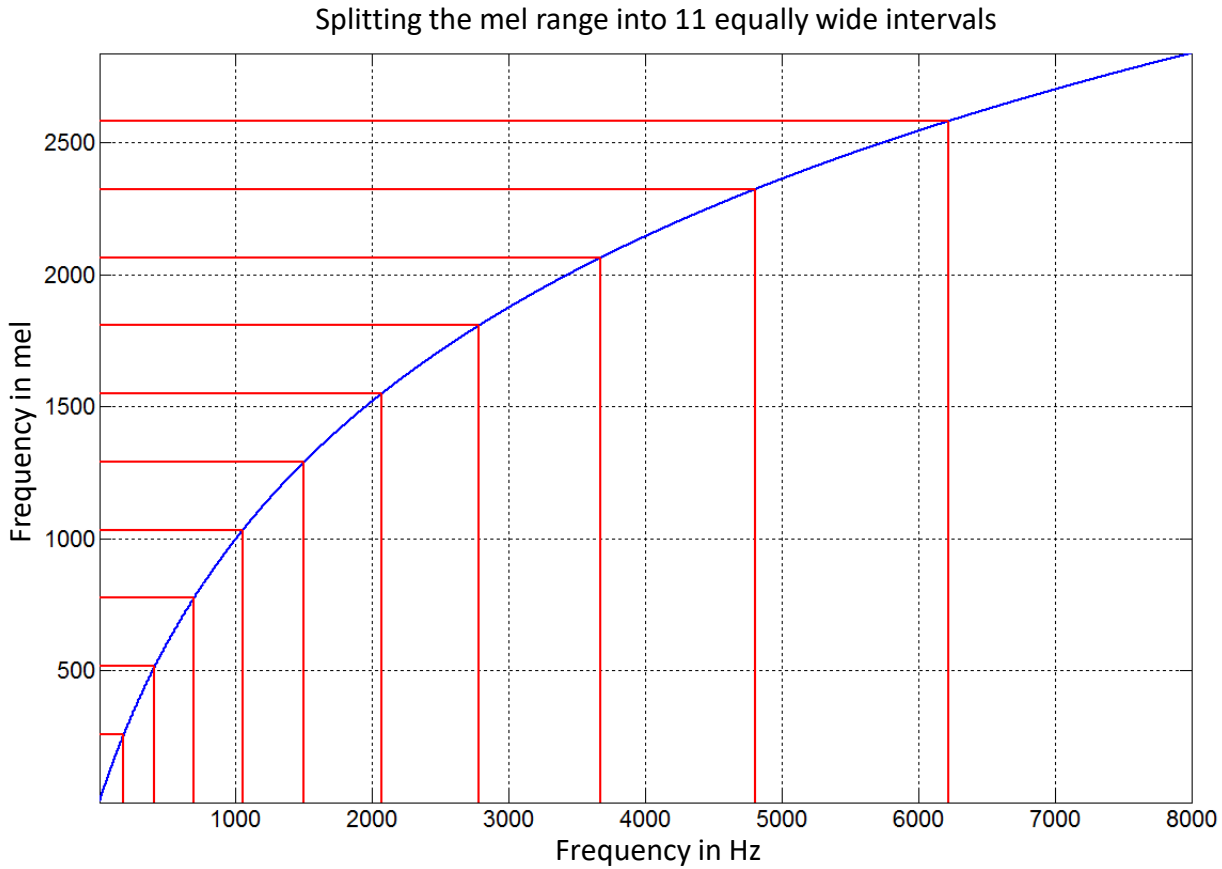
- Mel-frequency relation:

$$m = 2595\,\mathrm{Mel}\,\log_{10}\left\{\frac{f}{700\,\mathrm{Hz}} + 1\right\}$$

- Linear splitting of the mel domain into N intervals of the same width
- Overlapping of the intervals by 50 % percent with the left and right neighbor
- Usually, triangular-shaped windows (in the linear frequency domain) are used
- The triangular filters are usually normalized such that the produce the same output power when they are excited with white noise.

