# Deep Learning-based Speech Enhancement

Laurent Girin

Grenoble-INP / Phelma / GIPSA-lab

# An important preliminary information

- This is a preparation for the Speech / Audio Processing Project (32h) - These slides are available on Chamilo
- However, the SAPP currently comprises three subjects:
  - Speech enhancement in noise using deep neural networks
  - Speech enhancement in noise using conventional Wiener filters $\rightarrow$ NON deep! [detailed subject available on Chamilo]
  - Male-to-female voice conversion using conventional signal processing techniques $\rightarrow$ NON deep! [idem]
- You are free to choose the subject, but:
  - If you choose DLSE, you will get poor technical support from the project supervisors = only pedagogical support (more information at the end of presentation)
  - If you choose Wiener-SE or M2F-VC, you will have both pedagogical and technical support
  - Better with a balanced number of students on each subject
- IMPORTANT: Even if you do not choose DLSE, this presentation remains a good illustration of the use of DL in speech/audio processing

# Speech enhancement: The problem

- Task: Remove the noise from noisy speech recordings = estimate the clean speech signal
- Goal: improve the quality and intelligibility of speech for telecommunication systems, improve scores of automatic speech recognition (ASR) systems
- An old topic of signal processing. Current renewal with the arrival of home assistants + deep learning
- Single-channel (one-microphone recording): $x(t) = s(t) + b(t) \rightarrow \text{process} \rightarrow \hat{s}(t)$
- Multi-channel ($I$-microphone array recording): $\mathbf{x}(t) = \mathbf{s}(t) + \mathbf{b}(t) \in \mathbb{R}^I \rightarrow \text{process} \rightarrow \hat{s}(t)$

# Speech enhancement: The problem

- Task: Remove the noise from noisy speech recordings
  = estimate the clean speech signal
- Goal: improve the quality and intelligibility of speech for telecommunication systems, improve scores of automatic speech recognition (ASR) systems
- An old topic of signal processing. Current renewal with the arrival of home assistants + deep learning
- Single-channel (one-microphone recording):
  $x(t) = s(t) + b(t) \rightarrow \text{process} \rightarrow \hat{s}(t)$
- Multi-channel ($I$-microphone array recording):
  $\mathbf{x}(t) = \mathbf{s}(t) + \mathbf{b}(t) \in \mathbb{R}^I \rightarrow \text{process} \rightarrow \hat{s}(t)$
- Solution(s): 50 years of literature !!! $\rightarrow$ We are going to see:
  - Basics of speech enhancement (100% signal processing)
  - New trends: Deep-learning based signal processing
- Note: In the SAPP, we will focus on single-channel SE, but here we will also have a very short overview of multi-channel SE, just for fun!

# TF-domain processing

- Almost all methods (old school SP and "modern" DL) work in the time-frequency (TF) domain: SE is typically considered as an estimation problem in the TF domain
- Most methods use the **short-term Fourier transform**[1]
  = a series of discrete Fourier transform (DFT) applied on successive short-term *frames* of signal, with some overlap:

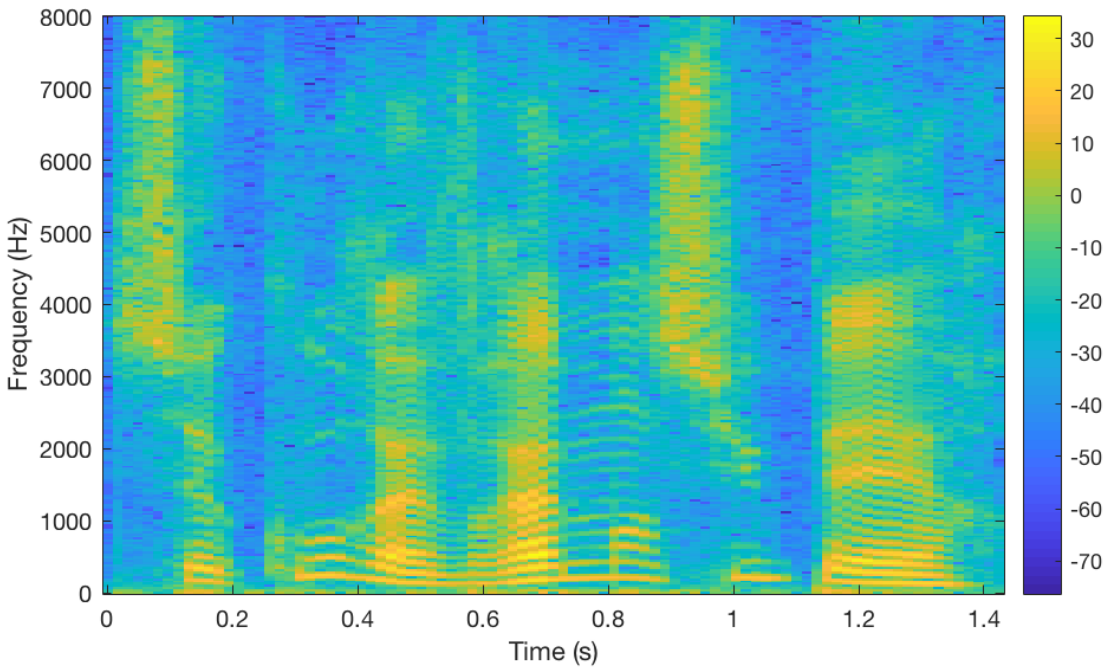
$$x_{fn} = \sum_{t=0}^{N-1} x(t+nH)w(t)e^{-j2\pi\frac{ft}{N}}, \quad f \in [0, N-1], n \in [0, P-1]$$

  - $x(t)$ = time-domain digital signal
  - $w(t)$ = STFT analysis window
  - $f$ = frequency bin, $n$ = time frame index, $N$ = window size, $H$ = hop size (aka shift), $P$ = number of frames

[1] L. R. Rabiner and B. Gold. *Theory and application of digital signal processing*. Englewood Cliffs, NJ, 1975.

# TF-domain processing

- Typical settings: $N = 512$ or 1024 (32 ms or 64 ms at 16-kHz sampling rate) and $H = N/2$ (50% overlap)
- Enables to follow speech non-stationarity + optimize DFT calculation with FFT algorithm
- Example of speech STFT log-magnitude spectrogram



# TF-domain processing

- After processing, time-domain signal reconstruction is obtained with inverse STFT

$$x_n(t) = x(t+nH)w(t) = \sum_{f=0}^{N-1} x_{fn} e^{j2\pi \frac{ft}{N}}, \quad n \in [0, P-1], t \in [0, N-1]$$

  and overlap-add = summation of the contributions of the different STFT frames, weighted by a synthesis window
- There exist a general condition on analysis and synthesis window for perfect signal reconstruction (in the absence of modification in the transformed domain), see SP literature. For simplicity we can take analysis window = synthesis window in a reduced set of windows

# Speech enhancement measures

- Input SNR = ratio of signal power over noise power in the mixture signal

$$SNR_{\text{in}}\,(\text{dB}) = 10\log_{10}\frac{P_s}{P_b} \quad \text{with} \quad P_s = \sum_{t=0}^{T-1} s(t)^2$$

- Output SDR = ratio of signal power over residual noise (= distortion) power in the enhanced signal

$$SDR_{\text{out}}\,(\text{dB}) = 10\log_{10}\frac{P_s}{P_e} \quad \text{with} \quad e(t) = \hat{s}(t) - s(t)$$

  Careful: Not robust to scaling, time shift or dephasing
  $\rightarrow$ may need to introduce compensation
- Gain = $SDR_{\text{out}} - SNR_{\text{in}}$
- Can be calculated in the time domain or in the DFT domain, thanks to orthogonal properties of DFT. Averaged frame-wise versions are called segmental SNR/SDR

