Microsoft

Sound capture and speech enhancement for speech-enabled devices

Dr. Ivan Tashev, Partner Software Architect
Dr. Sebastian Braun, Researcher

Audio and Acoustics Research Group,
Microsoft Research Labs, Redmond, WA, USA

Agenda

- Audio processing pipeline and statistical speech enhancement
- Application of deep learning methods in speech enhancement
- Conclusions

Introduction and Brief History

- Sound capture? Speech enhancement?
- Speech enhancement pipeline in Windows XP
  - NetMeeting – grandfather of Skype, Teams, etc.
- Microphone array support in Windows Vista
  - For Windows Live Messenger
- Microsoft Auto Platform
- Kinect for Xbox 360, for Windows, for Xbox One, for Azure
- HoloLens, HoloLens 2, Mixed Reality Platform
- Major update in Windows 10
- Teams

Microsoft Auto

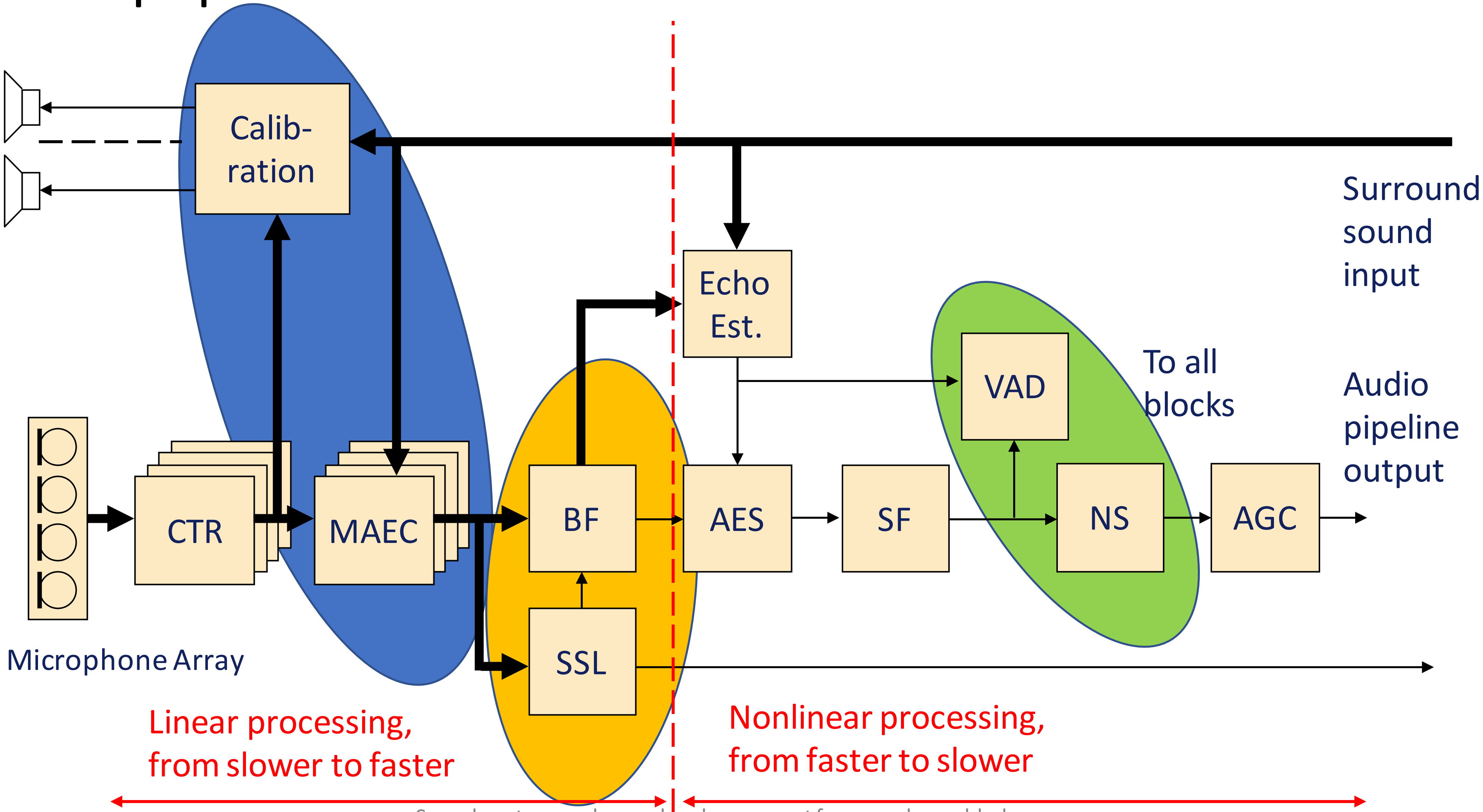
Windows Mixed Reality

Microsoft HoloLens

Windows 10

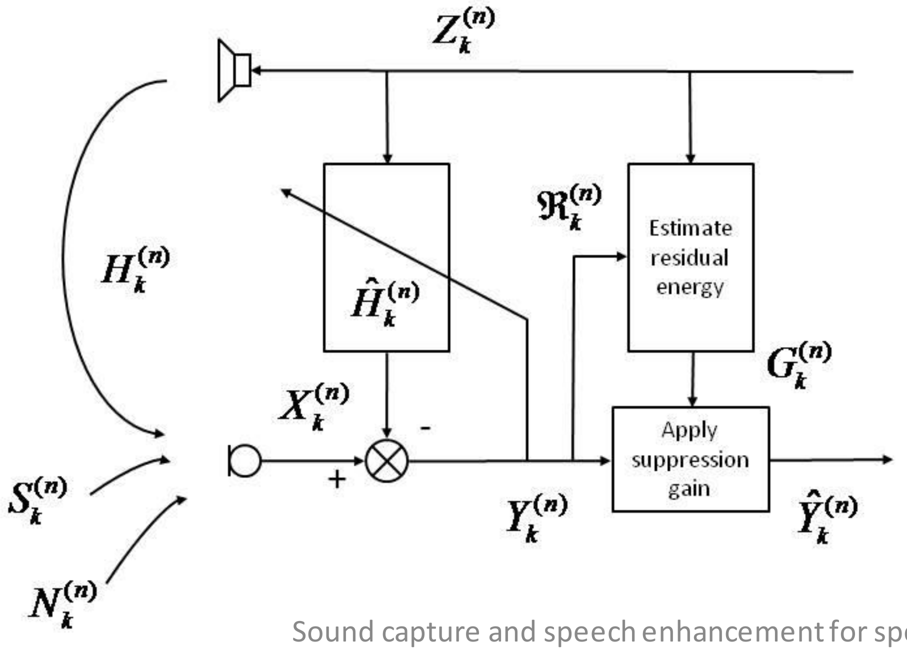
Audio processing pipeline and statistical speech enhancement

Audio pipeline architecture



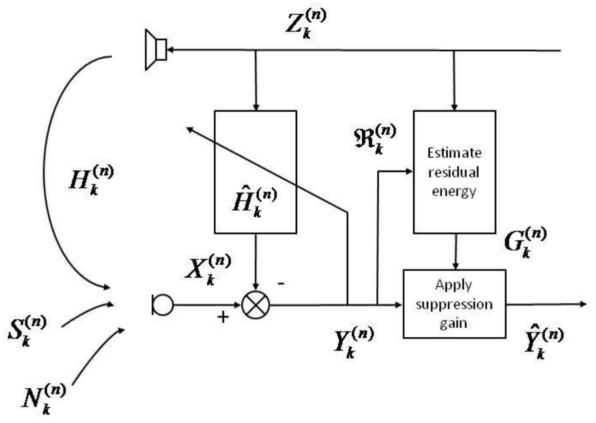
Acoustic echo reduction systems

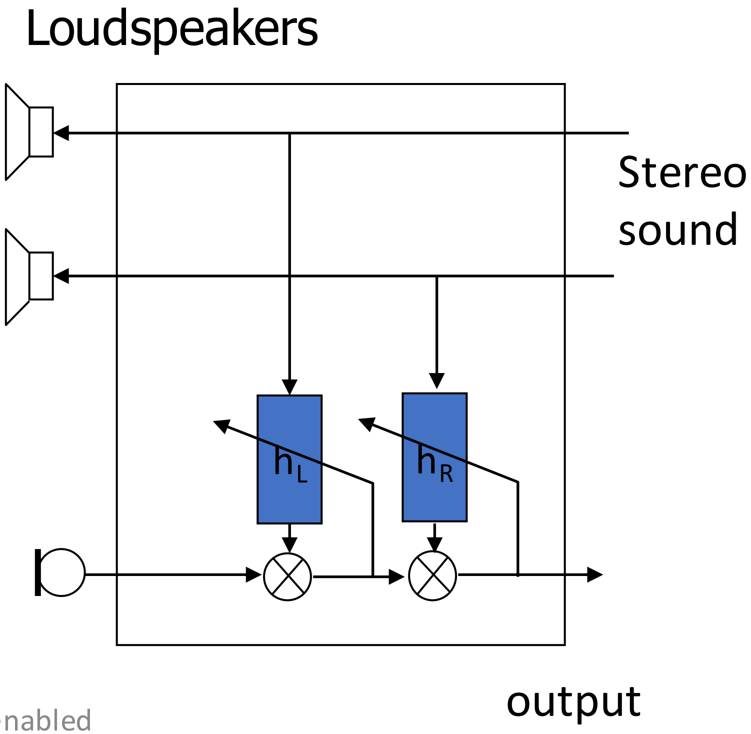
- Acoustic echo cancellation (AEC): $\hat{H}_k^{(n+1)} = \hat{H}_k^{(n)} - \mu \frac{\Re_k^{(n)} X_k^{(n)}}{\left|X_k^{(n)}\right|^2}$
- Acoustic echo suppression (AES)
- Mono AEC – part of every speakerphone