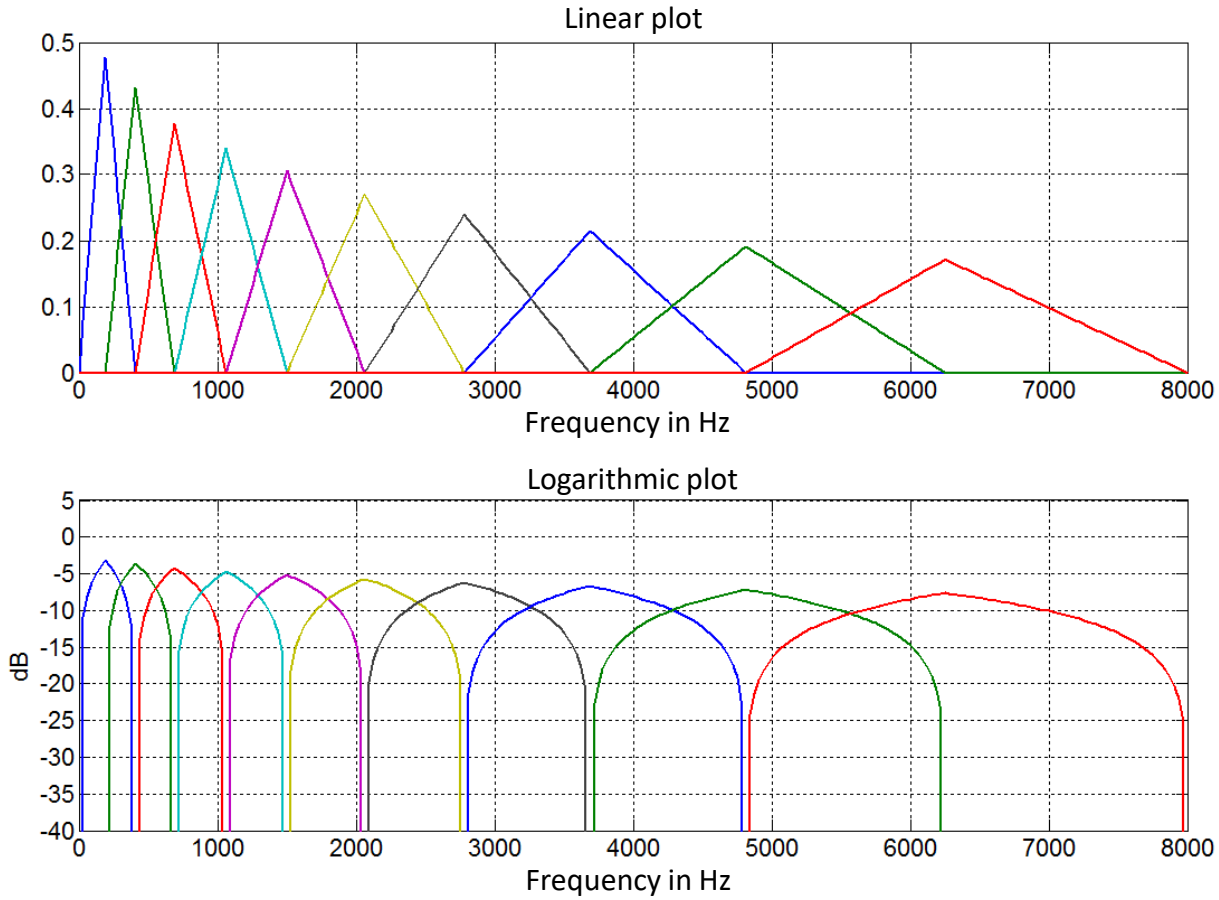
Mel-Filtered Cepstral Coefficients (MFCCs) – Part 7

Mel filtering – part 2:



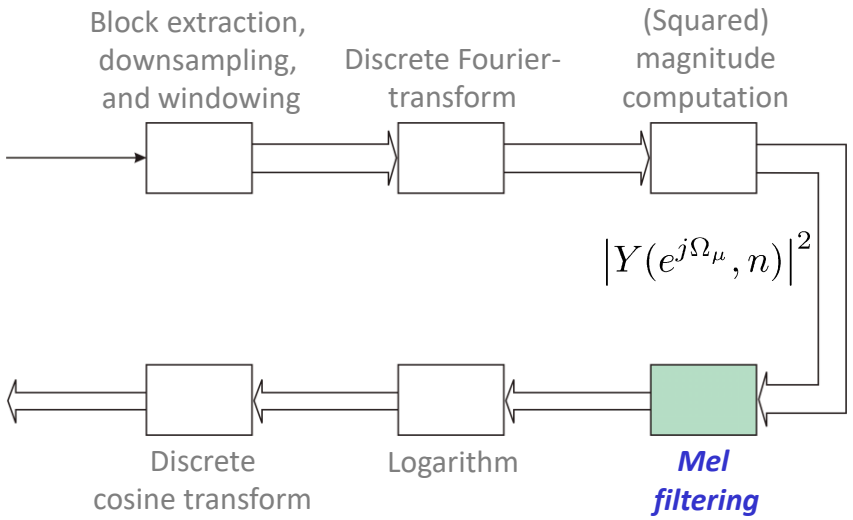
Mel-Filtered Cepstral Coefficients (MFCCs) – Part 8

Mel filtering – part 3:



Mel-Filtered Cepstral Coefficients (MFCCs) – Part 9

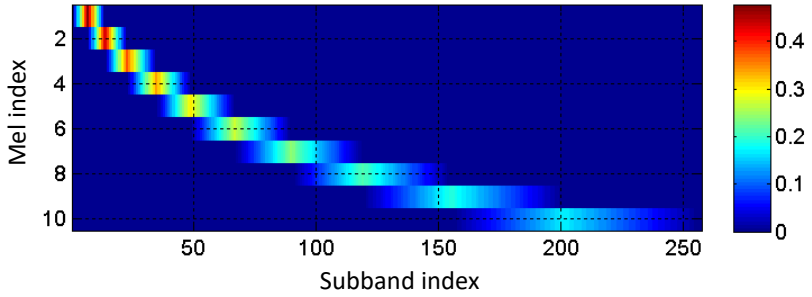
Mel filtering – part 4:



- Typically, 15 to 30 mel filters are used for sample rates between 8 and 16 kHz
- Matrix-vector notation:

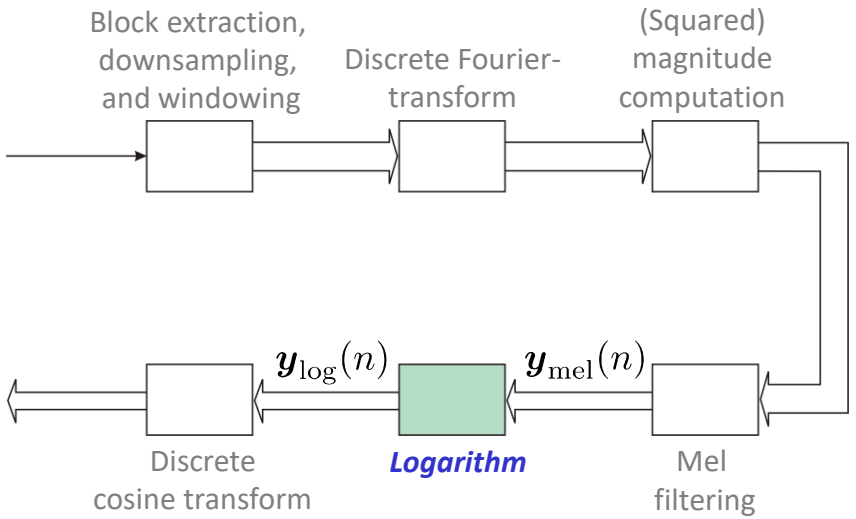
$$\boldsymbol{y}_{\text{mel}}(n) \quad = \quad \boldsymbol{M}\, \boldsymbol{y}_{\text{abs}}(n)$$