# Old school signal processing approaches (1970-2010)[3]

## (single-channel case)

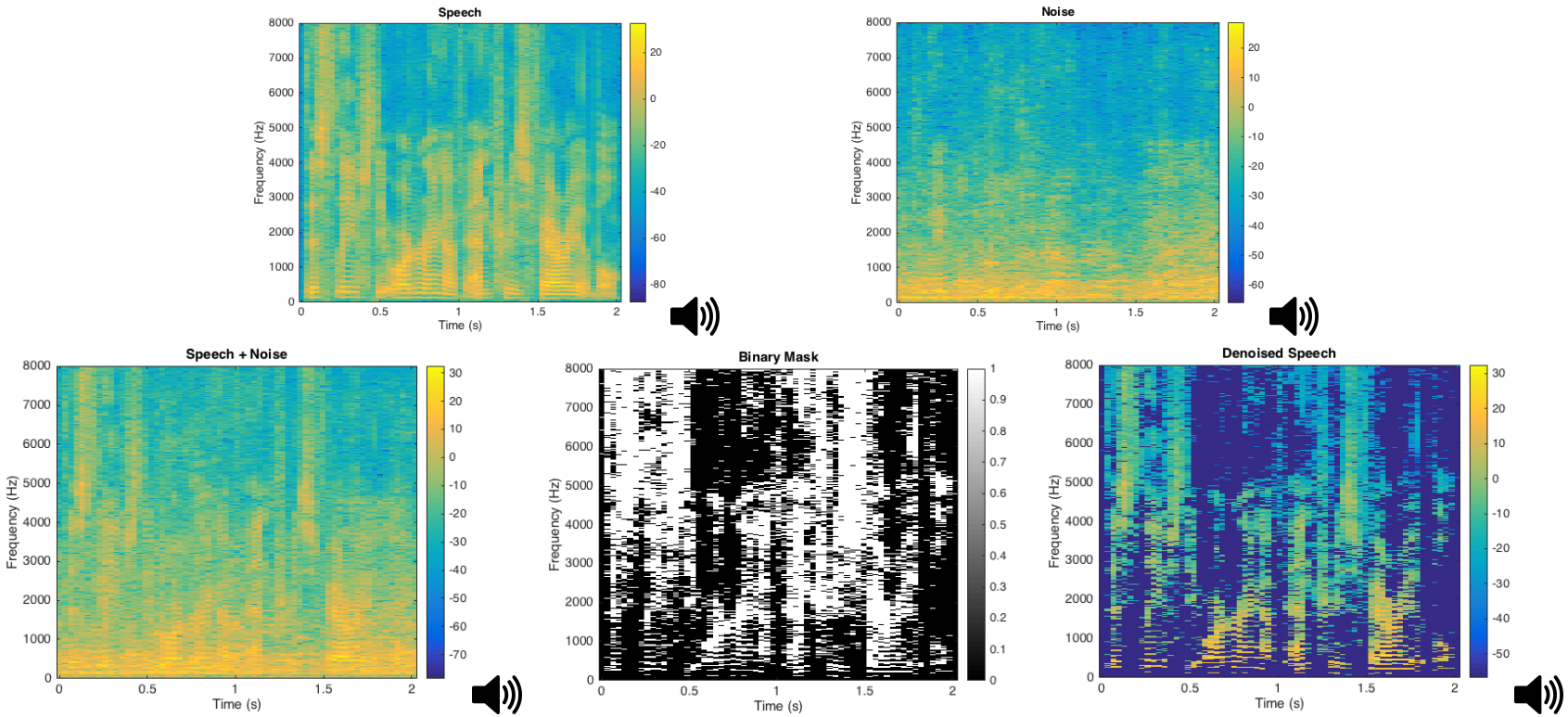
- Typically an estimation problem in the STFT domain
- Historical method 1: Spectral subtraction and Wiener filtering
  - Wiener filter = optimal MMSE (= max $SDR_{\text{out}}$) linear estimator for stationary signal and noise: $\widehat{s}_f = \frac{\gamma_{s,f}}{\gamma_{s,f}+\gamma_{b,f}} x_f$ with $\gamma_{s,f} = E[|s_f|^2]$ = power spectral density of $s(t)$
  - In practice, detect voice activity/silence, estimate the short-term PSD of noise during speech silence, then apply power spectral subtraction $\widehat{\gamma}_{s,fn} = \max(|x_{fn}|^2 - \widehat{\gamma}_{b,fn}, 0)$ and (practical) Wiener filtering $\widehat{s}_{fn} = \frac{\widehat{\gamma}_{s,fn}}{\widehat{\gamma}_{s,fn}+\widehat{\gamma}_{b,fn}} x_{fn}$
- Historical method 2: Bayesian estimation of short-term spectral amplitude $|s_{fn}|^2$
- Speech and noise must have different spectral characteristics. Usually, noise is assumed more stationary than speech
- Single-channel *multispeaker* separation is extremely difficult.