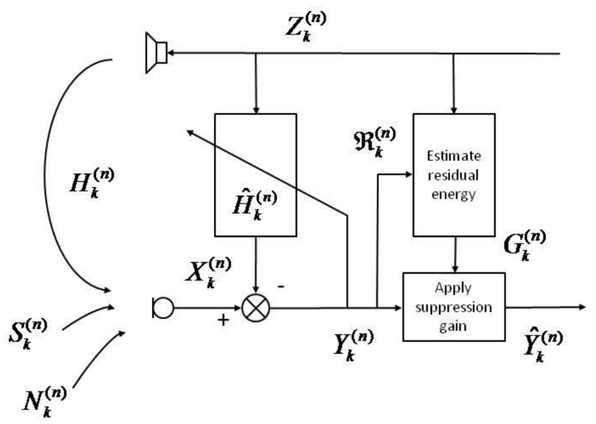
Acoustic echo reduction systems

- Acoustic echo cancellation (AEC): $\hat{H}_k^{(n+1)} = \hat{H}_k^{(n)} - \mu \frac{\Re_k^{(n)} X_k^{(n)}}{\left|X_k^{(n)}\right|^2}$
- Acoustic echo suppression (AES)
- Mono AEC – part of every speakerphone
- Stereo AEC: non-uniqueness problem

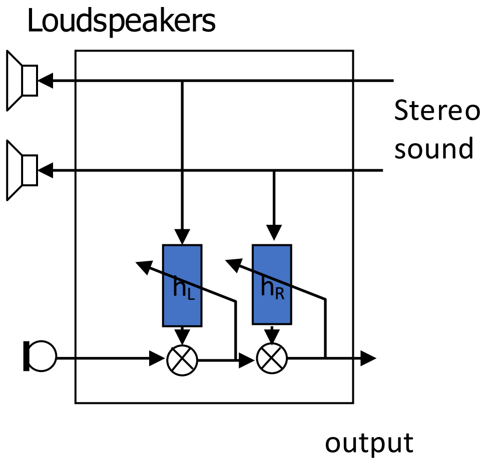


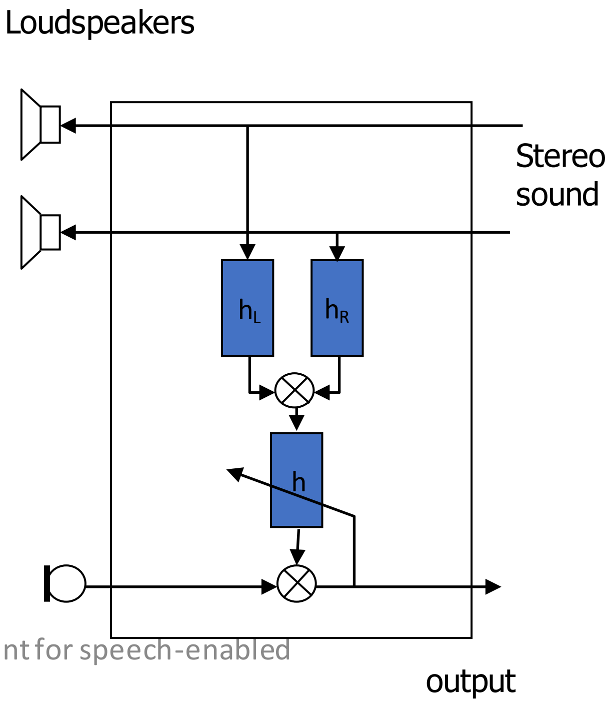


Acoustic echo reduction systems



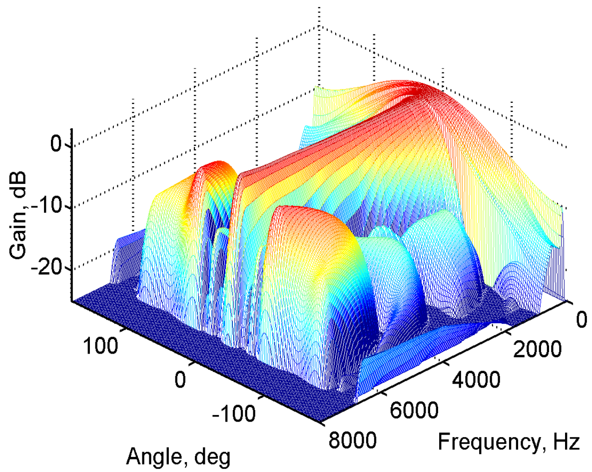
- Acoustic echo cancellation (AEC): $\hat{H}_k^{(n+1)} = \hat{H}_k^{(n)} - \mu \frac{\Re_k^{(n)} X_k^{(n)}}{\left|X_k^{(n)}\right|^2}$
- Acoustic echo suppression (AES)
- Mono AEC – part of every speakerphone
- Stereo AEC: non-uniqueness problem
- Stereo and surround sound AEC
  - Estimate impulse responses
  - Reduces the dimensionality
  - Always one solution, close to optimal

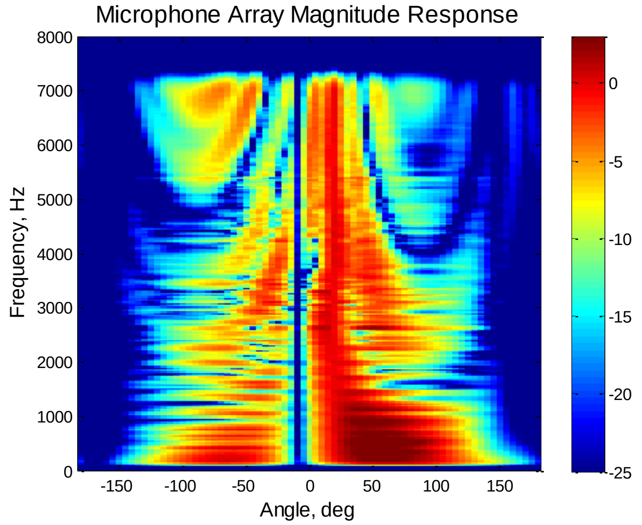




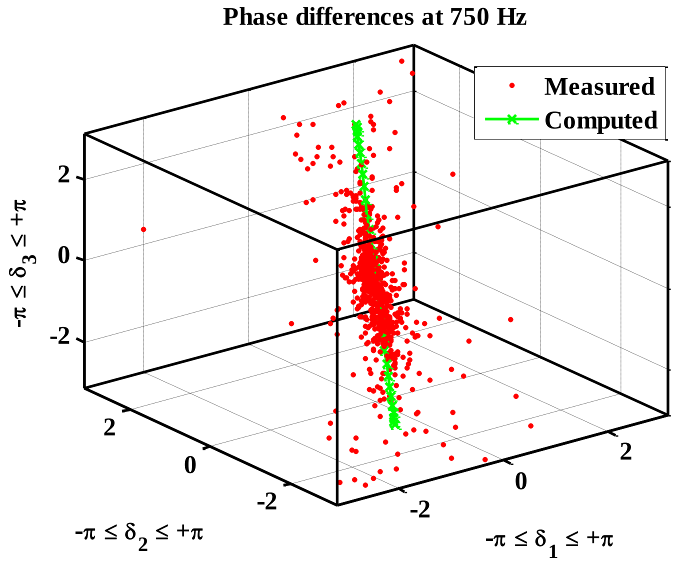
Beamforming

- Beamforming: $Y^{(n)}(k) = \mathbf{W}(k)\mathbf{X}^{(n)}(k)$
- Time invariant beamformer
- Adaptive beamformer
  - On the fly computation of the weights
  - Higher CPU requirements
  - Does null-steering
- MVDR beamformer
  - $\mathbf{W}_{MVDR}(f) = \dfrac{\mathbf{D}_c^H(f)\mathbf{\Phi}_{NN}^{-1}(f)}{\mathbf{D}_c^H(f)\mathbf{\Phi}_{NN}^{-1}(f)\mathbf{D}_c(f)}$
- Affine projection beamformer
- Other adaptive beamformers exist

