Part II.
Noise Reduction – Beamforming

Reinhold Haeb-Umbach

Speech capture in noisy environments

- Forming a beam of increased sensitivity towards the desired speaker reduces noise and other distortions

Table of contents in part II

- Some physics
- From physics to signal processing
- Optimal beamforming design criteria
- Speech presence probability (mask) estimation
  - Spatial mixture models
  - Neural networks
- Speaker-conditioned spectrogram masking

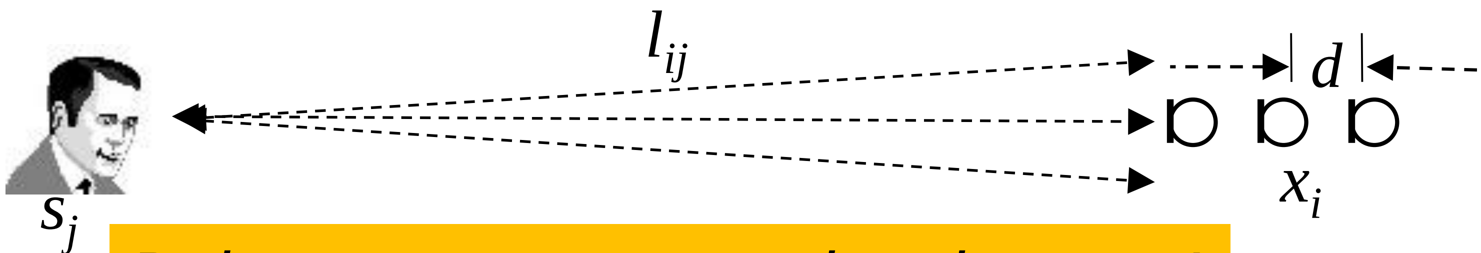
Some physics

- In free space, waveform at point i caused by a waveform emitted at point j

$$x_i[\tilde{t}] = \frac{1}{\sqrt{4\pi l_{ij}}} s_j \left[\tilde{t} - \frac{l_{ij}}{c} \right]$$

  where l_{ij} is distance from position i to j

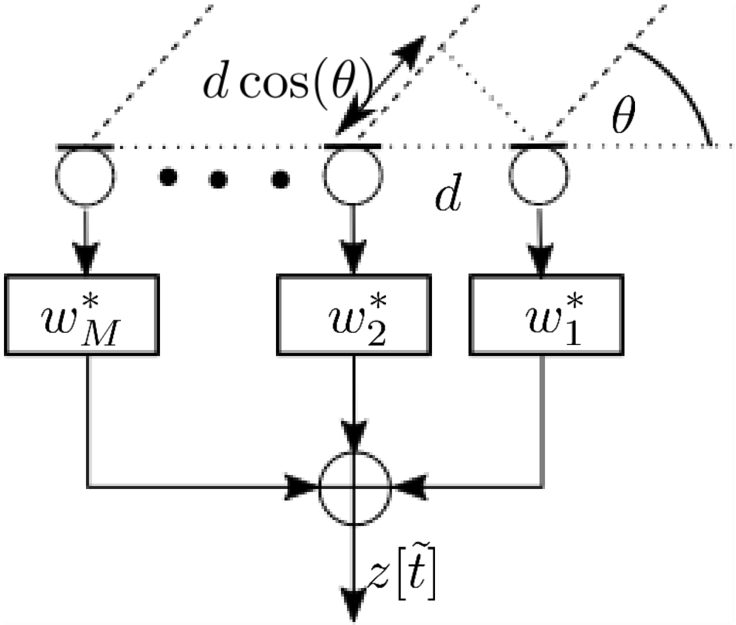
- Far-field: l_{ij} much larger than inter-microphone distance d
  - Plane wave
  - Attenuation factor $1/\sqrt{4\pi l_{ij}}$ the same for all mics
  - Signal delay between microphones $\tilde{\tau} = d/c$ where $c \approx 340\,\mathrm{m/s}$
    - Example: for $d = 10\,\mathrm{cm} \Rightarrow \tilde{\tau} = 0.3\,\mathrm{ms} = 4.7$ samples @ 16 kHz



Delay matters, attenuation does not!

Basics of acoustic beamforming

$$s[\tilde{t}] = \mathrm{e}^{j\omega_0 \tilde{t}} = \mathrm{e}^{j\frac{2\pi c}{\lambda_0}\tilde{t}}$$



Signal at *m*th microphone:

$$x_m[\tilde{t}] = s[\tilde{t} - \tilde{\tau}_m] = \mathrm{e}^{j\omega_0(\tilde{t}-\tilde{\tau}_m)}$$

$$\tilde{\tau}_m = \frac{(m-1)d\cos\theta}{c}; \; m = 1, \ldots, M$$

Beamformer output:

$$z[\tilde{t}] = \sum_{m=1}^{M} w_m^* x_m[\tilde{t}]$$

$$= \ldots$$

$$= \mathrm{e}^{j\omega_0\tilde{t}} \mathbf{w}^{\mathrm{H}} \mathbf{v}(\theta, \lambda_0)$$

Beamformer coeff.: $\mathbf{w} = [w_1, \ldots, w_M]^\top$

Steering vector: $\mathbf{v}(\theta, \lambda_0) = \begin{pmatrix} 1 & \mathrm{e}^{-j2\pi\left(\frac{d}{\lambda_0}\right)\cos(\theta)} & \ldots & \mathrm{e}^{-j2\pi\left(\frac{d}{\lambda_0}\right)\cos(\theta)(M-1)} \end{pmatrix}$

Delay-Sum Beamformer (DSB)

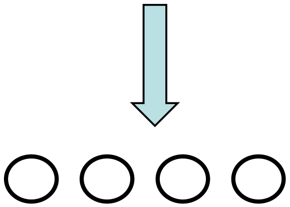
- Delay-Sum Beamformer: $\mathbf{w} = \frac{1}{M}\begin{pmatrix} 1 & \mathrm{e}^{-j\phi_0} & \cdots & \mathrm{e}^{-j(M-1)\phi_0} \end{pmatrix}^\mathsf{T}$

  with phase term $\phi_0 = \omega_0 \tau_0 = \omega_0 \frac{d \cos\theta_0}{c} = 2\pi \frac{d}{\lambda_0} \cos(\theta_0)$

  – DSB steered towards geometric angle θ_0

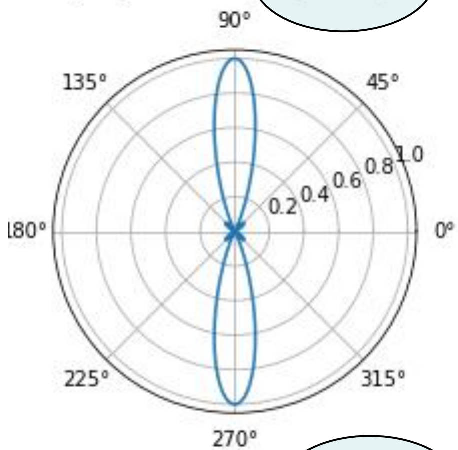
- Beampattern: $$\begin{aligned} \left|z[\tilde{t}]\right| &= \left|\mathrm{e}^{j\omega_0 \tilde{t}} \cdot \mathbf{w}^{\mathrm{H}} \mathbf{v}\right| \\ &= \cdots \\ &= \frac{1}{M} \left| \frac{\sin\left(\frac{M}{2} 2\pi \frac{d}{\lambda_0} (\cos(\theta) - \cos(\theta_0))\right)}{\sin\left(\frac{1}{2} 2\pi \frac{d}{\lambda_0} (\cos(\theta) - \cos(\theta_0))\right)} \right| \end{aligned}$$