[2] Y. Ephraim and D. Malah. "Speech enhancement using a minimum-mean square error short-time spectral amplitude estimator". In: *IEEE Transactions on Acoustics, Speech, and Signal Processing* 32.6 (1984).

[3] P. Loizou. *Speech enhancement: theory and practice*. CRC press, 2007.

# The CASA approach (mid 90's-2010's)[4]

- Based on the sparsity of speech/audio in the TF domain
- Estimate a binary mask $M_{fn} = 1$ if $x_{fn}$ is dominated by speech, $M_{fn} = 0$ if $x_{fn}$ is dominated by noise, and then $\hat{s}_{fn} = M_{fn} \cdot x_{fn}$




- Extension to soft masks ($0 \leq M_{fn} \leq 1$), related to Wiener filters

[4] DL Wang and G.J. Brown. *Computational auditory scene analysis: Principles, algorithms, and applications*. Wiley-IEEE press, 2006.

# The CASA approach (mid 90's-2010's)

- In practice the mask is estimated from the noisy speech spectrogram. You must identify and group TF points with speech characteristics (e.g. harmonicity, common onsets/offsets/modulations.) This is done with a mixture of techniques involving statistical signal processing, speech analysis, human audition modeling, computer vision and pattern analysis, etc.
- This is a very difficult task in single-channel configuration, especially when noise is strong
- Not suited for multispeaker separation
- Relatively "poor" results (compared to ideal binary masking which can be impressive); Limited quality of enhanced signals (musical noise artefacts); Not very well suited for ASR

# Multichannel speech enhancement

- $\mathbf{x}(t) = \sum_{j=1}^{J} \mathbf{y}_j(t) + \mathbf{b}(t) \in \mathbb{R}^I$ is recorded from a distant $I$-microphone array.
- Here we can have several directional sound sources + diffuse noise. $\mathbf{y}_j(t)$ is the multichannel image of mono source signal $s_j(t)$ filtered by the source-to-microphone channel
- We want to estimate $\mathbf{y}_1(t) = \mathbf{y}_{\text{target}}(t)$ (or $s_1(t) = s_{\text{target}}(t)$) and remove the other sources and the noise
- Exploit the spatial diversity of sources (in addition to spectral diversity) to build separating spatial filters.
- Convolutive source image model + Multiplicative Transfer Function approximation in the STFT domain: $\mathbf{y}_j(t) = \sum_{\tau=0}^{L_a-1} \mathbf{a}_j(\tau) s_j(t-\tau) \leftrightarrow \mathbf{y}_{j,fn} \approx \mathbf{a}_{j,f} s_{j,fn}$
- $\mathbf{a}_{j,f}$ is the steering (directional) vector of source $j$, that depends on source-to-sensor position and room acoustics
- In the STFT domain: $\mathbf{x}_{fn} \approx \sum_{j=1}^{J} \mathbf{a}_{j,f} s_{j,fn} + \mathbf{b}_{fn} = \mathbf{A}_f \mathbf{s}_{fn} + \mathbf{b}_{fn}$