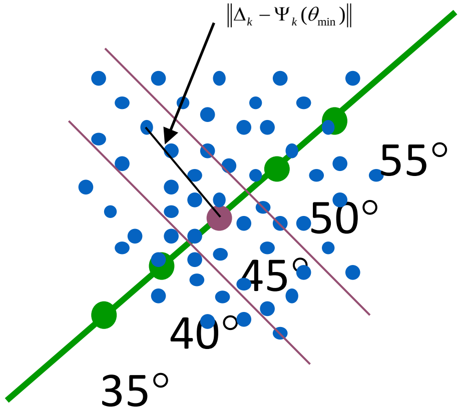




Spatial probability estimation



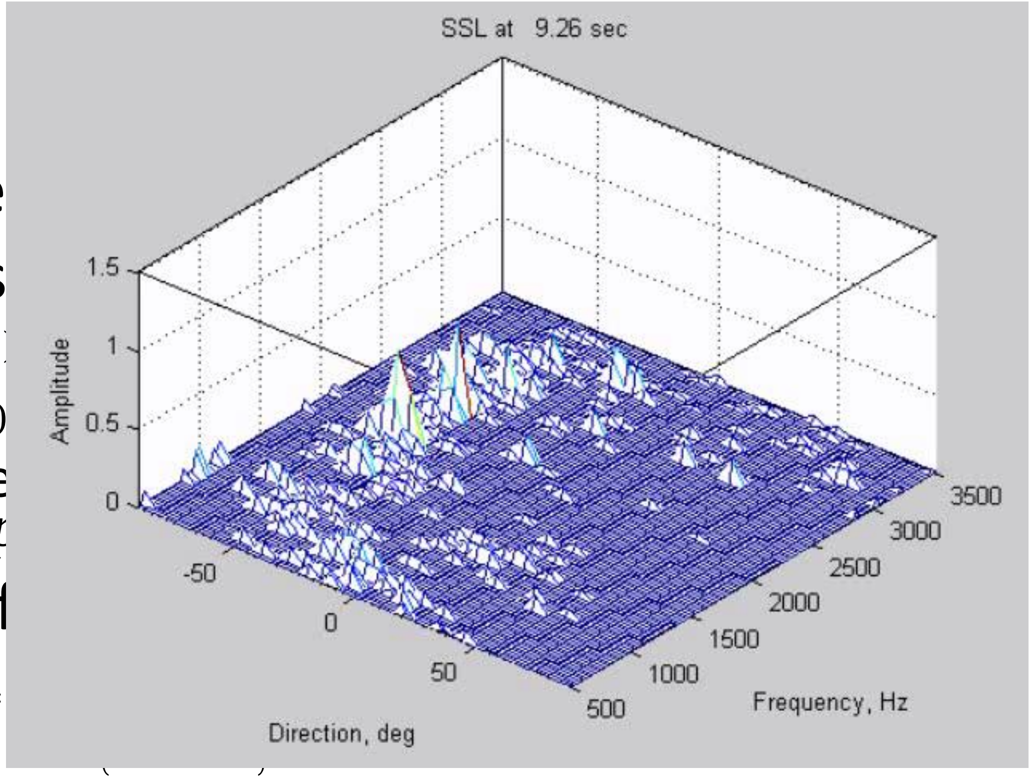
- Estimates the probability of sound source presence for each direction $p_n(\theta)$
- Instantaneous Direction Of Arrival (IDOA)[1]
  - $\Delta(f) \triangleq [\delta_1(f), \delta_2(f), \ldots, \delta_{M-1}(f)]$
  - where $\delta_{j-1}(f) = \arg(X_1(f)) - \arg(X_j(f))$
  - Compute the variation $\sigma_n(\theta)$ and the probability distribution $p_n(\theta)$
- Relative Transfer Function (RTF)[2]
  - RTF: $\hat{B}_{m,1}(k,n) = \dfrac{\mathrm{E}\{Y_m(k,n)Y_1^*(k,n)\}}{\mathrm{E}\{|Y_1(k,n)|^2\}}$
  - Distance measure: $\Delta = \cos\langle \mathbf{b}_\theta(k), \hat{\mathbf{b}}(k)\rangle$
  - $p_n(\theta)$ derived per PDFs



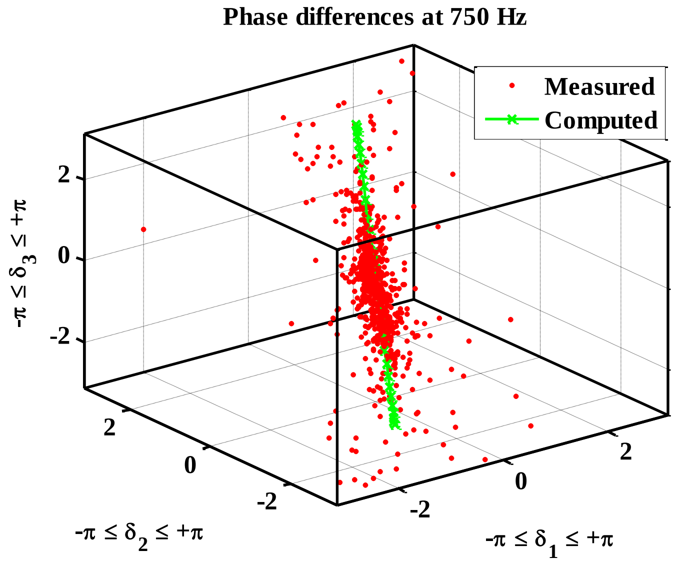
Sound source at 45° noise

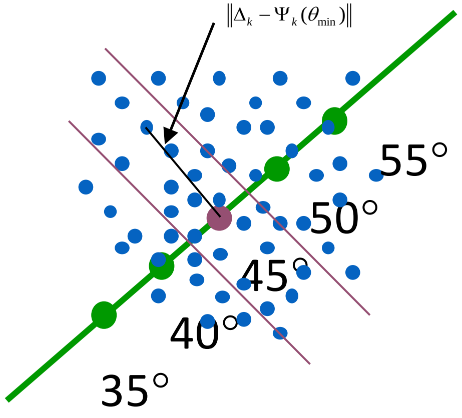
[1] I. Tashev, A. Acero, "Microphone Array Post-Processor Using Instantaneous Direction of Arrival", IWAENC 2006
[2] S. Braun, I. Tashev, "Directional interference suppression using a spatial relative transfer function feature", ICASSP 2019

Spatial probability estimation



- Estimates the
  presence for e
- Instantaneous
  - $\Delta(f) \triangleq [\delta_1(f), \delta_2(f$
  - where $\delta_{j-1}(f)$
  - Compute the
    distribution p
- Relative Transf
  - RTF: $\hat{B}_{m,1}(k,n) =$
  - Distance measure: $\Delta = \cos\left\langle \mathbf{b}_\theta(k), \hat{\mathbf{b}}(k) \right\rangle$
  - $p_n(\theta)$ derived per PDFs





Sound source at 45° noise

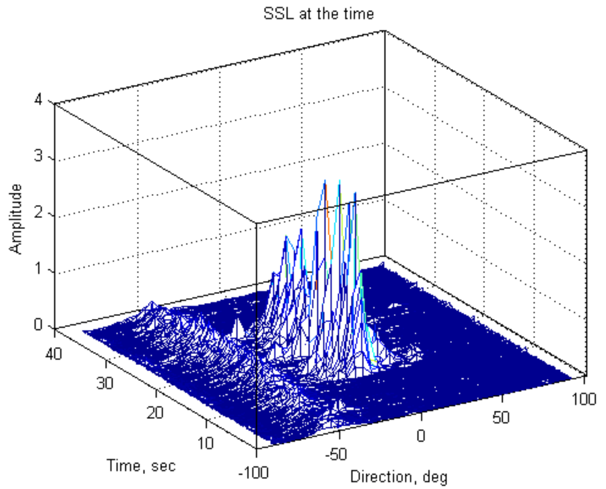
[1] I. Tashev, A. Acero, "Microphone Array Post-Processor Using Instantaneous Direction of Arrival", IWAENC 2006

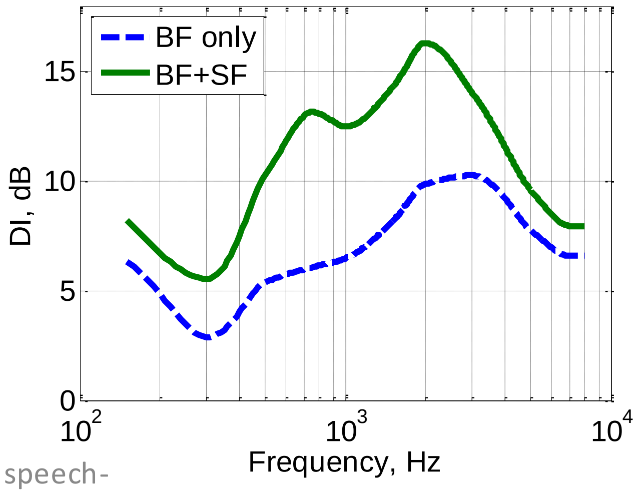
[2] S. Braun, I. Tashev, "Directional interference suppression using a spatial relative transfer function feature", ICASSP 2019

Sound source localization and spatial filtering

- Given $p_n(\theta)$ for the current frame: estimate where the sound source is
  - Find maxima
  - Cluster and average
- Given $p_n(\theta,k)$ for the current frame: estimate suppression gain
  - $\Delta\theta = 3.0\ \sigma(\theta_0)$
  - Smooth and apply