Example beampatterns

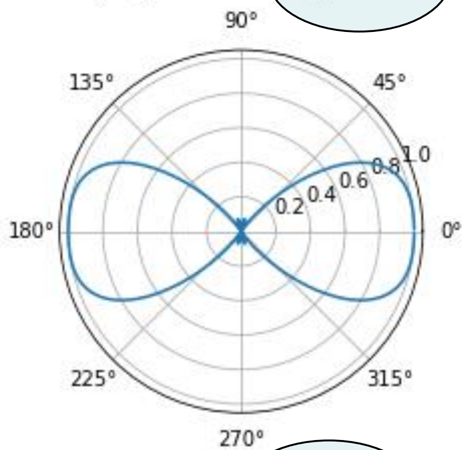


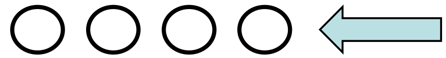
Broadside
(here: top/bottom)

$d/\lambda_0 = 0.5; \theta_0 = \pi/2$



$d/\lambda_0 = 0.5; \theta_0 = 0$

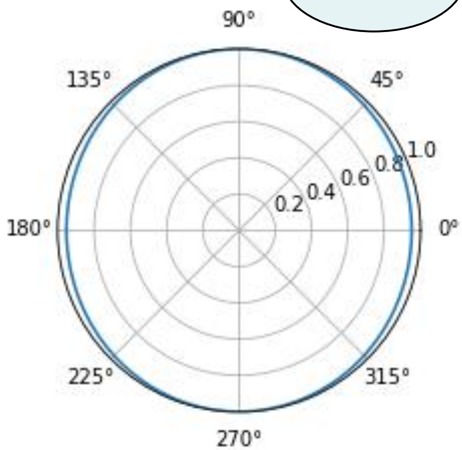




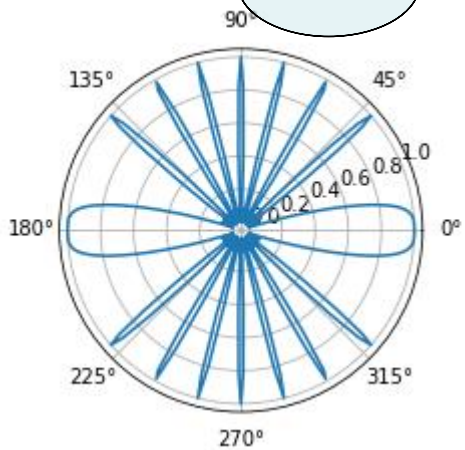
Endfire
(here: left/right)

small
inter-element
distance /
low frequency

$d/\lambda_0 = 1./32; \theta_0 = \pi/2$



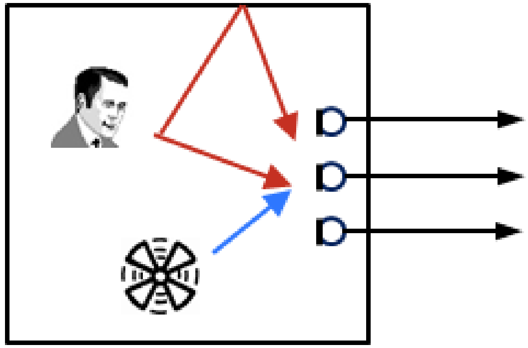
$d/\lambda_0 = 4; \theta_0 = \pi/2$



large
inter-element
Distance /
high frequency

From physics to signal processing

Real acoustic environments:



- Reverberation
  - Time differences of arrival (TDOAs) inappropriate
- Wideband beamforming
  - Fourier transform domain processing
- Interferences
  - Need appropriate objective functions
- Unknown and time-varying acoustic environment
  - Estimation of beamformer coefficients

Most common model

- Signal at *m*-th microphone:

$$x_m[\tilde{t}] = s[\tilde{t} - \tilde{\tau}_m] \;\; \rightarrow \;\; y_m[\tilde{t}] = x_m[\tilde{t}] + n[\tilde{t}] = \sum_{\tilde{\tau}=0}^{\tilde{L}-1} a_m[\tilde{\tau}] s[\tilde{t} - \tilde{\tau}] + n[\tilde{t}]$$

- Short-Time Fourier Transform (STFT): $y_m[\tilde{t}] \;\rightarrow\; y_{m,t,f}$

- Narrowband assumption (multiplicative transfer function approx.): length of acoustic impulse response << STFT analysis window
  - convolution in time domain corresponds to multiplication in STFT domain
- Time-invariant Acoustic Transfer Function (ATF)

$$y_{m,t,f} = a_{m,f} s_{t,f} + n_{t,f}; \quad m = 1, \ldots, M$$

$$\mathbf{y}_{t,f} = \mathbf{a}_f s_{t,f} + \mathbf{n}_{t,f} := \mathbf{x}_{t,f} + \mathbf{n}_{t,f}$$

ATF vs RTF

- Scale ambiguity of ATF

$$\mathbf{x}_{t,f} = \mathbf{a}_f s_{t,f} = (\mathbf{a}_f \cdot C) \cdot s_{t,f} / C; \quad C \in \mathbb{C}$$

- Fix ambiguity: Relative transfer function (RTF)

$$\tilde{\mathbf{a}}_f = \frac{\mathbf{a}_f}{a_{1,f}} = \left(1, \frac{a_{2,f}}{a_{1,f}}, \ldots, \frac{a_{M,f}}{a_{1,f}}\right)^{\mathsf{T}}$$

$$\Rightarrow \mathbf{x}_{t,f} = \mathbf{a}_f s_{t,f} = \tilde{\mathbf{a}}_f a_{1,f} s_{t,f} = \tilde{\mathbf{a}}_f x_{1,t,f}$$

- Thus our goal is to estimate the <u>*image*</u> of the source at a reference microphone (e.g., mic. #1)

$$x_{1,t,f} = a_{1,f} s_{t,f}$$

  – Thus, we do not attempt to dereverberate the signal!

Optimal beamforming design criteria: MMSE

- Beamformer output: $z_{t,f} = \mathbf{w}_f^{\mathsf{H}} \mathbf{y}_{t,f}$
- MMSE:

$$\min_{\mathbf{w}_f} \mathbb{E}\left[\left|\mathbf{w}_f^{\mathsf{H}} \mathbf{y}_{t,f} - x_{1,t,f}\right|^2\right] = \min_{\mathbf{w}_f} \mathbb{E}\left[\left|\mathbf{w}_f^{\mathsf{H}} \mathbf{x}_{t,f} - x_{1,t,f}\right|^2\right] + \mathbb{E}\left[\left|\mathbf{w}_f^{\mathsf{H}} \mathbf{n}_{t,f}\right|^2\right]$$

Add weight μ

Results in:

$$\mathbf{w}_f^{\text{SDW-MWF}} = \left(\mathbf{\Psi}_{\mathbf{xx},f} + \mu \mathbf{\Psi}_{\mathbf{nn},f}\right)^{-1} \mathbf{\Psi}_{\mathbf{xx},f} \mathbf{u}_1$$

where $\mathbf{\Psi}_{\mathbf{xx},f} = \mathbb{E}\left[\mathbf{x}_{t,f}\mathbf{x}_{t,f}^{\mathsf{H}}\right]$ (spatial covar. matrix of speech)

$\mathbf{\Psi}_{\mathbf{nn},f} = \mathbb{E}\left[\mathbf{n}_{t,f}\mathbf{n}_{t,f}^{\mathsf{H}}\right]$ (spatial covar. matrix of noise)