# Classical signal processing solutions[7]

- Beamforming = Estimate a spatial linear filter $\mathbf{w}_f \in \mathbb{C}^I$ so that we have: $\hat{s}_{1,fn} = \mathbf{w}_f^\top \cdot \mathbf{x}_{fn}$
- Different optimality criteria lead to different BF solutions. A few examples (assuming $\mathbf{a}_1$ is estimated and either only noise is present or all interfering sources are included in $\mathbf{b}$. $f$ and $n$ are omitted for clarity):
  - $\mathbf{w}_{\text{SDW-MWF}} = \underset{\mathbf{w}}{\text{argmin}}\left(\{| 1 - \mathbf{a}_1^H\mathbf{w} |^2 \sigma_{s_1}^2 + \mu\mathbf{w}^H\mathbf{\Sigma_b}\mathbf{w}\right\} = \frac{\sigma_{s_1}^2 \mathbf{\Sigma_b}^{-1}\mathbf{a}_1}{\mu+\sigma_{s_1}^2 \mathbf{a}_1^H\mathbf{\Sigma_b}^{-1}\mathbf{a}_1}$
  - $\mathbf{w}_{\text{MWF}} = \frac{\sigma_{s_1}^2 \mathbf{\Sigma_b}^{-1}\mathbf{a}_1}{1+\sigma_{s_1}^2 \mathbf{a}_1^H\mathbf{\Sigma_b}^{-1}\mathbf{a}_1} \qquad (\mu = 1)$
  - $\mathbf{w}_{\text{MVDR}} = \frac{\mathbf{\Sigma_b}^{-1}\mathbf{a}_1}{\mathbf{a}_1^H\mathbf{\Sigma_b}^{-1}\mathbf{a}_1} \qquad (\mu = 0)$
  - $\mathbf{w}_{\text{MSNR}} = \underset{\mathbf{w}}{\text{argmax}}\left\{ \frac{|\mathbf{a}_1^H\mathbf{w}|^2}{\mathbf{w}^H\mathbf{\Sigma_b}\mathbf{w}} \right\} = k\mathbf{\Sigma_b}^{-1}\mathbf{a}_1$ Implemented with GEV-BF[5]
- Strong connexion with audio source separation[6]

---

[5] E. Warsitz and R. Haeb-Umbach. "Blind acoustic beamforming based on generalized eigenvalue decomposition". In: *IEEE Transactions on Audio, Speech, and Language Processing* 15.5 (2007), pp. 1529–1539.

[6] S. Gannot et al. "A consolidated perspective on multimicrophone speech enhancement and source separation". In: *IEEE/ACM Transactions on Audio, Speech and Language Processing* 25.4 (2017), pp. 692–730.

[7] B. D. Van Veen and K. M. Buckley. "Beamforming: A versatile approach to spatial filtering". In: *IEEE ASSP Magazine* 5.2 (1988), pp. 4–24.

# CASA solutions

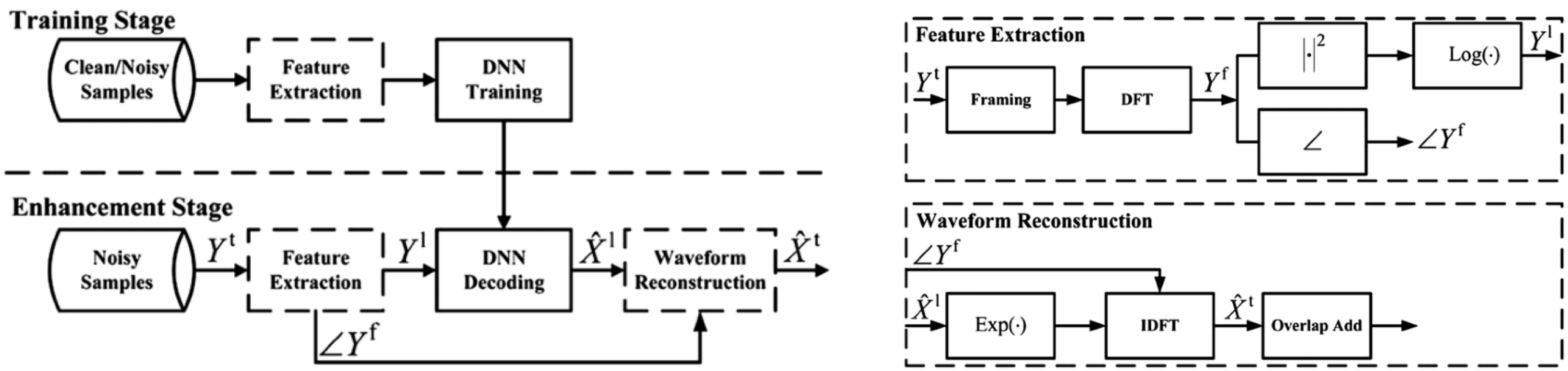
- General principle: the (binary or soft) TF masks are estimated from spatial information[8]
- General pipeline:
  - extraction of spatial features from multichannel observed signal, e.g. frequency-wise interchannel time/phase/level difference (ITD / IPD / ILD)
  - clustering of TF bins into sources (e.g. using mixture models)
  - generation of masks: $M_{j,fn} = 1$ and $M_{k \neq j,fn} = 0$ if TF-bin $\{fn\}$ is associated to source $j$
  - masking: $\hat{s}_{j,fn} = M_{j,fn} \cdot x_{i,fn}$ or average across mixture channels
- Good separation for a reasonable amount of sources, reasonable spatial separation, and low reverberation. Good interference rejection but quality is still quite limited (musical noise artefacts). Not very well suited for ASR.