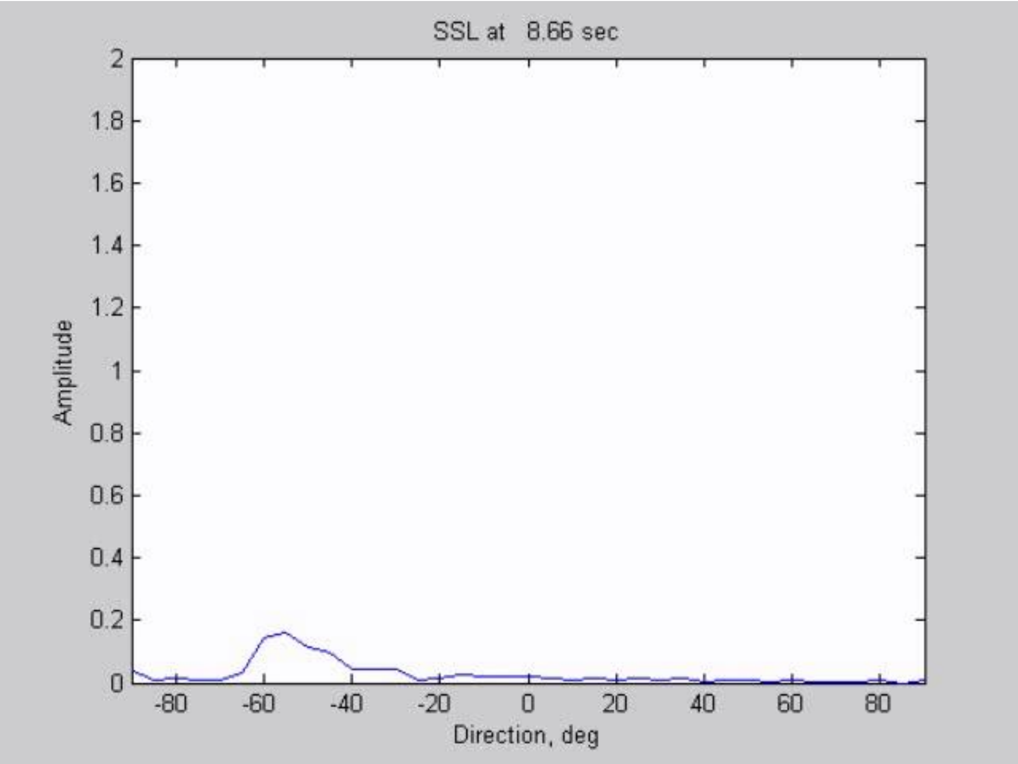
$$G_k^{(n)} = \frac{\int_{\theta_0-\Delta\theta}^{\theta_0+\Delta\theta} p(\theta)\,d\theta}{\int_{-\pi}^{+\pi} p(\theta)\,d\theta}$$



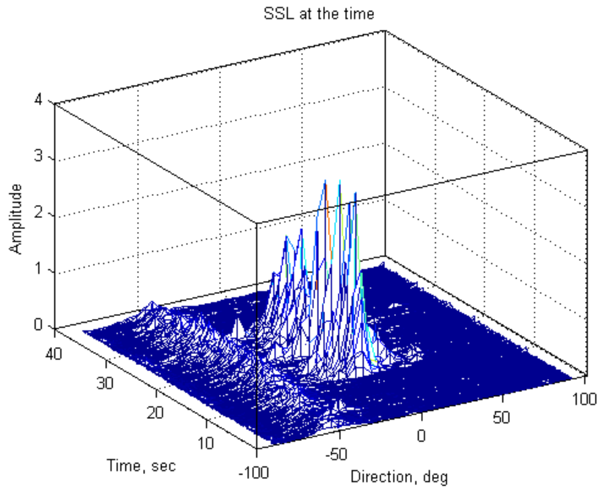


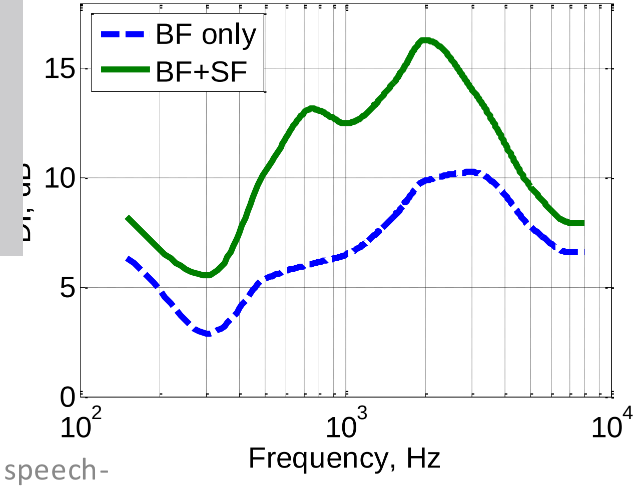
Sound source localization and spatial filtering

- Given $p_n($
  estimate
  - Find m
  - Cluster
- Given $p_n($
  estimate
  - $\Delta\theta = 3$
  - Smoot



$\int_{-\pi} p(\theta)d\theta$

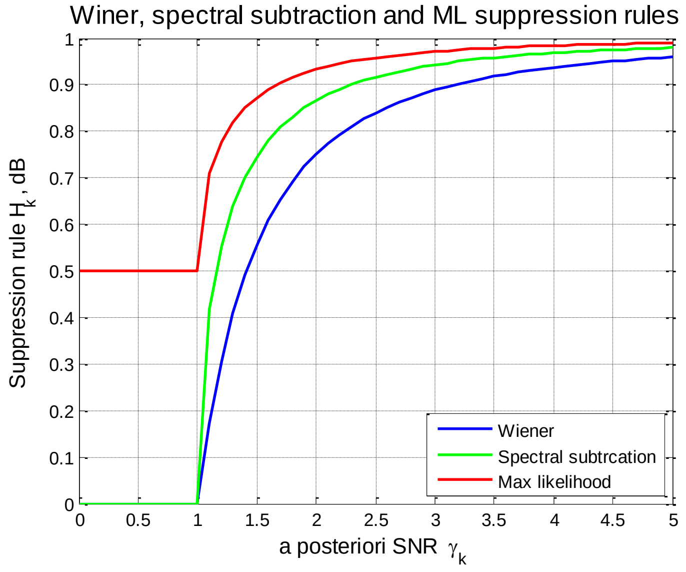




Noise suppression: Gain-based processing

- Given signal $x_n(t)$ and noise $d_n(t)$ mixed in $y_n(t)$
- Observed in frequency domain, n-th frame, k-th frequency bin: $Y_k = X_k + D_k$
- Noise suppression:
  - $\tilde{X}_k = \left(G_k \left|Y_k\right|\right)\dfrac{Y_k}{\left|Y_k\right|} = G_k.Y_k$
  - G_k – time varying, non-negative, real value gain (or suppression rule)
  - The estimator keeps the same phase as Y_k: under Gaussian assumptions the best phase estimator is observed phase
- The goal of noise suppression is for each frame to estimate G_k vector optimal in certain way

Noise suppression: Suppression rules



- Prior and posterior SNRs:

$$\xi_k \triangleq \frac{\lambda_s(k)}{\lambda_d(k)}, \gamma_k \triangleq \frac{|X_k|^2}{\lambda_d(k)}$$

$$\lambda_d(k) \triangleq E\left\{|D_k|^2\right\} \qquad \lambda_s(k) \triangleq E\left\{|S_k|^2\right\}$$

- MMSE, Wiener (1947)

$$G_k = \frac{\lambda_s(k)}{\lambda_s(k)+\lambda_d(k)} = \frac{\xi_k}{1+\xi_k}$$

- Spectral subtraction, Boll (1975): $G_k = \sqrt{\frac{\xi_k}{1+\xi_k}}$
- Maximum Likelihood, McAulay&Malpass (1981):

$$G_k = \frac{1}{2} + \frac{1}{2}\sqrt{\frac{\xi_k}{1+\xi_k}}$$

Noise suppression: Suppression rules (2)

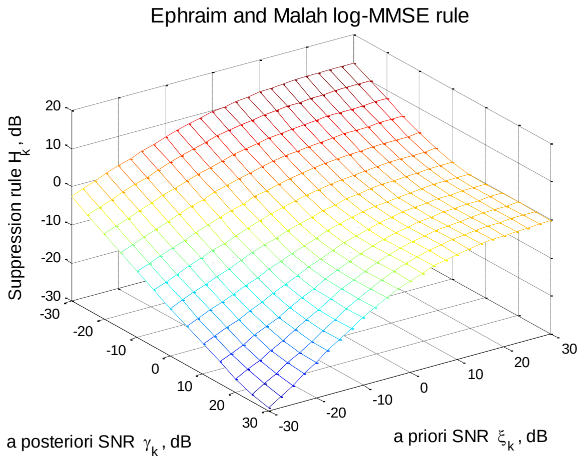
- ST-MMSE, Ephraim&Malah (1984):

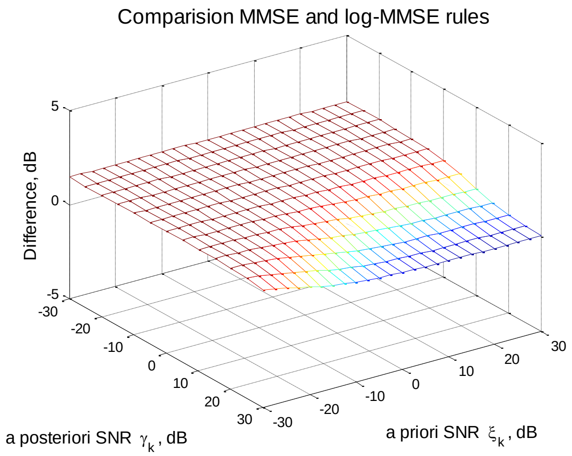
$$G_k = \frac{\sqrt{\pi \nu_k}}{2\gamma_k}\left[(1+\nu_k)\,\mathrm{I}_0\!\left(\frac{\nu_k}{2}\right) + \nu_k\,\mathrm{I}_1\!\left(\frac{\nu_k}{2}\right)\right]\exp\!\left(\frac{\nu_k}{2}\right) \qquad \nu(k) \triangleq \frac{\xi_k}{1+\xi_k}\gamma_k$$