$\mathbf{u}_1 = [1, 0, \ldots, 0]^{\top}$ (points to reference microphone)

Speech Distortion Weighted Multi-channel Wiener Filter (SDW-MWF)

Optimal beamforming design criteria: M(P|V)DR

- MPDR: Minimum Power Distortionless Response:

$$\min_{\mathbf{w}_f} \mathbb{E}\left[\left|\mathbf{w}_f^{\mathsf{H}} \boldsymbol{\Psi}_{\mathbf{yy},f} \mathbf{w}_f\right|^2\right] \text{ subject to } \mathbf{w}_f^{\mathsf{H}} \tilde{\mathbf{a}}_f = 1$$

gives $$\mathbf{w}_f^{\mathrm{MPDR}} = \frac{\boldsymbol{\Psi}_{\mathbf{yy},f}^{-1} \tilde{\mathbf{a}}_f}{\tilde{\mathbf{a}}_f^{\mathsf{H}} \boldsymbol{\Psi}_{\mathbf{yy},f}^{-1} \tilde{\mathbf{a}}_f}$$

- MVDR: Minimum Variance Distortionless Response:

$$\min_{\mathbf{w}_f} \mathbb{E}\left[\left|\mathbf{w}_f^{\mathsf{H}} \boldsymbol{\Psi}_{\mathbf{nn},f} \mathbf{w}_f\right|^2\right] \text{ subject to } \mathbf{w}_f^{\mathsf{H}} \tilde{\mathbf{a}}_f = 1$$

gives $$\mathbf{w}_f^{\mathrm{MVDR}} = \frac{\boldsymbol{\Psi}_{\mathbf{nn},f}^{-1} \tilde{\mathbf{a}}_f}{\tilde{\mathbf{a}}_f^{\mathsf{H}} \boldsymbol{\Psi}_{\mathbf{nn},f}^{-1} \tilde{\mathbf{a}}_f}$$

Optimal beamforming design criteria: maxSNR

- Maximize output SNR:

$$\max_{\mathbf{w}_f} \frac{\mathbf{w}_f^{\mathsf{H}} \mathbf{\Psi}_{\mathbf{xx},f} \mathbf{w}_f}{\mathbf{w}_f^{\mathsf{H}} \mathbf{\Psi}_{\mathbf{nn},f} \mathbf{w}_f}$$

leads to generalized eigenvalue problem. $\mathbf{\Psi}_{\mathbf{xx},f} \mathbf{w}_f = \lambda \mathbf{\Psi}_{\mathbf{nn},f} \mathbf{w}_f$
which can be transformed to ordinary eigenvalue problem by Cholesky factorization: $\mathbf{\Psi}_{\mathbf{nn},f} = \mathbf{L}_f \mathbf{L}_f^{\mathsf{H}}$

$$\left(\mathbf{L}_f^{-1} \mathbf{\Psi}_{\mathbf{xx},f} \mathbf{L}_f^{-H}\right) \left(\mathbf{L}_f^{H} \mathbf{w}_f\right) = \lambda \left(\mathbf{L}_f^{H} \mathbf{w}_f\right)$$

Solution:

$$\mathbf{w}_f^{\mathrm{maxSNR}} = \mathbf{L}_f^{-H} \mathcal{P}\left(\mathbf{L}_f^{-1} \mathbf{\Psi}_{\mathbf{xx},f} \mathbf{L}_f^{-H}\right)$$

(Notation: $\mathcal{P}(\mathbf{A})$: Eigenvector corresponding to largest Eigenvalue of $\mathbf{A}$)

Rank-1 Constraint

Narrowband (rank-1) assumption: $\mathbf{x}_{t,f} = \tilde{\mathbf{a}}_f x_{1,t,f} \Rightarrow \mathbf{\Psi}_{\mathbf{xx},f} = \tilde{\mathbf{a}}_f \tilde{\mathbf{a}}_f^{\mathsf{H}} \sigma_{x_1,f}^2$

Use in SDW-MWF: gives[1]: $\mathbf{w}_f^{\text{r1-SDW-MWF}} = \dfrac{\mathbf{\Psi}_{\mathbf{nn},f}^{-1} \tilde{\mathbf{a}}_f \tilde{\mathbf{a}}_f^{\mathsf{H}} \sigma_{x_1,f}^2}{\mu + \operatorname{tr}\left\{ \mathbf{\Psi}_{\mathbf{nn},f}^{-1} \tilde{\mathbf{a}}_f \tilde{\mathbf{a}}_f^{\mathsf{H}} \sigma_{x_1,f}^2 \right\}} \mathbf{u}_1$

With μ=0 we obtain $\mathbf{w}_f^{\text{r1-SDW-MWF-0}} = \dfrac{\mathbf{\Psi}_{\mathbf{nn},f}^{-1} \tilde{\mathbf{a}}_f}{\tilde{\mathbf{a}}_f^{\mathsf{H}} \mathbf{\Psi}_{\mathbf{nn},f}^{-1} \tilde{\mathbf{a}}_f} = \mathbf{w}^{\text{MVDR}}$

Enforcing rank-1 constraint on maxSNR beamformer gives

$$\begin{aligned}
\mathbf{w}_f^{\text{maxSNR}} &= \mathbf{L}_f^{-H} \mathcal{P}\left(\mathbf{L}_f^{-1} \tilde{\mathbf{a}}_f \tilde{\mathbf{a}}_f^{\mathsf{H}} \sigma_{x_1,f}^2 \mathbf{L}_f^{-H} \right) = \mathbf{L}_f^{-H} \mathbf{L}_f^{-1} \tilde{\mathbf{a}}_f \\
&= \mathbf{\Psi}_{\mathbf{nn},f}^{-1} \tilde{\mathbf{a}}_f
\end{aligned}$$

All beamformers point in same direction and differ only in complex (freq.dep.) constant

[1] employ matrix inversion lemma

Beamforming Criteria: Discussion

- maxSNR beamformer introduces speech distortions, while MVDR does not
  - Can be compensated by postfilter [Warsitz and Haeb-Umbach, 2007]

- There is no unanimous opinion which of the beamformers performs best for enhancement for ASR
  - Advice: try out all of them

- A good estimate of the spatial covariance matrices is more important

How do we estimate the spatial covariance matrix?

- Spatial covariance estimation:

$$\hat{\boldsymbol{\Psi}}_{\boldsymbol{\nu}\boldsymbol{\nu},f} = \sum_{t=1}^{T} \gamma_{t,f}^{(\nu)} \mathbf{y}_{t,f} \mathbf{y}_{t,f}^{\mathsf{H}} \Big/ \sum_{t} \gamma_{tf}^{(\boldsymbol{\nu})}; \quad \boldsymbol{\nu} \in \{\mathbf{x}, \mathbf{n}\}$$

where: $\gamma_{t,f}^{(x)} = \hat{\Pr}(M_{t,f}^{(x)} = 1 | \mathcal{Y})$ speech presence prob. (SPP), speech mask

$\gamma_{t,f}^{(n)} = \hat{\Pr}(M_{t,f}^{(n)} = 1 | \mathcal{Y})$ noise presence prob., noise mask

How do we estimate the RTF?