- The filter matrix ***M***:



Mel-Filtered Cepstral Coefficients (MFCCs) – Part 10

Logarithm – part 1:



- Logarithm:

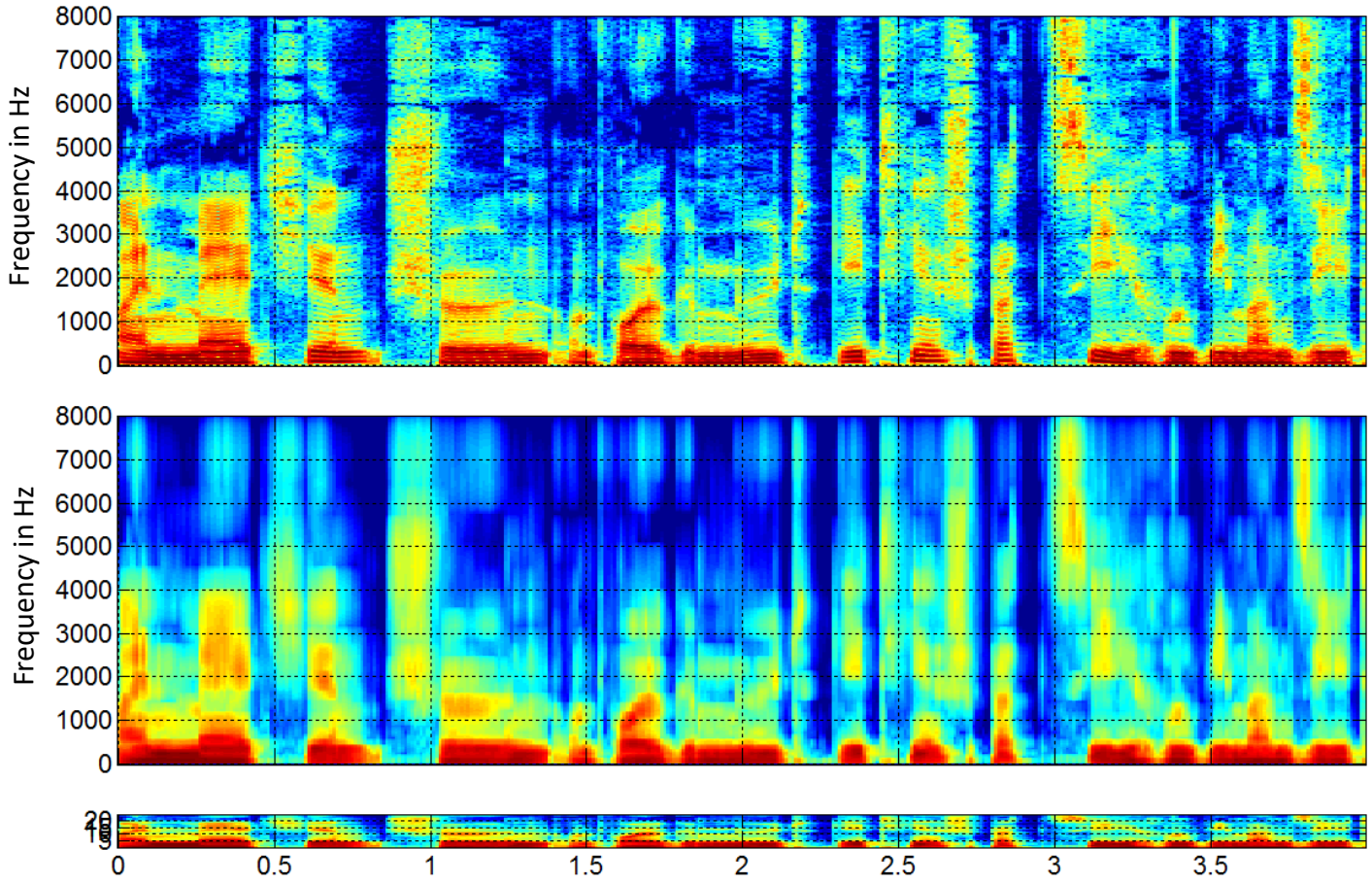
$$\boldsymbol{y}_{\log}(n) \quad = \quad \log_e \left\{ \boldsymbol{y}_{\mathrm{mel}}(n) \right\}$$