[8] DL Wang and G.J. Brown. *Computational auditory scene analysis: Principles, algorithms, and applications*. Wiley-IEEE press, 2006.

# Deep learning based single-channel speech enhancement

- General principle: SE is turned into a supervised/data-driven regression problem using DNNs
- Basically, two strategies:
  - Regression from noisy speech spectrogram to clean speech spectrogram + use of the phase of noisy signal
  - Regression from noisy speech spectrogram to mask + application of the estimated mask to noisy signal

  (spectrogram stands for magnitude or power or log-magn. spectrogram)




(Figure taken from[9])

[9] Y. Xu et al. "An experimental study on speech enhancement based on deep neural networks". In: *IEEE Signal Processing Letters* 21.1 (2013), pp. 65–68.

# Deep learning based single-channel speech enhancement

- Within 2013-2015, tens of papers to discuss the effects of different DNN models (FF-DNNs, CNNs, LSTMs, etc.), different i/o data representations, different training criteria and strategies, different types of masks, etc., often leading to similar results[10]
- Requires a huge amount of parallel clean and noisy training data (not really a problem these days)
- New standards of performances – Typically 8–12dB gain = +3–10dB compared to old school SP and CASA
- Some trends:
  - Mask estimation works better / is easier to handle than direct estimation of clean spectra
  - CNNs, LSTMs and their combination (CRNN) is SoA

[10] DL Wang and J. Chen. "Supervised speech separation based on deep learning: An overview". In: *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 26.10 (2018), pp. 1702–1726.

# Complement: Deep learning based single-channel speech **separation**

- In single-channel configuration, **separation of two speakers** (and more) is much more difficult than SE in noise because target and interference are similar. So how can a DNN (or CASA model) learn two different models of the same thing?
- In other words, if the input is $\mathbf{x} = \mathbf{s}_1 + \mathbf{s}_2$ and the output is (a concatenation of) $\mathbf{s}_1$ and $\mathbf{s}_2$, how can the network work for both $(\mathbf{s}_1, \mathbf{s}_2)$ and $(\mathbf{s}_2, \mathbf{s}_1)$? This is called, the speaker permutation problem.

# Deep learning based single-channel speech separation
## One breakthrough technique: Deep Clustering[11] [12]

- Huge BLSTM network trained to project data into a separating embedding space
- Supervised training with mixture and individual source signals
- Input = TF power spectrogram of mixture (for a complete utterance) reshaped as a vector, output = matrix of embedding vectors of fixed arbitrary dimension (1 embedding vector for each input TF bin). At training time, calculation of a similarity matrix encoding if 2 TF bins belong to the same source or not. Then, based on this matrix, embeddings vectors corresponding to the same source are forced to be aligned whereas embeddings vectors corresponding to different sources are forced to be orthogonal. At testing time, clustering of the embedding vectors leads to efficient separation (done with masks).
- Unpreceded separation performance for a two-speaker single-channel mixture (typically up to 15dB gain, close to oracle TF masking)