- Estimation of RTF $\tilde{\mathbf{a}}_f$:
  - Solve above (generalized) eigenvalue problem: $\tilde{\mathbf{a}}_f = \mathbf{\Psi}_{\mathbf{nn},f} \mathbf{w}_f^{\mathrm{maxSNR}}$
  - Exploit nonstationarity of speech [Gannot et al., 2001] – not described here

- Advice: use beamformer formulation, which avoids explicit computation of RTF, e.g.,

$$\mathbf{w}_f^{\text{r1-SDW-MWF}} = \frac{\mathbf{\Psi}_{\mathbf{nn},f}^{-1} \mathbf{\Psi}_{\mathbf{xx},f}}{\mu + \mathrm{tr}\left\{ \mathbf{\Psi}_{\mathbf{nn},f}^{-1} \mathbf{\Psi}_{\mathbf{xx},f} \right\}} \mathbf{u}_1$$

[Souden et al., 2010]

Summary: processing steps

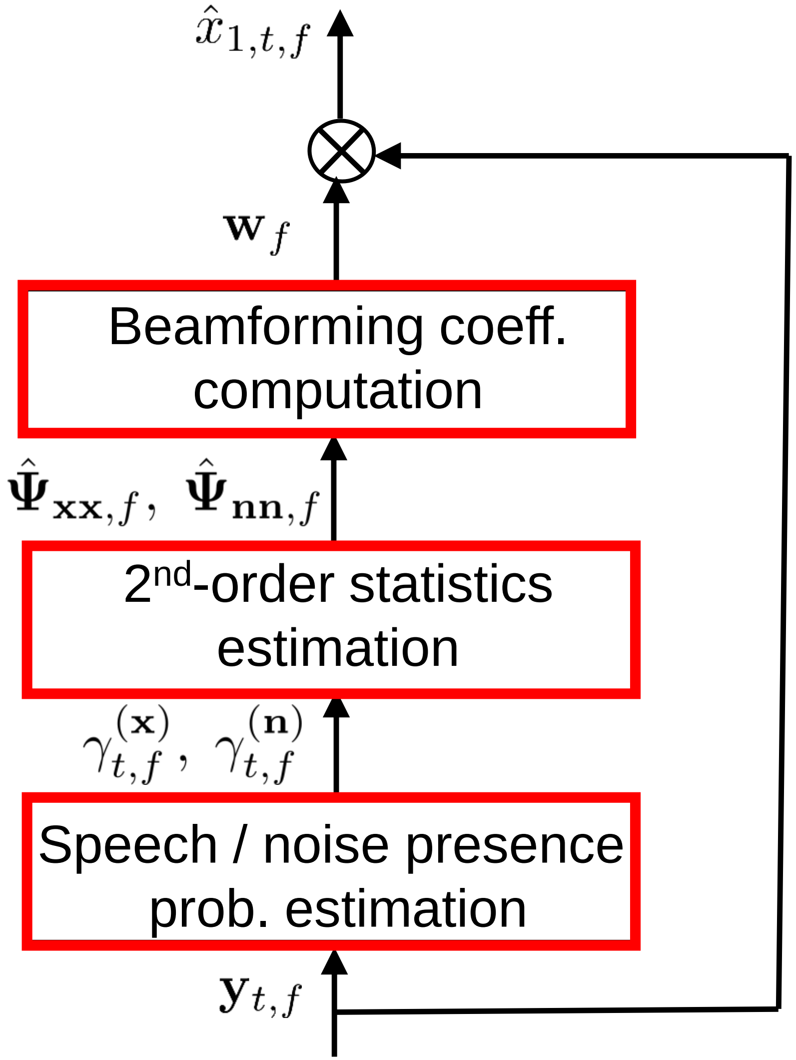
$$\hat{x}_{1,t,f} = \mathbf{w}_f^{\mathsf{H}} \mathbf{y}_{t,f}$$

e.g.: $$\mathbf{w}_f^{\text{r1-SDW-MWF}} = \frac{\hat{\mathbf{\Psi}}_{\mathbf{nn},f}^{-1} \hat{\mathbf{\Psi}}_{\mathbf{xx},f}}{\mu + \mathrm{tr}\left\{\hat{\mathbf{\Psi}}_{\mathbf{nn},f}^{-1} \hat{\mathbf{\Psi}}_{\mathbf{xx},f}\right\}} \mathbf{u}_1$$

$$\hat{\mathbf{\Psi}}_{\mathbf{xx},f} = \sum_t \gamma_{t,f}^{(\mathbf{x})} \mathbf{y}_{tf} \mathbf{y}_{tf}^{\mathrm{H}} \Big/ \sum_t \gamma_{tf}^{(\mathbf{x})}$$

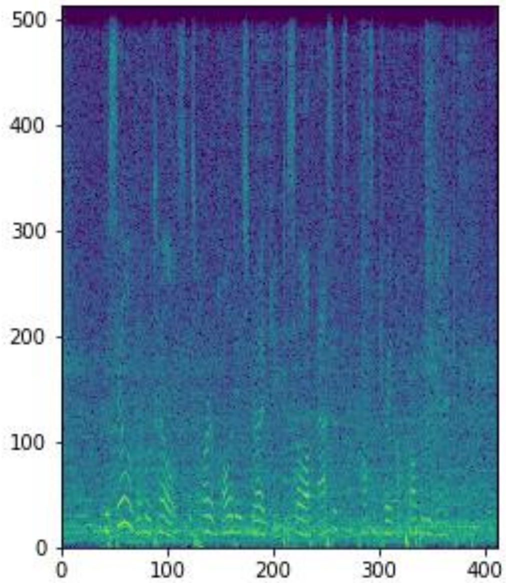
$$\hat{\mathbf{\Psi}}_{\mathbf{nn},f} = \sum_t \gamma_{t,f}^{(\mathbf{n})} \mathbf{y}_{tf} \mathbf{y}_{tf}^{\mathrm{H}} \Big/ \sum_t \gamma_{tf}^{(\mathbf{n})}$$

to be discussed next!



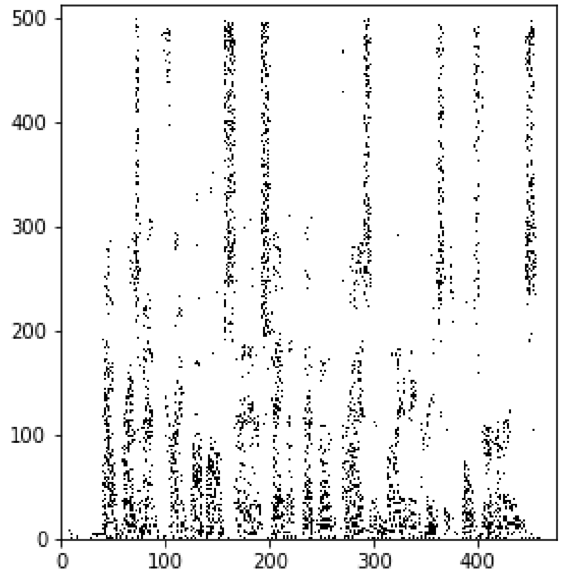
Speech Presence Probability (SPP) / mask estimation

Given:

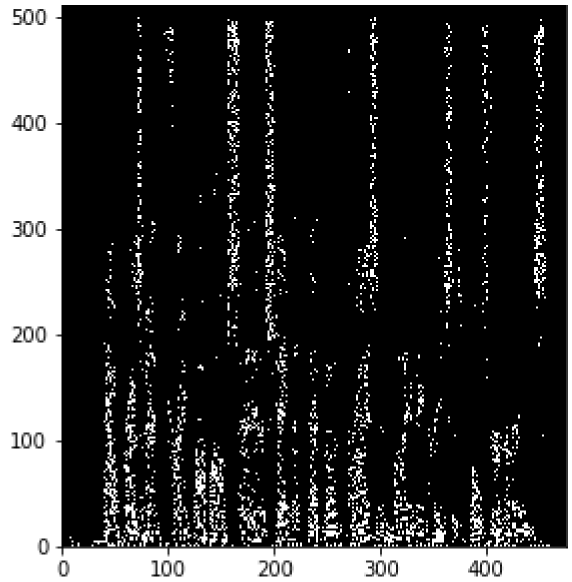


$\mathbf{y}_{t,f}$

Wanted:



$\gamma_{t,f}^{(\mathbf{x})}$



$\gamma_{t,f}^{(\mathbf{n})}$

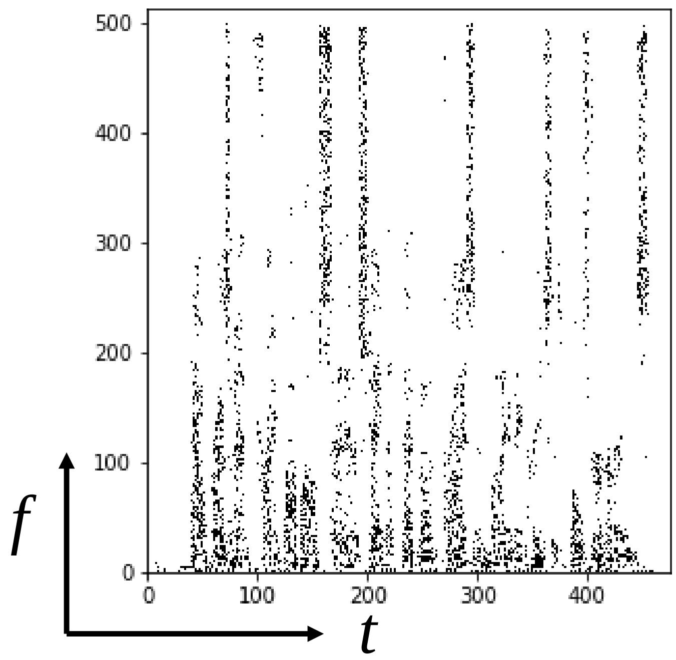
- Estimate for each tf-bin, the probability that it contains speech and the probability that it contains noise, using
  - spatial information
  - or spectral information
  - or both

Options for SPP estimation

- ~~Hand-crafted spectro-temporal smoothing~~
- Spatial mixture models
- Neural networks

Spatial mixture model

- Sparsity assumption [Yilmaz and Rickard, 2004]
  - 90% of the speech power is concentrated in 10% of the tf-bins
  - sparsity most pronounced for STFT window lengths of approx 64 ms



$$M_{t,f} := M_{t,f}^{(x)} = 1 - M_{t,f}^{(n)} \in \{0,1\}$$

$$\gamma_{t,f}^{(i)} := \hat{\Pr}(M_{t,f} = i | \mathbf{y}_{t,f}); i \in \{0,1\}$$

- Mixture model for vector of microphone signals $\mathbf{y}_{t,f}$ or for representation derived from it

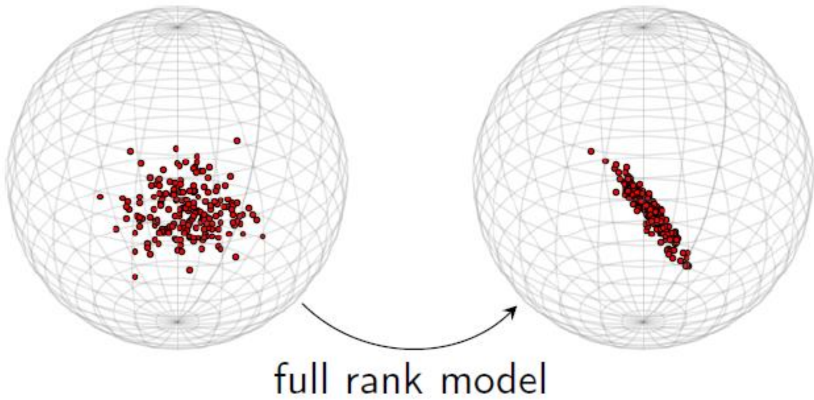
$$p(\mathbf{y}_{t,f}) = \sum_{i=0}^{1} \Pr(M_{t,f} = i) p(\mathbf{y}_{t,f} | M_{t,f} = i)$$

Example spatial mixture model

- Complex angular central Gaussian (cACG) Mixture Model for normalized observation vector $\tilde{\mathbf{y}}_{t,f} = \mathbf{y}_{t,f}/\|\mathbf{y}_{t,f}\|$ [Ito et al., 2016]:

$$p(\tilde{\mathbf{y}}_{t,f}) = \sum_{i=0}^{1} \Pr(M_{t,f}=i)p(\tilde{\mathbf{y}}_{t,f}|M_{t,f}=i) = \sum_i \pi_f^{(i)} \mathrm{cACG}(\tilde{\mathbf{y}}_{t,f};\mathbf{B}_f^{(i)})$$

$$\mathrm{cACG}(\tilde{\mathbf{y}}_{t,f};\mathbf{B}_f^{(i)}) = \frac{(M-1)!}{2\pi^M \det \mathbf{B}_f^{(i)}} \frac{1}{(\tilde{\mathbf{y}}_{t,f}^{\mathsf{H}}(\mathbf{B}_f^{(i)})^{-1}\tilde{\mathbf{y}}_{t,f})^M}$$



Parameter estimation

- Parameter Estimation via Expectation Maximization (EM) alg.
  - E-step: estimate source activity indicator $\gamma_{t,f}^{(i)}$ for all t, f and $i = 0,1$
  - M-step: estimate model parameters: $\pi_f^{(i)}, \mathbf{B}_f^{(i)}; \; i \in \{0, 1\}$
  - Iterate until convergence
- Actually, we are only interested in $\gamma_{t,f}^{(i)}$

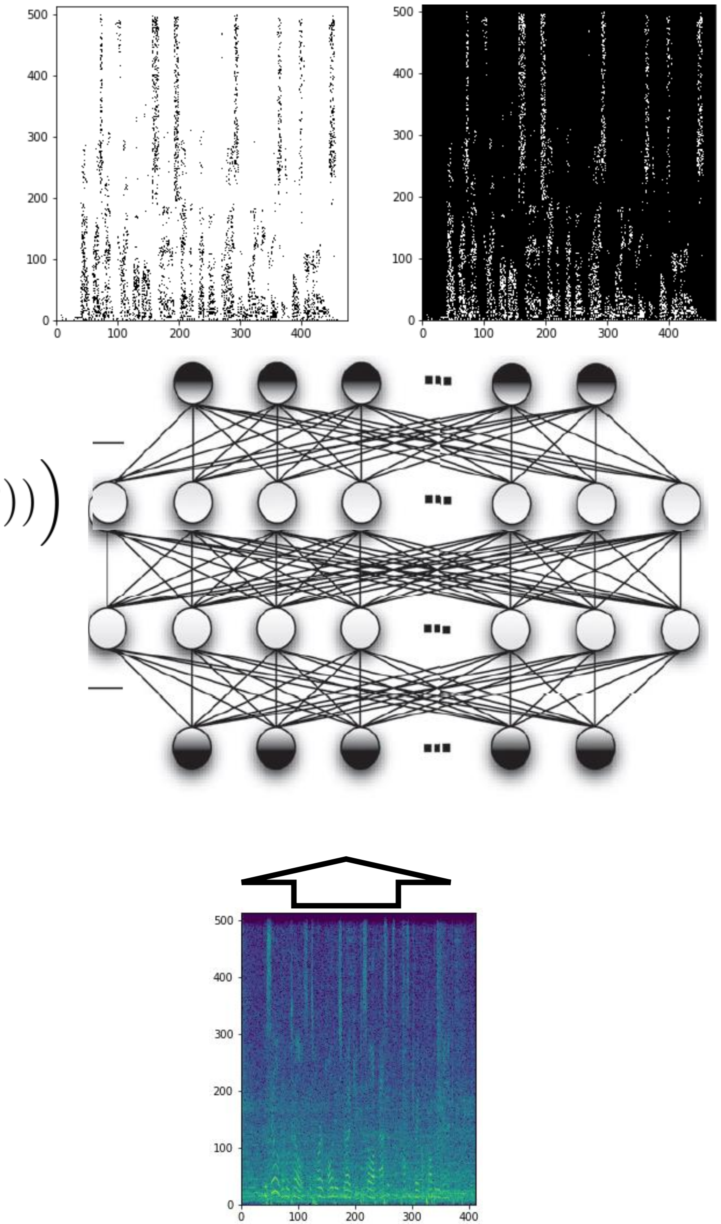
Note: separate EM for each frequency causes frequency permutation problem:
In one frequency i=1 may stand for speech, in another for noise!
Permutation solver required, e.g. [Sawada et al., 2011]
(or use permutation-free model with time-variant mixture weights [Ito et al., 2013])