- Alternatively, another base can be used for the logarithm.
- Similar to the mel filter bank, also the logarithm is motivated by the human hearing. It is a simple approximation of the loudness.

Mel-Filtered Cepstral Coefficients (MFCCs) – Part 11

Logarithm – part 2:

The size of the picture respresents the amount of data!



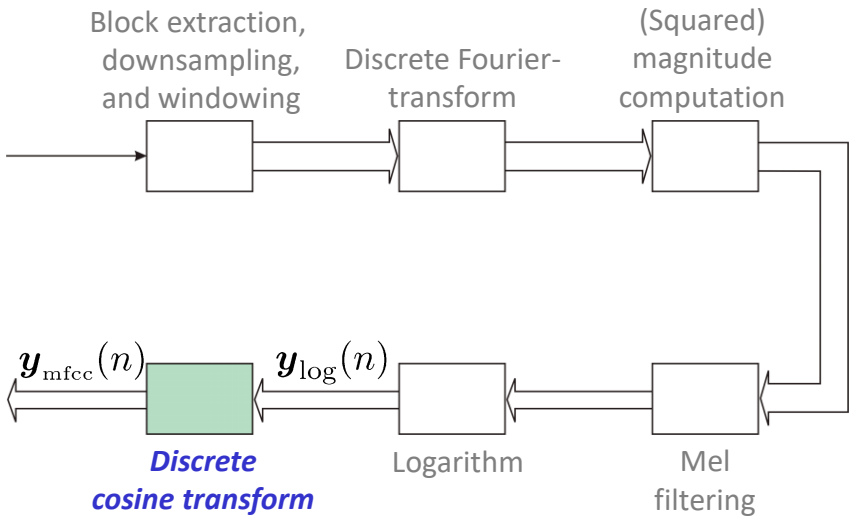
Representation of
$\log_e\{\boldsymbol{y}_{\text{abs}}(n)\}$
over the time

Representation of
$\log_e\{\boldsymbol{M}^{\text{T}}\,\boldsymbol{y}_{\text{mel}}(n)\}$
over the time

Representation of
$\boldsymbol{y}_{\log}(n)$
over the time

Mel-Filtered Cepstral Coefficients (MFCCs) – Part 12

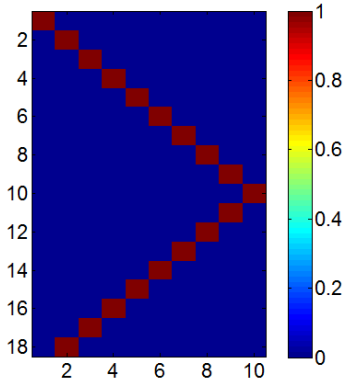
Discrete cosine transform – part 1:



- Symmetric extension of the logarithmic mel regions:

$$\tilde{\boldsymbol{y}}_{\log}(n) \quad = \quad \boldsymbol{E}\, \boldsymbol{y}_{\log}(n)$$

- Extension matrix ***E***:

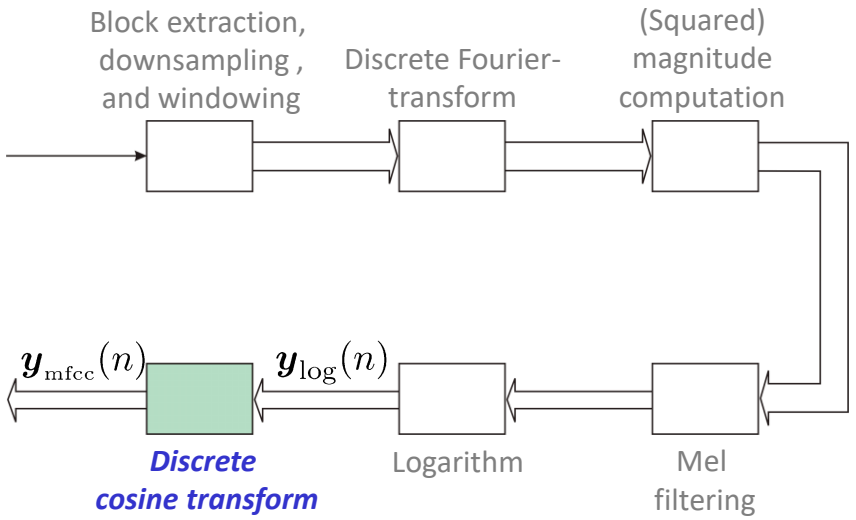


- Transform into the „time-domain“:

$$\tilde{\boldsymbol{y}}_{\mathrm{mfcc}}(n) \quad = \quad \boldsymbol{T}_M^{-1}\, \tilde{\boldsymbol{y}}_{\log}(n)$$

Mel-Filtered Cepstral Coefficients (MFCCs) – Part 13

Discrete cosine transform – part 2:



- Because the input vectors are real-valued, the IDFT can be transformed into (a variant) of the IDCT.
- Shortening of the inversely transformed vector:

$$\boldsymbol{y}_{\mathrm{mfcc}}(n) \quad = \quad \boldsymbol{P}\,\boldsymbol{T}_M^{-1}\,\boldsymbol{E}\,\boldsymbol{y}_{\log}(n)$$

- The transformation causes a „decorrelation“ of the logarithmic features. It is an approximation of a principal component analysis.
- The shortening should reduce the influence of the fundamental speech frequency, i.e. coefficients for the high frequencies are omitted. Typically, the last third of the vector is removed.

Mel-Filtered Cepstral Coefficients (MFCCs) – Part 14

Discrete cosine transform – part 3:

- For analysis of the decorrelation property of the inverse DCT, the feature vectors are first normalized by their variance after the mean has been removed.

$$\begin{aligned} \boldsymbol{y}_{\log,\mathrm{nor}}(n) &= \boldsymbol{N}_{\log}\left[\boldsymbol{y}_{\log}(n) - \boldsymbol{\mu}_{\log}\right], \\ \boldsymbol{y}_{\mathrm{mfcc},\mathrm{nor}}(n) &= \boldsymbol{N}_{\mathrm{mfcc}}\left[\boldsymbol{y}_{\mathrm{mfcc}}(n) - \boldsymbol{\mu}_{\mathrm{mfcc}}\right]. \end{aligned}$$

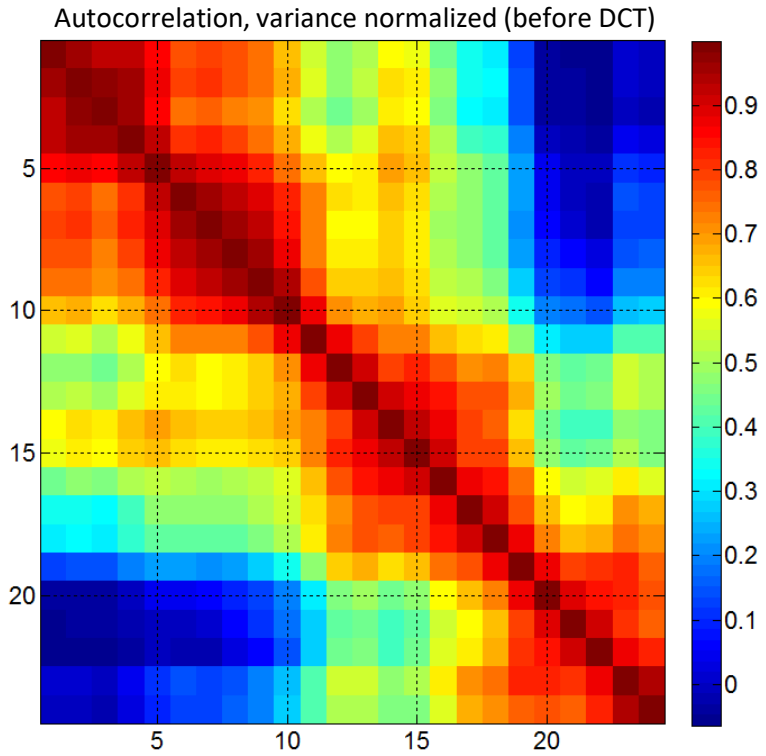
  The normalization matrices contain the inverse standard deviations on their main diagonals.