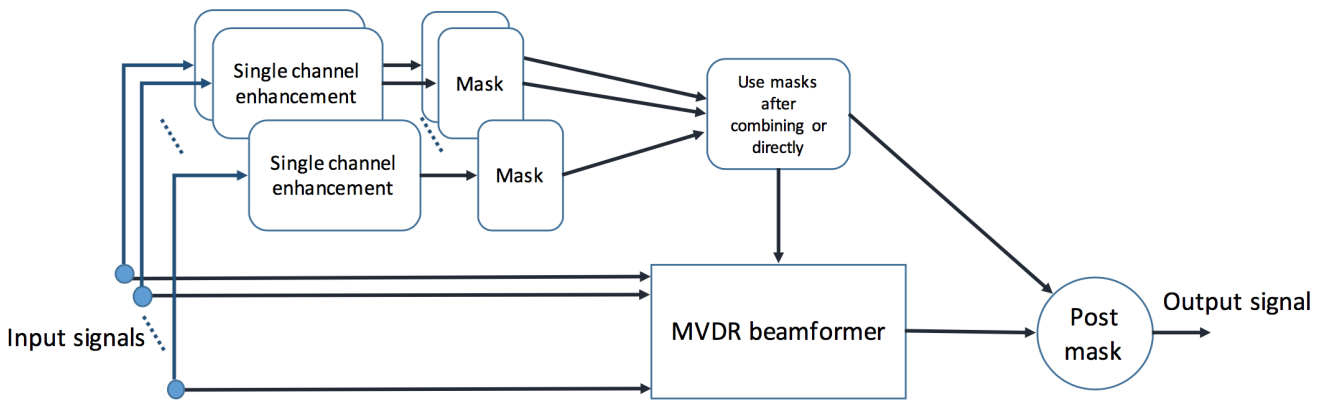
---

[11] J. R. Hershey et al. "Deep clustering: Discriminative embeddings for segmentation and separation". In: *IEEE ICASSP*. 2016.

[12] Y. Isik et al. "Single-channel multi-speaker separation using deep clustering". In: *arXiv preprint arXiv:1607.02173* (2016).

# Deep learning based multi-channel speech enhancement

- First approach: Combination of DL-based single-channel speech enhancement (for recovering the spectral information) and multichannel beamforming (for exploiting the spatial info)
- A typical example[13]: LSTM networks applied on each mixture channel provide TF masks, used to estimate $\mathbf{\Sigma_s}$ (or $\sigma^2_{s_1}$ and $\mathbf{a}_1$) and $\mathbf{\Sigma_b}$, used in turn to build BF filters

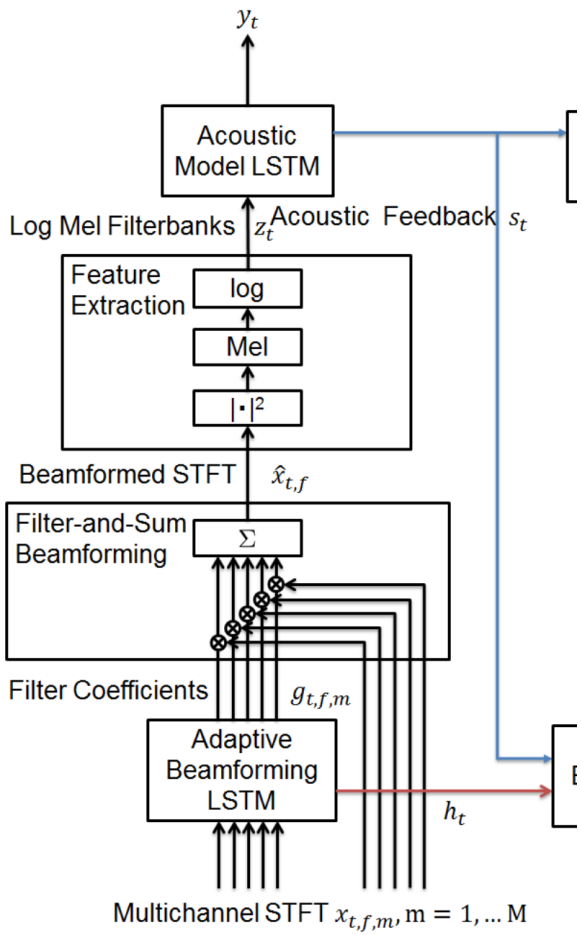



- BF filtered speech is observed to provide better ASR scores than "direct" mask-filtered speech
- Indep. of mic array config. Training only concerns LSTM net.