SPP estimation with neural network

- SPP as supervised learning problem
  - Mask estimation formulated as classification problem
  - Objective function: binary cross entropy:

$$J(\theta) = -\sum_{\nu \in \{x,n\}} \sum_{t,f} \left(M_{t,f}^{(\nu)} \log \gamma_{t,f}^{(\nu)}(\theta) + (1 - M_{t,f}^{(\nu)}) \log(1 - \gamma_{t,f}^{(\nu)}(\theta)) \right)$$

- Note: masks need not sum up to one!



Example configuration

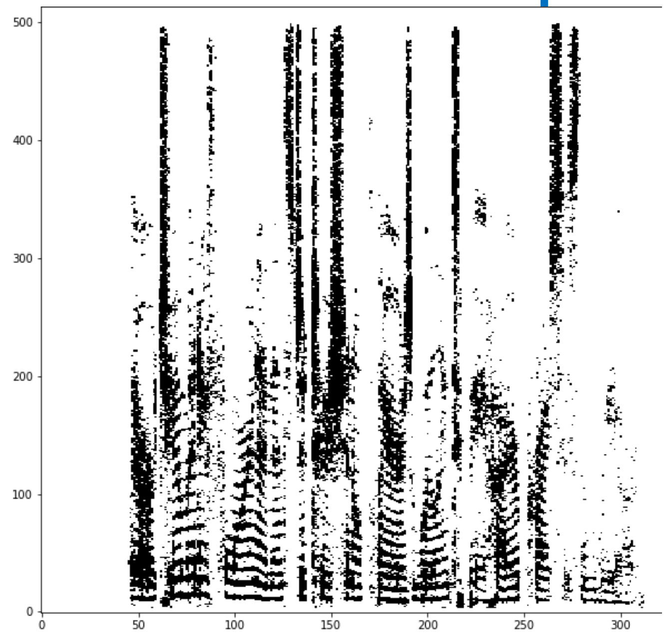
- Input: spectral magnitudes $|\mathbf{y}_{t,f}|$

Layer	Units	Type	Non-linearity	$p_{dropout}$
L1	256	BLSTM	Tanh	0.5
L2	513	FF	ReLU	0.5
L3	513	FF	ReLU	0.5
L4	1026	FF	Sigmoid	0.0

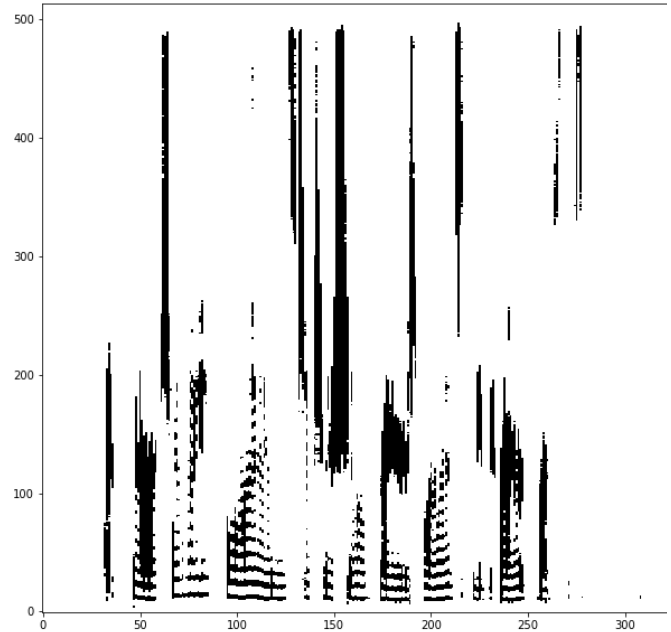
- Output: speech and noise masks $\gamma_{t,f}^{(x)}, \gamma_{t,f}^{(n)}$