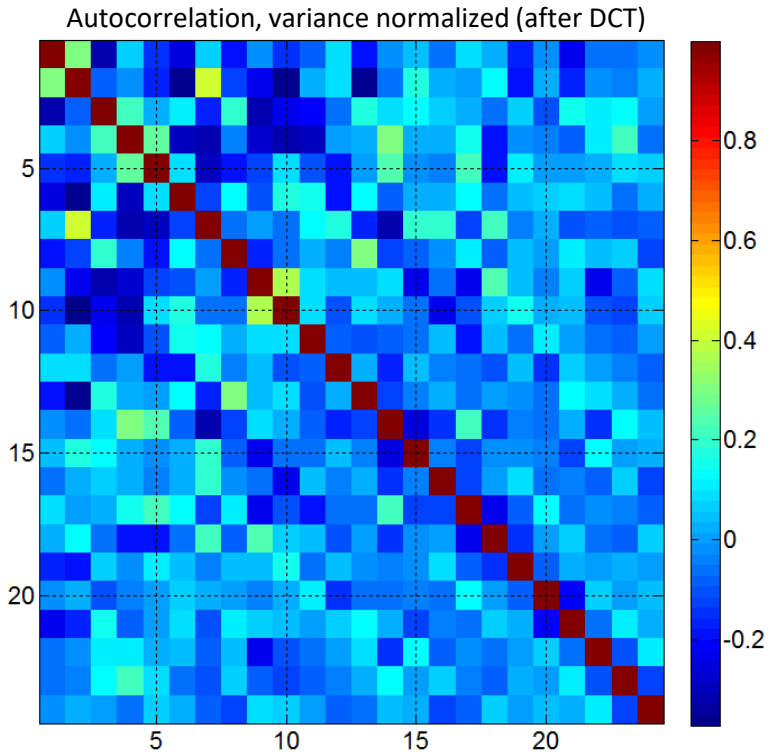
- Afterwards, the autocorrelation matrix of both types of feature vectors are estimated:

$$\begin{aligned} \boldsymbol{S}_{\log} &= \mathrm{E}\left\{\boldsymbol{y}_{\log,\mathrm{nor}}(n),\, \boldsymbol{y}^{\mathrm{T}}_{\log,\mathrm{nor}}(n)\right\}, \\ \boldsymbol{S}_{\mathrm{mfcc}} &= \mathrm{E}\left\{\boldsymbol{y}_{\mathrm{mfcc},\mathrm{nor}}(n),\, \boldsymbol{y}^{\mathrm{T}}_{\mathrm{mfcc},\mathrm{nor}}(n)\right\}. \end{aligned}$$

Mel-Filtered Cepstral Coefficients (MFCCs) – Part 15

Discrete cosine transform – part 4:

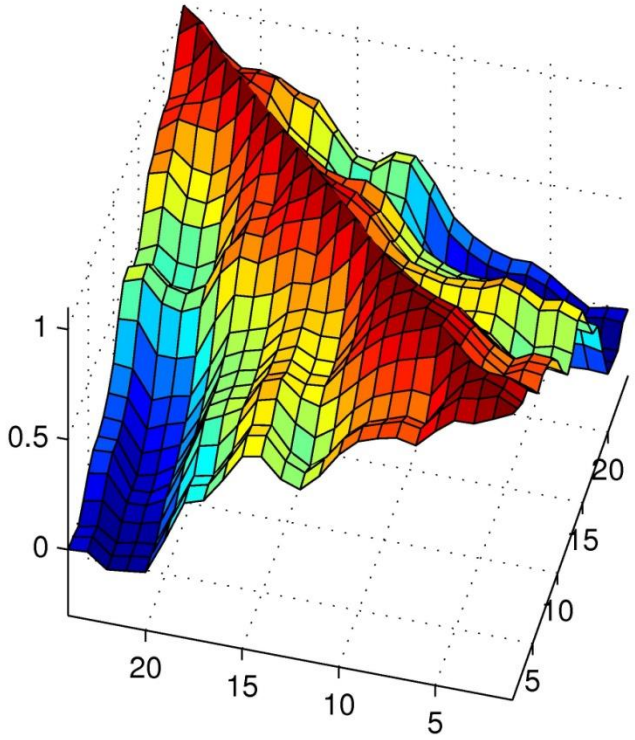




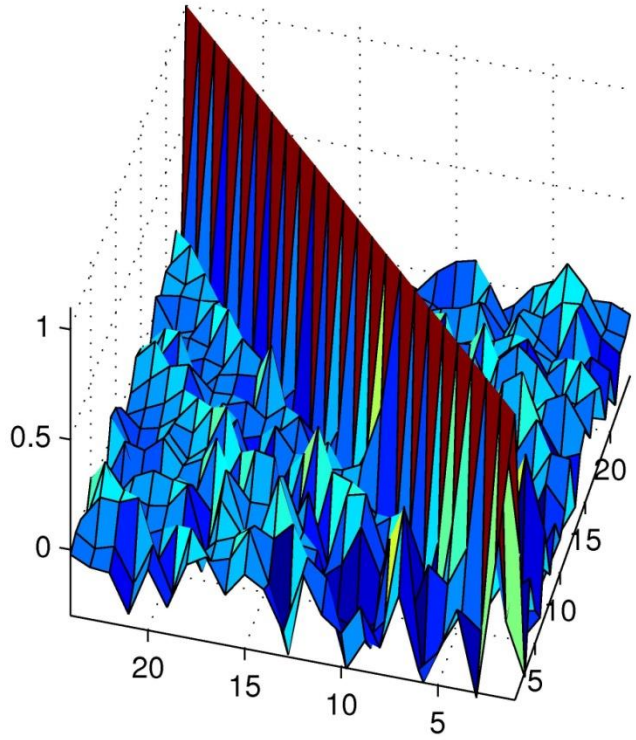
Mel-Filtered Cepstral Coefficients (MFCCs) – Part 16

Discrete cosine transform – part 4:

Autocorrelation, variance normalized (before DCT)



Autocorrelation, variance normalized (after DCT)



Postprocessing

Outlook:

- Often, ***several subsequent features are combined*** after the feature extraction. In some cases, the difference of to subsequent vectors is formed (so-called ***delta features***) or even the difference of two subsequent differences (so-called ***delta-delta features***).
- As an alternative, so-called ***super vectors*** can be formed by appending some subsequent feature vectors. Because the feature dimensionality is increased by doing so, so-called LDA matrices may be applied (LDA = ***linear discriminant analysis***). The goal is to reduce the variance of features that belong to one class, while maximizing the distance between classes. This allows to reduce the ***dimensionality of the feature space*** without loosing too much of the accuracy of the model.

Summary and Outlook

Summary:

- Introduction
- Features for speech and speaker recognition
  - Pitch frequency
  - Spectral envelope
- Representations for the spectral envelope
  - Coefficients of a prediction filter
  - Cepstral coefficients
  - Mel-filtered/frequency cepstral coefficients (MFCCs)

Next part:

- Training of codebooks