- ST-logMMSE, Ephraim&Malah (1985):

$$G_k = \frac{\xi_k}{1+\xi_k}\left\{\frac{1}{2}\int_{\upsilon_k}^{\infty}\frac{\exp(-t)}{t}\,dt\right\}$$

- Efficient alternatives, Wolfe&Godsill (2001):
  - Joint Maximum A Posteriori Spectral Amplitude and Phase (JMAP SAP) Estimator
  - Maximum A Posteriori Spectral Amplitude (MAP SA) Estimator
  - MMSE Spectral Power (MMSE SP) Estimator
- Also see Tashev, Slaney, ITA 2014



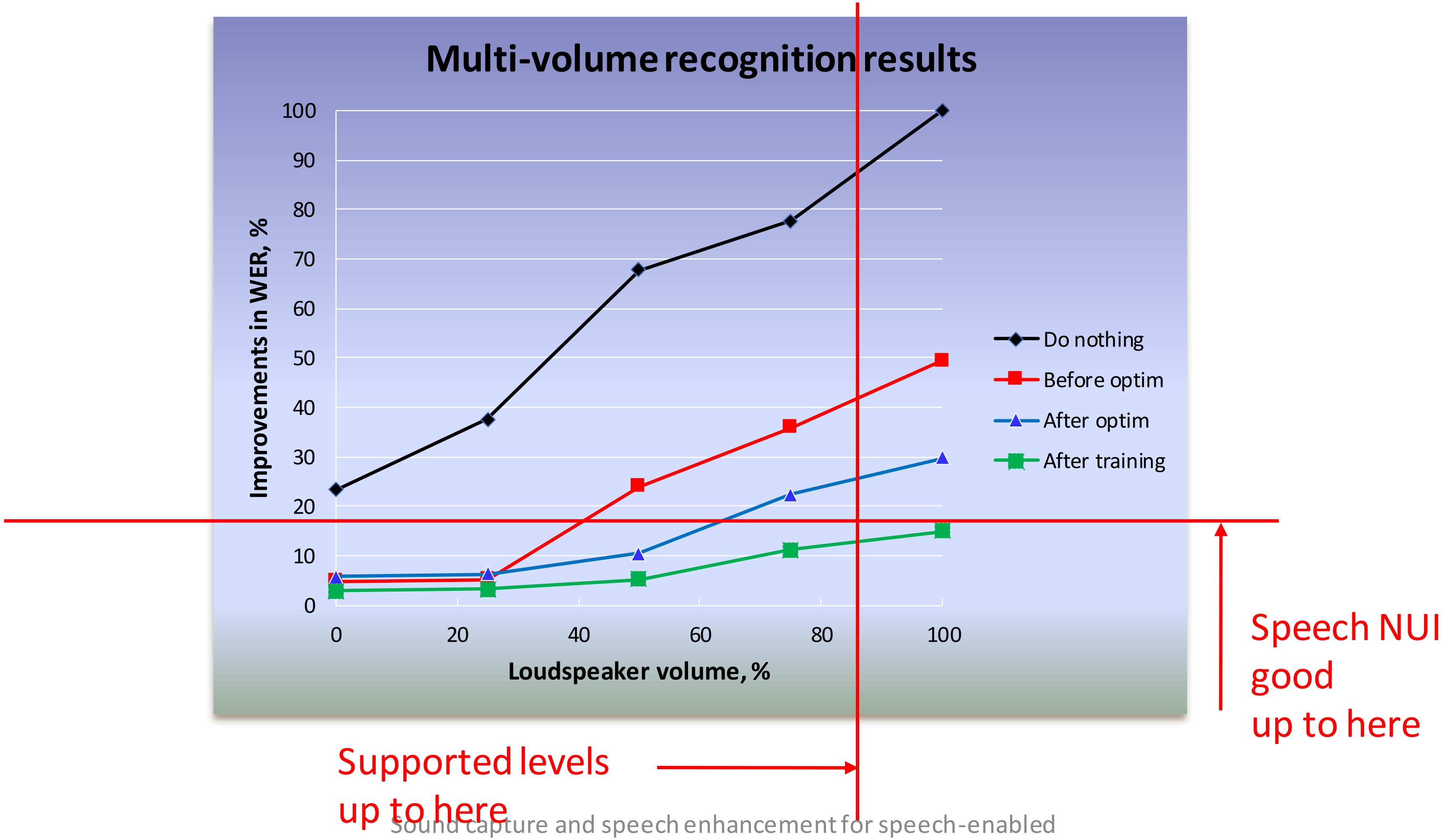


End-to-end optimization

- Mean Opinion Score (MOS), Perceptual Evaluation of Sound Quality (PESQ), Word Error Rate (WER)
- 75 parameters for optimization: time constants, limitations, etc.
- Optimization criterion:
  - $Q = PESQ + 0.05*ERLE + 0.5*WER + 0.001*SNR - 0.001*LSD - 0.01*MSE$
- Optimization algorithm
  - Gaussian minimization
- Data corpus with various distance, levels, reverberation
- Parallelized processing on computing cluster

I. Tashev, A. Lovitt, A. Acero, "Unified Framework for Single Channel Speech Enhancement", PacRim 2009

End-to-end optimization: results



Assumptions in classic speech enhancement

- Noise has Gaussian distribution
- Speech signal has Gaussian distribution
- Noise changes slower than the speech signal
- We need minimum mean squared error amplitude estimator,
    - or, minimum mean squared log-amplitude estimator,
    - or, maximum likelihood estimator, etc.
- The signals in different frequency bins are statistically independent
- The consecutive audio frames are statistically independent

Assumptions in classic speech enhancement

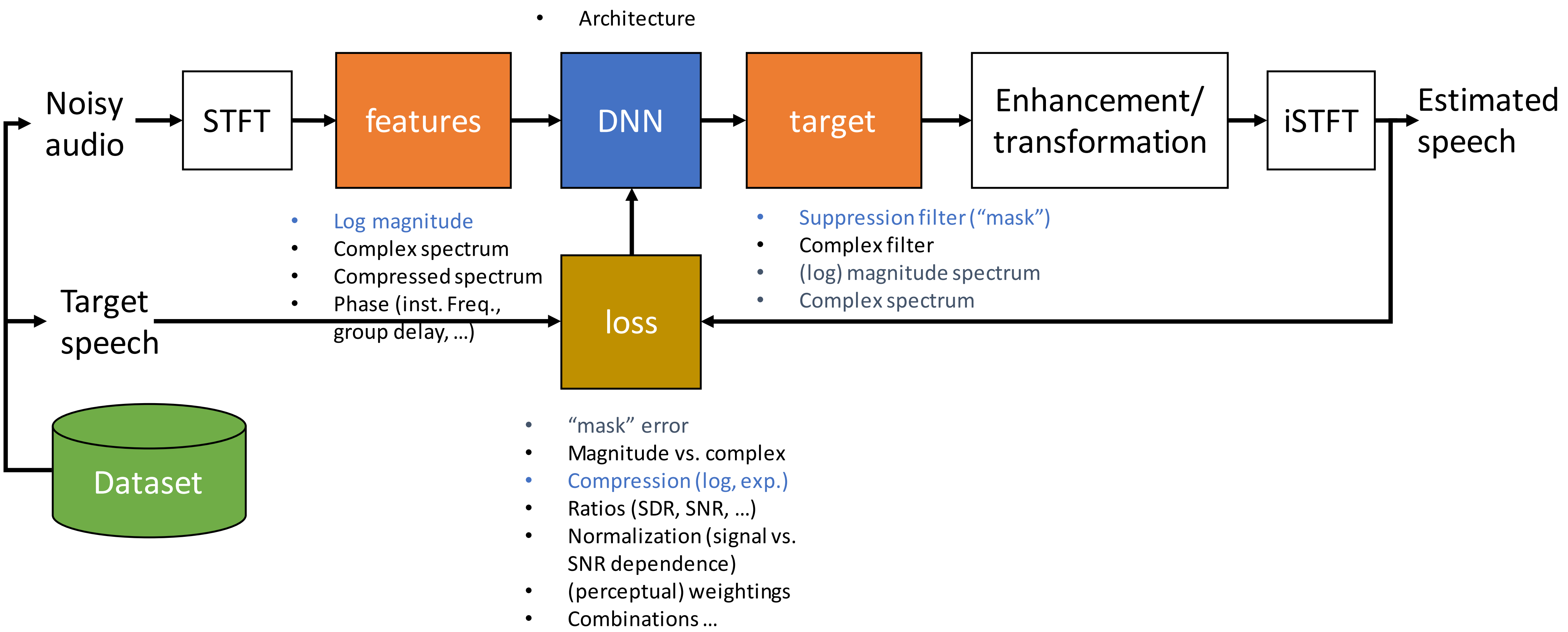
- Noise has Gaussian distribution
- Speech signal has Gaussian distribution
- Noise changes slower than the speech sign
- We need minimum mean squared error ar
  - or, minimum mean squared log-amplitude est
  - or, maximum likelihood estimator, etc.
- The signals in different frequency bins are
- The consecutive audio frames are statistically independent

Not correct!