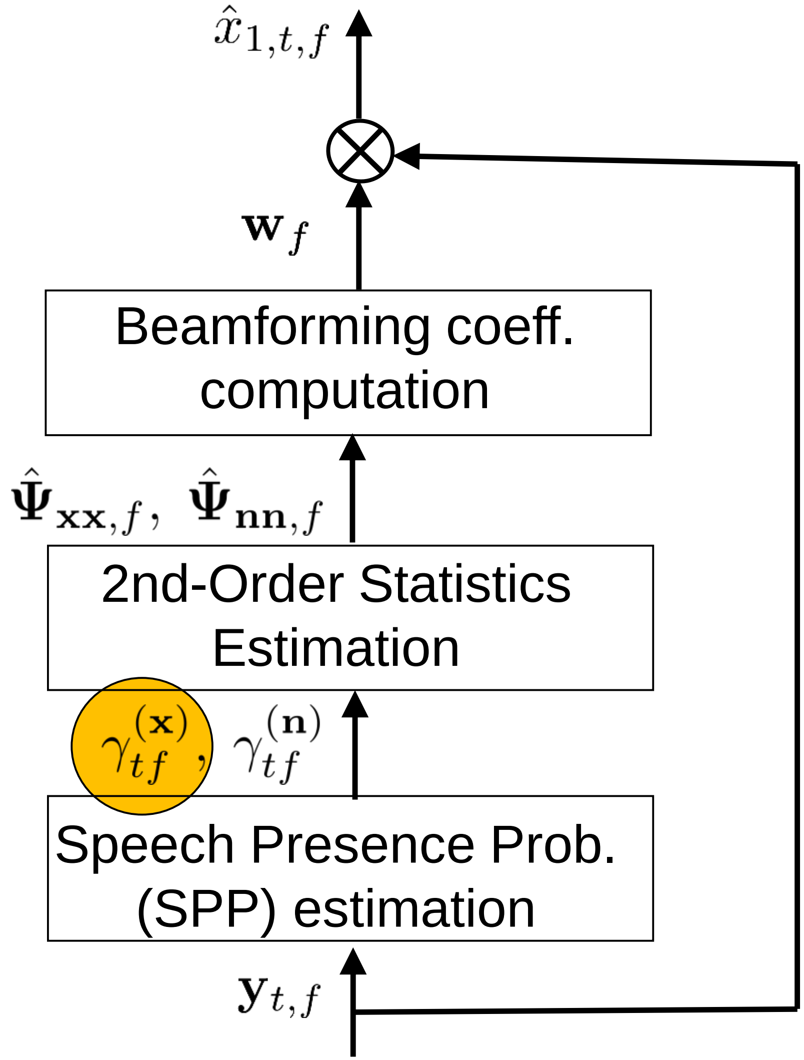
Example masks

Target speech mask $M_{t,f}^{(x)}$



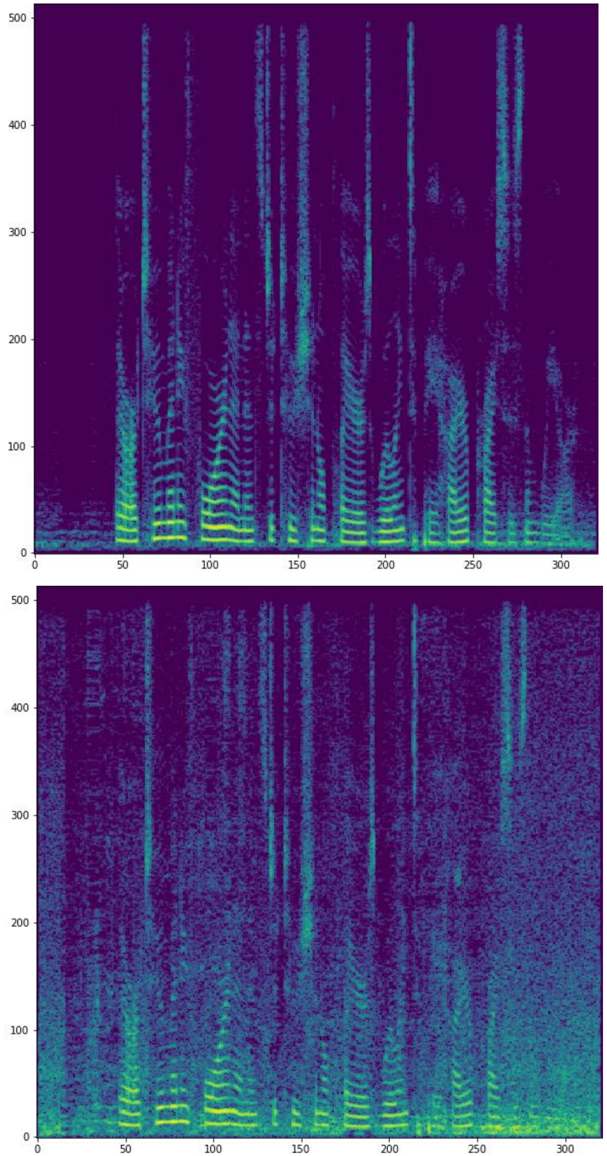
Estimated speech mask $\gamma_{t,f}^{(x)}$

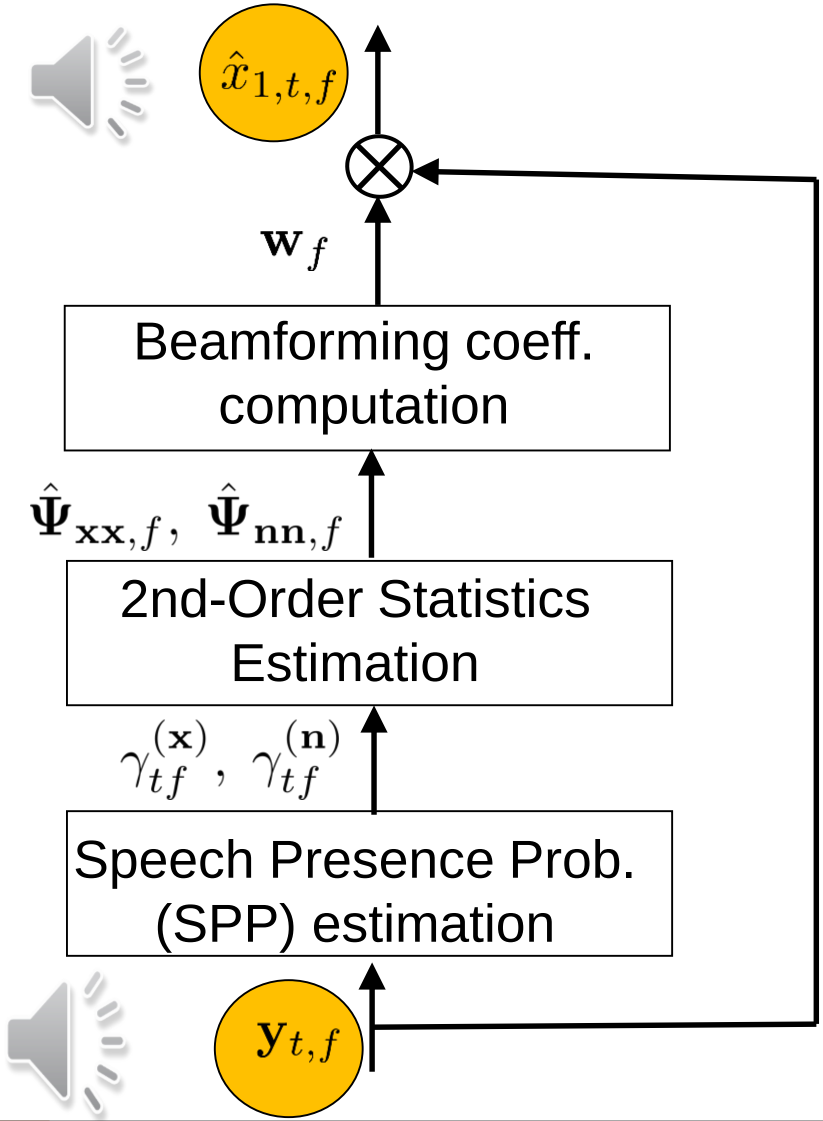




Demonstration NN-based mask estimation

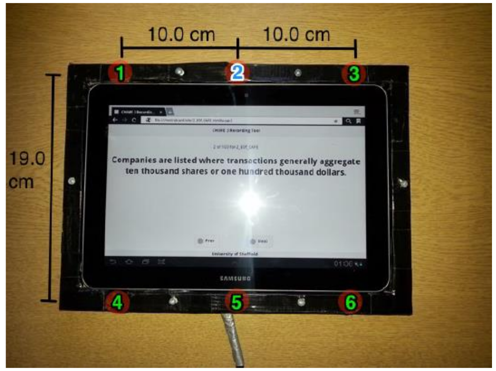
CHiME-3: Utterance ID:
f04_051c0112_str





ASR results: Spatial mixture model mask estimation

- CHiME-3 (2015) [Barker et al., 2017]
  - WSJ utterances
  - „Fixed“ speaker positions
  - Low reverberation
  - Noisy environment: bus, café, street, pedestrian
  - Trng set size: 18 hrs x 6 channels
- The winning system [Yoshioka et al., 2015, Higuchi et al., 2016] used a cACGMM spatial mixture model:



WER [%]	Dev Real	Test Real
No beamforming	9.0	15.6
DSB with DoA estimation	9.4	16.2
Spatial mixture model	4.8	8.9

ASR results: Neural network mask estimation

- CHiME-3 [Heymann et al., 2015]
  - Absolute WER values not comparable with last slide (different acoustic model, language model, data augmentation)

WER [%]	**Dev Real**	**Test Real**
No beamforming	18.7	33.2
NN supported beamforming	10.4	16.5

- CHiME-4 (2016):
  - All top 5 systems used mask-based beamforming (either NN or spatial mixture model)