[13] H. Erdogan et al. "Improved MVDR Beamforming Using Single-Channel Mask Prediction Networks". In: *INTERSPEECH*. 2016.

# Deep learning based multi-channel speech enhancement

- Second approach: Direct estimation of BF filters with DNNs
- A typical example:[a] TF-domain "adaptive" processing. LSTM network takes multichannel STFT input (with concatenation of real and imaginary parts, frequency bins and channels) and outputs adaptive multichannel BF filter coefficients
- Joint end-to-end training of BF network and ASR acoustic model (training criterion = optimization of ASR scores)




[a] Z. Meng et al. "Deep long short-term memory adaptive beamforming networks for multichannel robust speech recognition". In: *IEEE ICASSP*. 2017.

# DLSE in the Speech/Audio Processing Project

- Tasks = what you have to do (basically)
  - Implement a Deep Learning based single-channel speech enhancement system
  - Experiment!!! Report and analyse results!!!
- Objectives: Understand what is happening, do better
- Means = what you can/should use
  - Matlab + Deep Learning Toolbox [NOT currently available at Phelma] [light help from supervisors can be expected] or any DL framework (Tensorflow, Keras, Pytorch, whatever) [no help from supervisors can be expected]; In other words: You must use your own computer, have some DL framework installed on it, and learn how to use it on your own
  - A few available signal processing routines in Matlab will be provided (e.g. STFT and inverse STFT implementation)
  - Data! Clean speech dataset + noise dataset will be provided
  - Supervisors (moderately)
  - Literature! (very important; see next slide)

# About the literature

- There are good papers and bad papers
- Example: There is actually an "official" Matlab example of DLSE at `https://fr.mathworks.com/help/deeplearning/examples/denoise-speech-using-deep-learning-networks.html`.
  It is based on [14], which is not a very good paper...
- Other (better) papers: [15] [16] [17] [18] [19] etc. + review in [20]

[14] S. R. Park and J. W. Lee. "A Fully Convolutional Neural Network for Speech Enhancement". In: *Interspeech*. 2017.

[15] S.-W. Fu, Y. Tsao, and X. Lu. "SNR-Aware Convolutional Neural Network Modeling for Speech Enhancement." In: *Interspeech*. 2016, pp. 3768–3772.

[16] J. Chen, Y. Wang, and DL. Wang. "A feature study for classification-based speech separation at low signal-to-noise ratios". In: *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 22.12 (2014), pp. 1993–2002.

[17] P.-S. Huang et al. "Joint optimization of masks and deep recurrent neural networks for monaural source separation". In: *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 23.12 (2015), pp. 2136–2147.

[18] J. Du et al. "A regression approach to single-channel speech separation via high-resolution deep neural networks". In: *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 24.8 (2016), pp. 1424–1437.

[19] S. Chazan, S. Gannot, and J. Goldberger. "A phoneme-based pre-training approach for deep neural network with application to speech enhancement". In: *IEEE International Workshop on Acoustic Signal Enhancement (IWAENC)*. 2016.

[20] DL Wang and J. Chen. "Supervised speech separation based on deep learning: An overview". In: *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 26.10 (2018), pp. 1702–1726.

# The Speech/Audio Processing project: TODO list

- Go to Chamilo, get the 3 documents (these slides + the other 2 subjects), read them and choose a subject with a colleague
- If you choose the DLSE subject:
  - Install some DL framework on your own computer and learn how to use it
  - Get familiar with basic signal processing routines (e.g. STFT)
  - Generate and organise training/validation/test data
  - Implement and compare different DNN models and their combinations
  - Compare different data representations
  - Compare different TF masking strategies
  - Reflexion on the use of SE evaluation metrics/measures
  - Bonus: Theoretical derivations on TF masks, e.g. proof of optimality for Wiener filters, search for other types of masks (you may find out that there are better masks than Wiener!)
- else (= if you choose Wiener-SE or M2F-VC)
  - Objectives are detailed in the documents
  - You will use Matlab (at Phelma or on your own computer)
- In any case: Have fun!!!