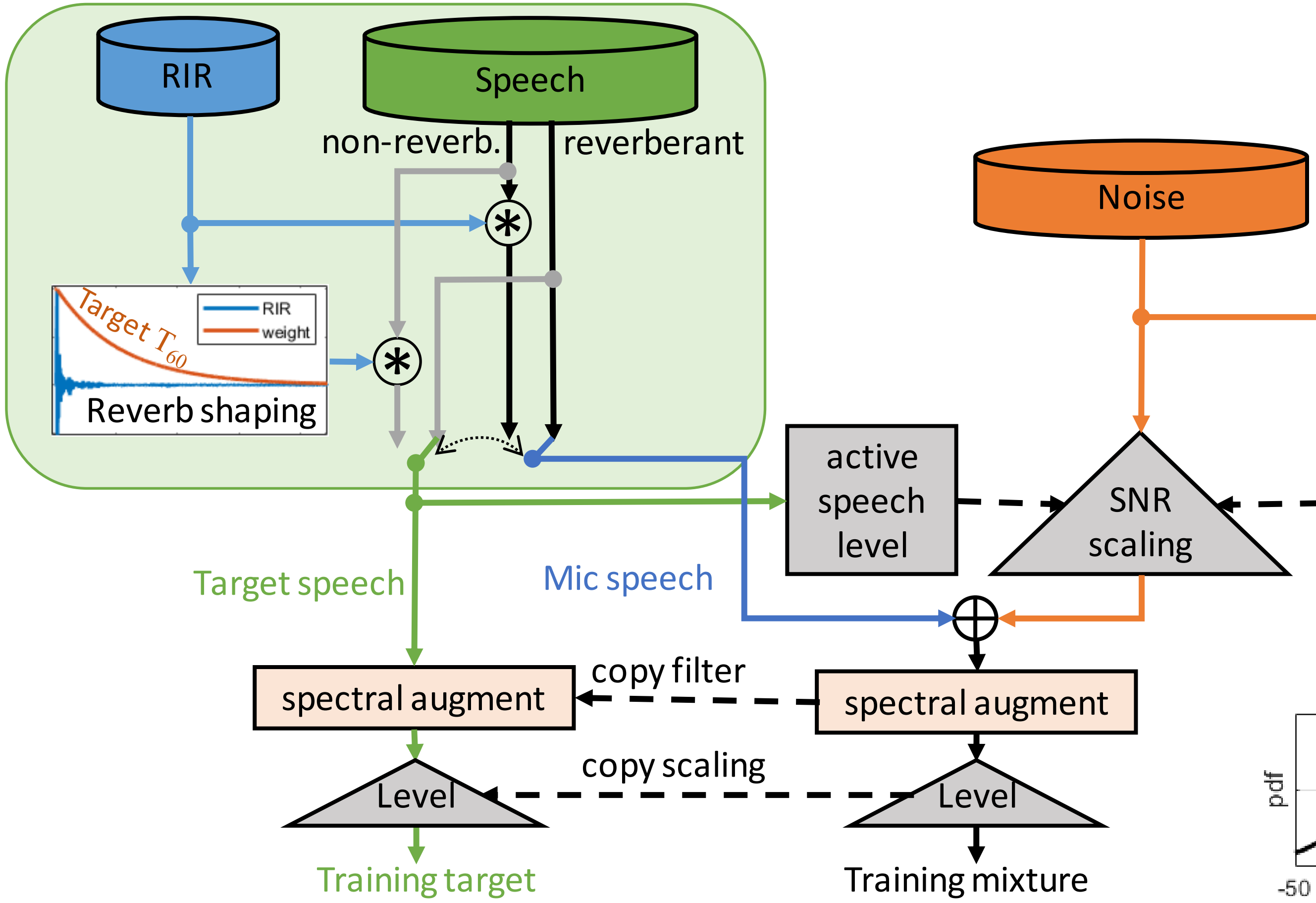
Still, worked well in RoundTable, Lync/Skype, Microsoft Auto, Kinect ☺

Application of deep learning methods in speech enhancement

Modular blocks for Speech Enhancement

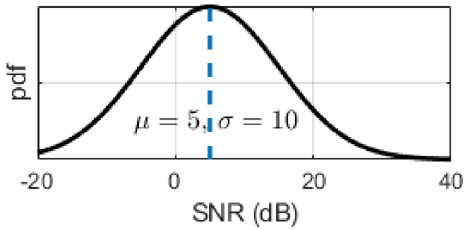


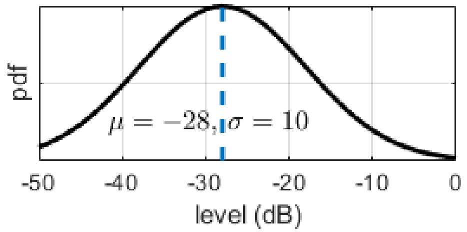
Training data generation and augmentation



- 500 h high MOS-rated speech (LibriVox)
- 8k measured + 140k simulated impulse responses
- 250 h noise from Audioset, Freesound

-> augmentation enables training networks on unique data of ~18 months



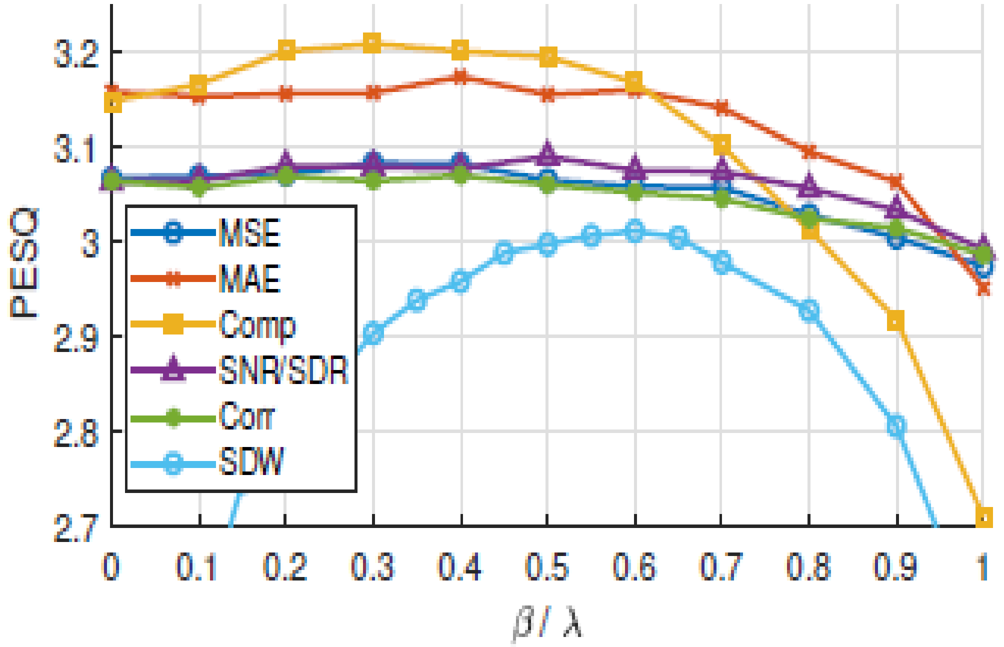


All data publicly available at https://github.com/microsoft/DNS-Challenge

S. Braun, H. Gamper, C. Reddy, I. Tashev, "Towards efficient models for real-time deep noise suppression", to appear in ICASSP 2021.

Spectral distance-based loss functions

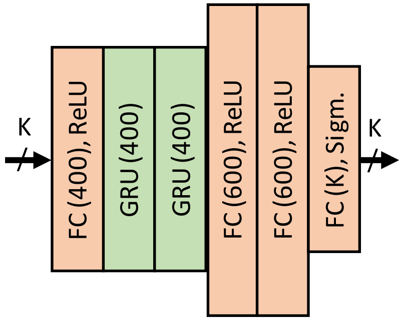
distance metric	magnitude	complex
MSE (L2)	$\|\|\mathbf{s}\| - \|\widehat{\mathbf{s}}\|\|_2^2$	$\|\mathbf{s} - \widehat{\mathbf{s}}\|_2^2$
MAE (L1)	$\|\|\mathbf{s}\| - \|\widehat{\mathbf{s}}\|\|_1$	$\|\mathbf{s} - \widehat{\mathbf{s}}\|_1$
Log spectral amplitude (LSA)	$\|\log\|\mathbf{s}\| - \log\|\widehat{\mathbf{s}}\|\|_2^2$	LSA x phase error
compressed MSE	$\|\|\mathbf{s}\|^c - \|\widehat{\mathbf{s}}\|^c\|_2^2$	$\|\|\mathbf{s}\|^c e^{j\varphi_{\mathbf{s}}} - \|\widehat{\mathbf{s}}\|^c e^{j\varphi_{\widehat{\mathbf{s}}}}\|_2^2$
Signal Ratios (SNR/SDR)	$\frac{\|\|\mathbf{s}\|\|_2^2}{\|\|\widehat{\|\mathbf{s}\|} - \|\mathbf{s}\|\|\|_2^2}$	$\frac{\|\mathbf{s}\|_2^2}{\|\widehat{\mathbf{s}} - \mathbf{s}\|_2^2}$
Correlation	$\frac{\|\mathbf{s}\|^T\|\widehat{\mathbf{s}}\|}{\|\mathbf{s}\|_2\|\widehat{\mathbf{s}}\|_2}$	$\frac{\|\mathbf{s}^H\widehat{\mathbf{s}}\|}{\|\mathbf{s}\|_2\|\widehat{\mathbf{s}}\|_2}$
Speech distortion weighted (SDW)	$\lambda\|\mathbf{g}\circ\mathbf{s} - \mathbf{s}\|_2^2 + (1-\lambda)\|\mathbf{g}\circ\mathbf{n}\|_2^2$	x