Extensions

- Spatial mixture models
  - Other mixture models, e.g., Watson MM [Tran Vu and Haeb-Umbach, 2010]
  - On test utterance, with NN-based masks as priors $\Pr(M_{t,f} = i)$ [Nakatani et al., 2017]
- NN-Supported Beamforming
  - Cross-channel features, e.g., [Liu et al., 2018]
  - Block-online processing, e.g., [Boeddeker et al., 2018]
  - Used for dereverberation [Heymann et al., 2017b]

Pros and cons of two mask estimation methods

	Spatial mixture models	Neural networks
Spatial characteristics modeling	• **Strong**	• Moderate (use of cross-channel features at input)
Spectro-temporal characteristics modeling (for speech)	• Weak - Permutation problem • No concept of human speech (pros and cons)	• **Very strong** - Strong speech model based training
#channels required	• Multi-channel	• Single channel
Leverage training data	• No training phase	• **Yes**, but parallel data required
Adaptation to test condition	• **Strong** - Unsupervised learning applicable	• Weak - Poor generalization - Sensitive to mismatch

Table of contents in part II

Speaker-Conditioned Spectrogram Masking

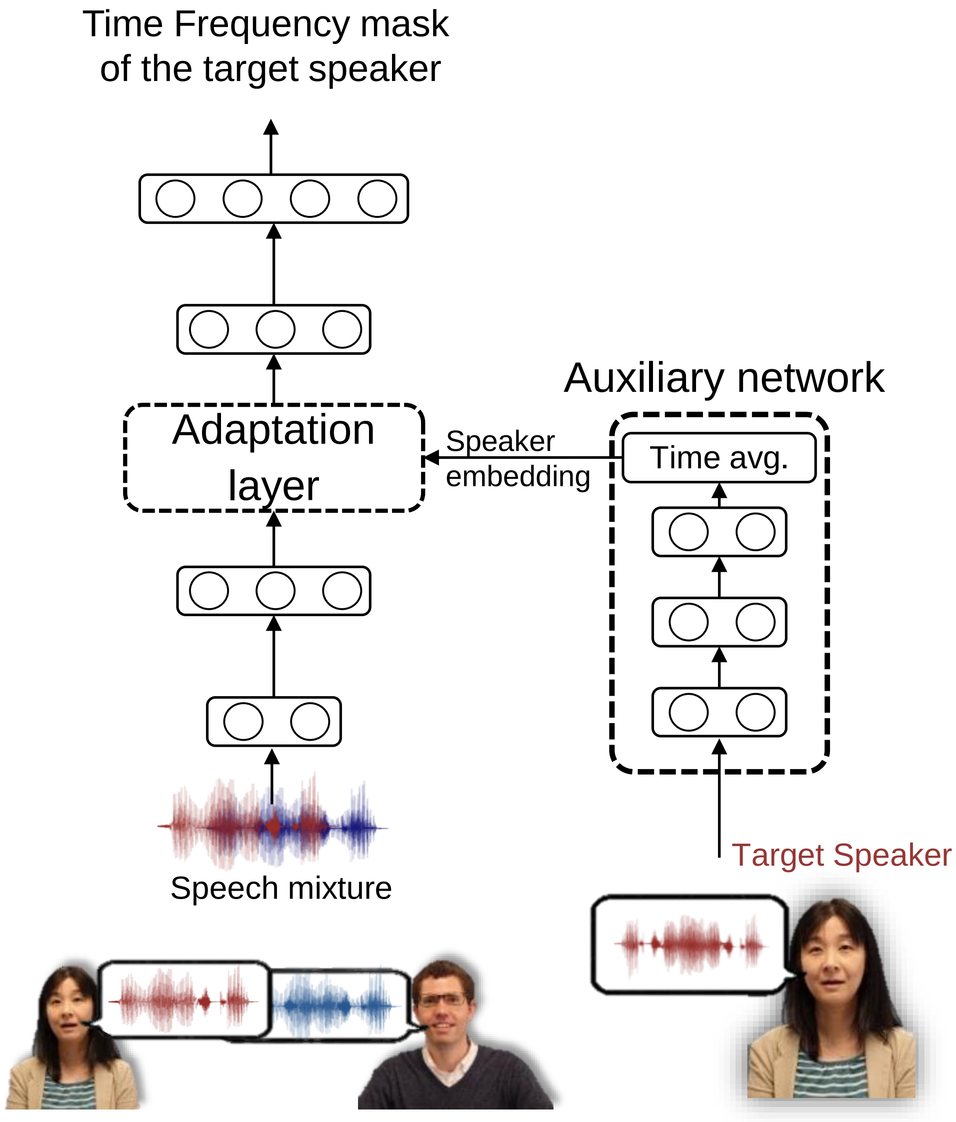
- In many application, we may be interested in recognizing speech from a target speaker even if there is noise or other people speaking, e.g., smart speaker

→ Target speaker extraction

  - Known target speaker position → use beamformer to extract speech from that direction
  - Unknown target speaker position → extract speaker based on his/her speech characteristics (SpeakerBeam)

- Idea of SpeakerBeam
  - NN for mask estimation can well discriminate a target speaker from noise, but not when interference is another speaker
  - This can be improved if the mask estimator is informed about the speaker to be extracted
  - We assume that we have about 10 sec. of enrollment/adaptation utterance spoken by the target speaker

SpeakerBeam [Zmolikova et al., 2017]



- Adaptation layer
  - Drive NN to output mask for the target speaker only, given target speaker embedding
  - Different implementations possible, e.g. factorized layer, scaling, etc.
- Auxiliary network
  - Compute speaker embedding given the enrollment/adaptation utterance
  - Implemented using sequence summary network [Vesely et al. 2016]
  - Jointly optimized with mask estimation NN
- SpeakerBeam performs 1ch processing to compute mask, but it can be combined with beamforming for multi-ch processing

Results [Zmolikova et al., 2019]

- WSJ2mix-MC
  - Artificial 2-speaker mixtures from WSJ utterances
  - 1ch no reverberation
  - 8 channels with reverberation T_{60} = 0.2 – 0.6 s

WER [%]	**1 ch (no reverb)**	**8 ch (w/ reverb)**
Single speaker	12.2	16.2
Mixtures	73.4	85.2
SpeakerBeam (1ch)	30.6	-
SpeakerBeam + Beamformer	-	22.5
SpeakerBeam + Beamformer (w/ AM joint training)	-	20.7

Software

- Spatial mixture models: https://github.com/fgnt/pb_bss
  - Different spatial mixture models
    - complex angular central Gaussian , complex Watson,von-Mises-Fisher
  - Methods: init, fit, predict
  - Beamformer variants
  - Ref: [Drude and Haeb-Umbach, 2017]

- NN supported acoustic beamforming: https://github.com/fgnt/nn-gev
  - NN-based mask estimator and maxSNR beamformer
  - Ref: [Heymann et al., 2016]
  - Part of Kaldi CHiME-3 baseline

Summary of part II

- Acoustic beamforming as a front-end for ASR
  - Exploits spatial information present in multi-channel input for noise suppression, which typical ASR feature sets (log-mel, cepstral) ignore
  - Leads to significant WER improvements
- SPP / Mask estimation is key component of beamformer
  - Both, spatial mixture models and neural networks are powerful mask estimators with complementary strengths
- Acoustic beamforming followed by DNN-based ASR is a typical representative of a combination of signal processing approaches with deep learning
  - Leads to interpretable, lightweight system compared to a NN with multi-channel input

But what about overall optimality? We‘ll come back to that…

Table of contents