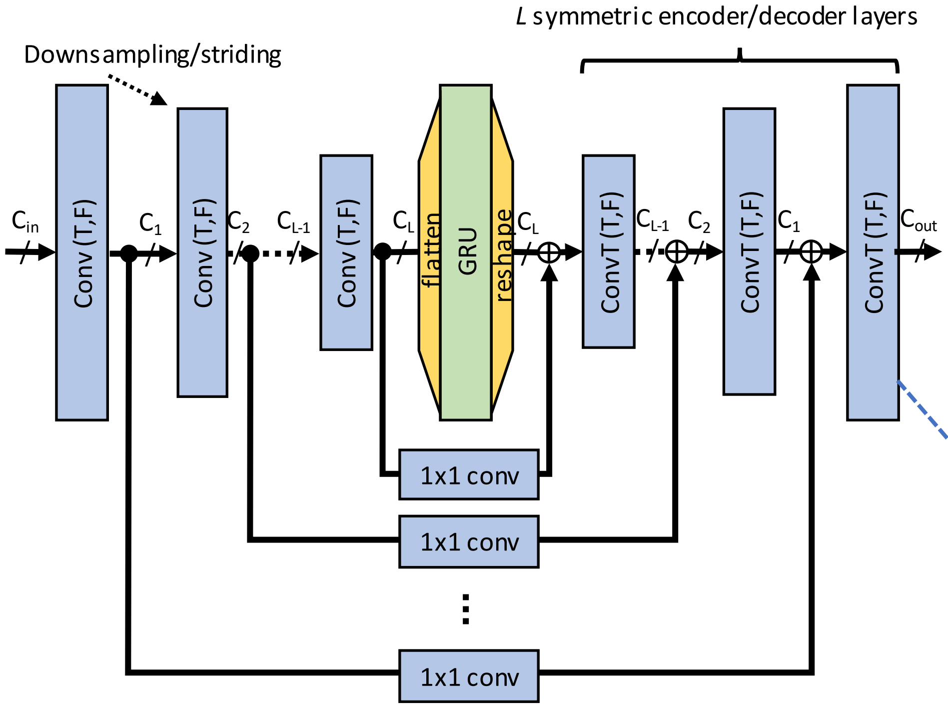
$$L = (1 - \lambda)\, L_{mag} + \lambda\, L_{complex}$$

S. Braun and I. Tashev, "A consolidated view of loss functions for supervised deep learning-based speech enhancement", arXiv:2009.12286, 2020.
Y. Xia, S. Braun, C. Reddy, R. Cutler, I. Tashev, "Weighted Speech Distortion Losses for Neural-Network-Based Real-Time Speech Enhancement", ICASSP 2020.

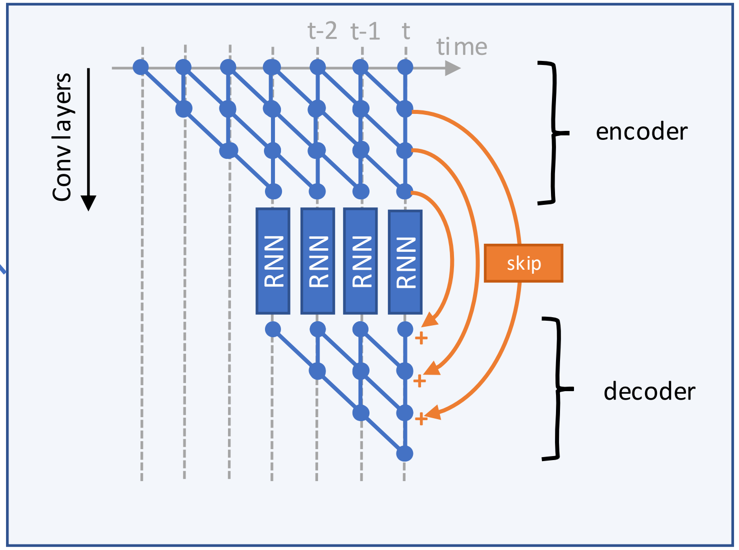
Efficient network architectures



NSnet2 [1]
(Recurrent net)



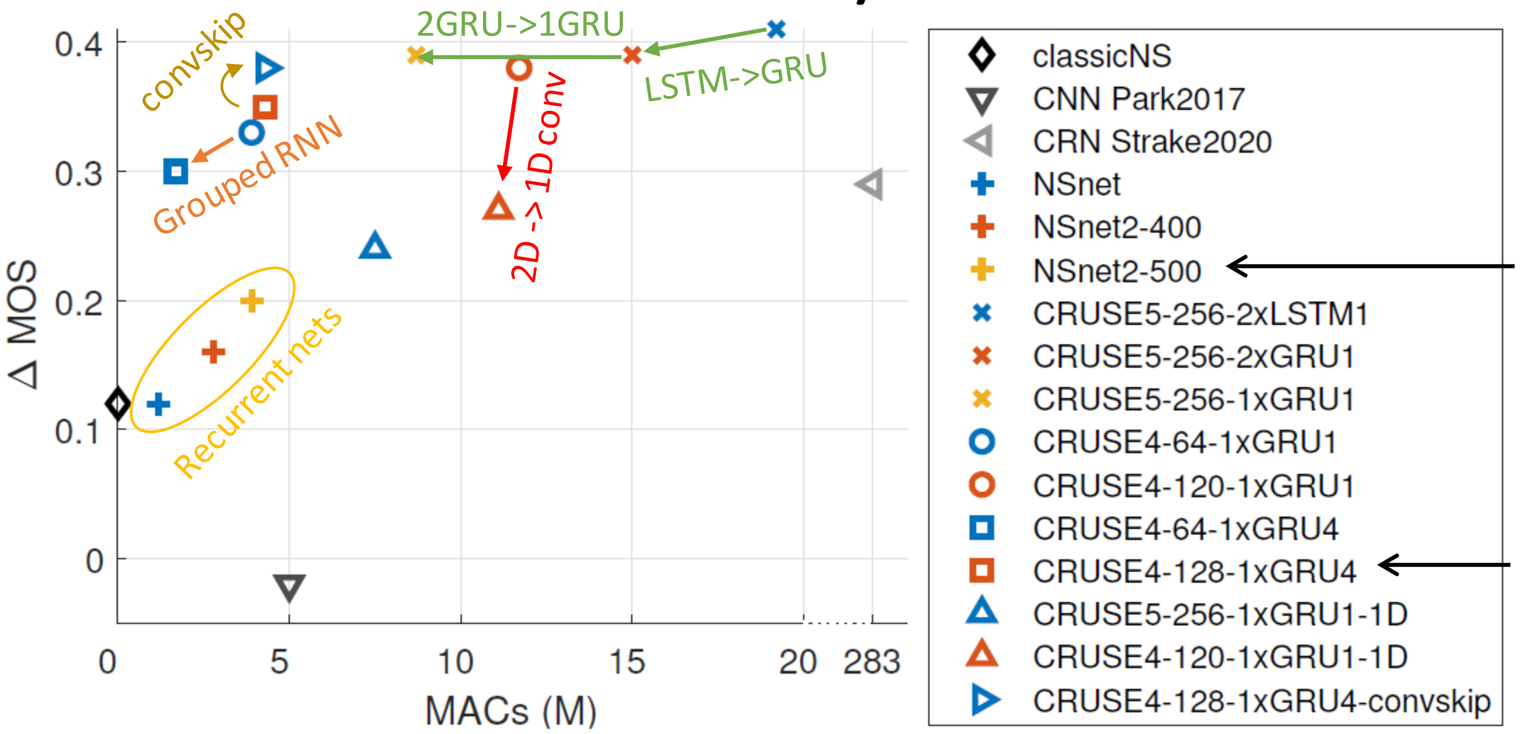
CRUSE (Convolutional Recurrent U-net for Speech Enhancement) [2]



Temporal view: causal convolutions with (T,F) kernel = (2,F)

[1] S. Braun and I. Tashev, *Data augmentation and loss normalization for deep noise suppression*, International Conference on Speech and Computer, 2020.
[2] S. Braun, H. Gamper, C. Reddy, I. Tashev, *Towards efficient models for real-time deep noise suppression*, to appear in ICASSP 2021.

Results model efficiency



≈ 15ms / sec @ 3.5 GHz CPU

≈ 30ms / sec @ 3.5 GHz CPU

model	MACs (M)	ΔMOS
no skips	4.3	0.32
add skips	4.3	0.35
add conv 1×1 skips	4.3	**0.38**
concat skips [Tan2019]	4.8	0.38

K. Tan, D. Wang, *A Convolutional Recurrent Neural Network for Real-Time Speech Enhancement*, in Proc. Interspeech, 2018.

S. R. Park, J. W. Lee, *A Fully Convolutional Neural Network for Speech Enhancement*, Proc. Interspeech, 2017.

M. Strake, et. al., Fully Convolutional Recurrent Networks for *Speech Enhancement*, in Proc. ICASSP, 2020.

2nd Deep Noise Suppression Challenge

Team	Team #	Singing			Tonal			Non-English (includes Tonal)			English			Emotional			Overall		
		MOS	DMOS	95% CI	MOS	DMOS	95% CI	MOS	DMOS	95% CI	MOS	DMOS	95% CI	MOS	DMOS	95% CI	MOS	DMOS	95% CI
Microsoft-1*		3.18	0.22	0.11	3.63	0.63	0.06	3.61	0.65	0.04	3.57	0.76	0.04	2.68	0.00	0.08	3.43	0.57	0.03
IACASlab9	24	3.14	0.17	0.11	3.44	0.44	0.06	3.50	0.53	0.04	3.49	0.69	0.04	2.92	0.25	0.08	3.38	0.53	0.03
Microsoft-2* (CRUSE)		3.00	0.03	0.12	3.53	0.53	0.06	3.53	0.57	0.04	3.52	0.72	0.04	2.76	0.08	0.08	3.38	0.52	0.03
Sogou	18	3.23	0.27	0.10	3.39	0.39	0.06	3.43	0.47	0.04	3.45	0.65	0.04	2.93	0.26	0.08	3.35	0.50	0.03
Amazon	23	3.16	0.20	0.10	3.40	0.41	0.07	3.42	0.46	0.04	3.47	0.66	0.04	2.90	0.22	0.08	3.34	0.49	0.03
Trident	14	3.01	0.05	0.10	3.35	0.35	0.07	3.40	0.44	0.04	3.42	0.62	0.04	2.96	0.28	0.07	3.32	0.46	0.03
Seoul National University-Supertone	16	3.08	0.12	0.10	3.38	0.38	0.07	3.43	0.46	0.04	3.41	0.61	0.04	2.88	0.21	0.07	3.32	0.46	0.03
UCAS	13	3.09	0.13	0.09	3.31	0.31	0.08	3.38	0.42	0.04	3.35	0.55	0.04	2.99	0.32	0.08	3.29	0.44	0.03
NPU	26	3.06	0.10	0.10	3.33	0.33	0.07	3.39	0.42	0.04	3.37	0.57	0.04	2.80	0.13	0.08	3.27	0.42	0.03
Baidu	21	2.93	(0.04)	0.10	3.33	0.33	0.07	3.39	0.42	0.04	3.31	0.51	0.04	2.69	0.01	0.08	3.22	0.36	0.03
Baseline-NSnet2		3.10	0.14	0.09	3.25	0.26	0.06	3.28	0.31	0.04	3.30	0.50	0.04	2.88	0.21	0.08	3.21	0.36	0.03
University Oldenburg	27	2.82	(0.14)	0.10	3.27	0.27	0.06	3.34	0.38	0.04	3.24	0.44	0.04	2.77	0.10	0.07	3.18	0.32	0.03
SDUT	9	2.92	(0.04)	0.10	3.16	0.16	0.06	3.21	0.25	0.04	3.20	0.40	0.04	2.65	(0.03)	0.07	3.10	0.25	0.02
Westlake University	22	2.99	0.03	0.09	3.06	0.06	0.07	3.15	0.19	0.04	3.09	0.29	0.04	2.80	0.12	0.07	3.06	0.21	0.02
TU Braunschweig	8	2.53	(0.43)	0.09	3.09	0.09	0.07	3.17	0.20	0.04	3.12	0.32	0.04	2.76	0.08	0.07	3.04	0.18	0.03
CILAB	10	2.73	(0.23)	0.09	3.06	0.06	0.06	3.05	0.08	0.04	2.90	0.10	0.03	2.63	(0.05)	0.07	2.91	0.05	0.02
Jadavpur University Innovators Lab	20	2.95	(0.02)	0.09	3.01	0.02	0.05	3.00	0.03	0.03	2.86	0.06	0.03	2.68	0.01	0.07	2.90	0.04	0.02
University of East London	4	2.66	(0.30)	0.10	2.89	(0.11)	0.07	2.94	(0.03)	0.04	2.90	0.10	0.03	2.65	(0.03)	0.07	2.86	0.00	0.02
Noisy		2.96	-	0.08	3.00	-	0.05	2.96	-	0.03	2.80	-	0.03	2.67	-	0.07	2.86	-	0.02
CASIA	11	2.38	(0.58)	0.09	2.58	(0.42)	0.06	2.61	(0.35)	0.04	2.56	(0.25)	0.04	2.43	(0.25)	0.07	2.55	(0.31)	0.02

*- The models from Microsoft will be ignored while picking the final winners

Demo recording

Conclusions

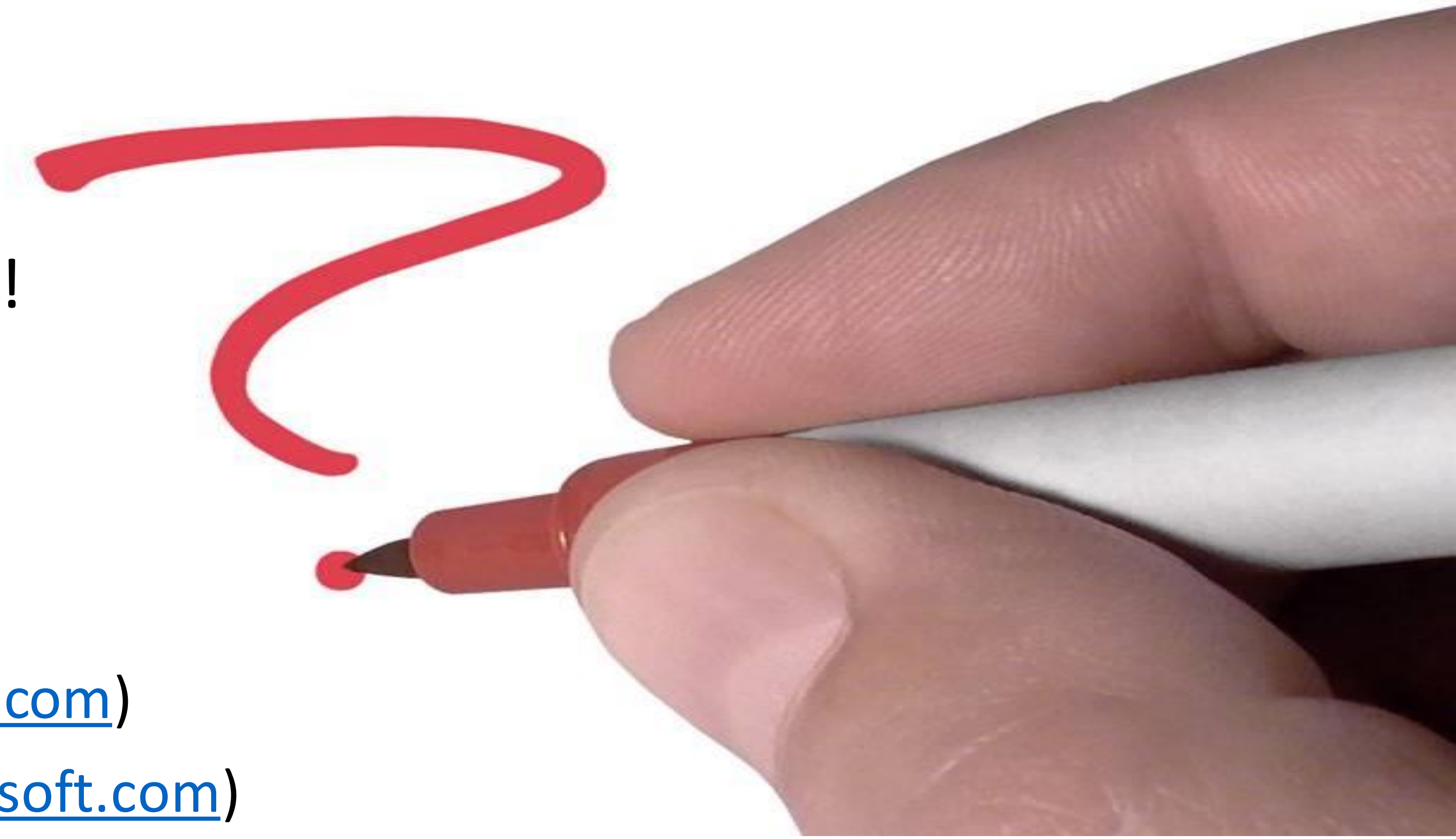
Conclusions

- Most of the modern devices include speech input for communication and speech recognition
- They operate in challenging environments: reverberation, echo, noise
- Using multiple microphones provides opportunities for better improvements for both near and far field capture
- Statistical signal processing:
    - Computationally and memory inexpensive
    - Pretty much saturated in terms of improvements

Conclusions (2)

- DNN-based speech enhancement without look-ahead in real-time is possible with smaller computational effort
- Critical for the success:
  - Dataset: defines the “signal model”. Data augmentation!
  - Loss function allows model improvement at zero inference cost. Our current best supervised loss is **signal-based**, including magnitude and phase, **compression** (human perception related), and **level-normalized** for smoother training.
  - Neural network architecture
    - Model size scales the quality: we found direct influence of model width and memory capacity on enhancement performance.
    - Recurrent networks seem more efficient for very small models, adding convolutional encoders achieve better quality at increased cost.

Finally



Thank you for your attention!

Questions?

Ivan Tashev (ivantash@microsoft.com)

Sebastian Braun (sebraun@microsoft.com)

Audio and Acoustics Research Group

https://www.microsoft.com/en-us/research/group/audio-and-acoustics-research-group/