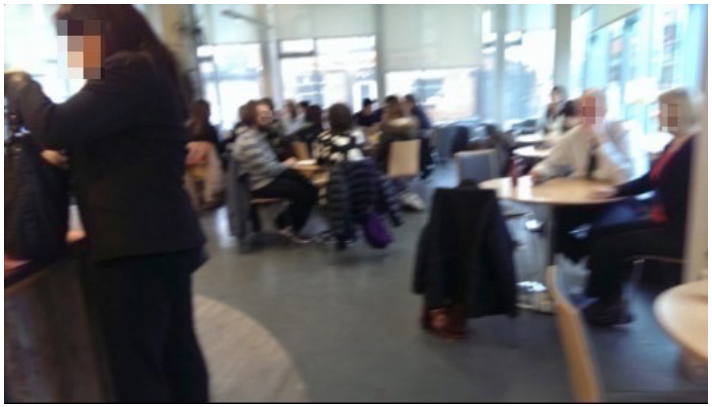
When mismatched training data outperform matched data

Emmanuel Vincent

WHEN MISMATCHED TRAINING DATA OUTPERFORM MATCHED DATA

Emmanuel Vincent
Inria Nancy – Grand Est, France

Noise-robust speech recognition



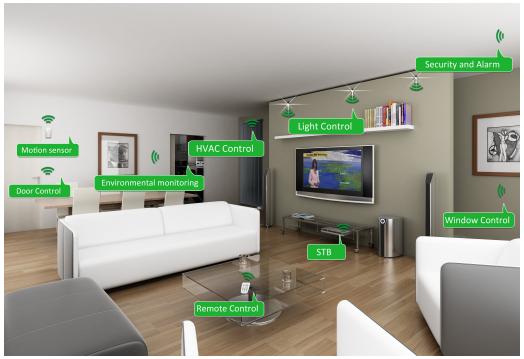
Sitting in a cafe (**CAF**)

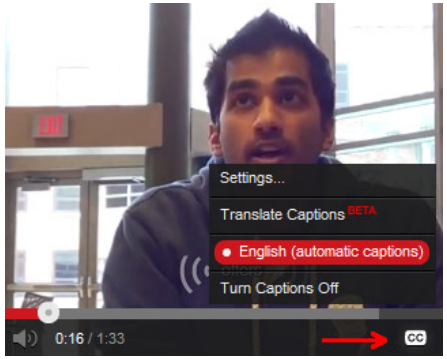
Standing at a street junction (**STR**)

Traveling on a bus (**BUS**)

In a pedestrian area (**PED**)

Some applications





Tasks: speech, speaker, or language recognition, paralinguistics. . .

Challenges and approach

Challenges:

- reverberation
- multiple, nonstationary noise sources
- overlapping speech
- moving sources and/or microphones.

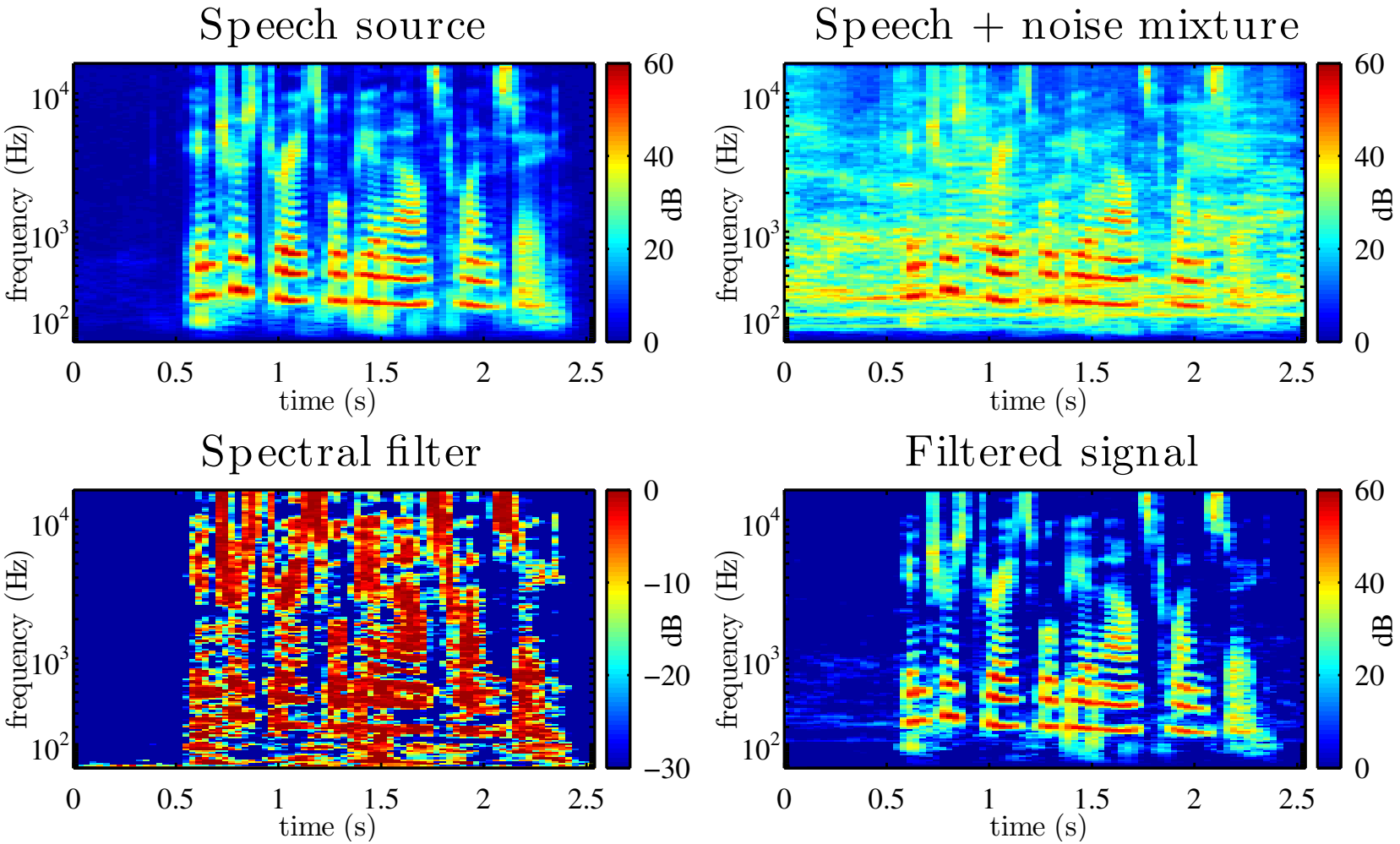
General approach:

- single- or multichannel speech enhancement/separation
- combined with better features and acoustic model.

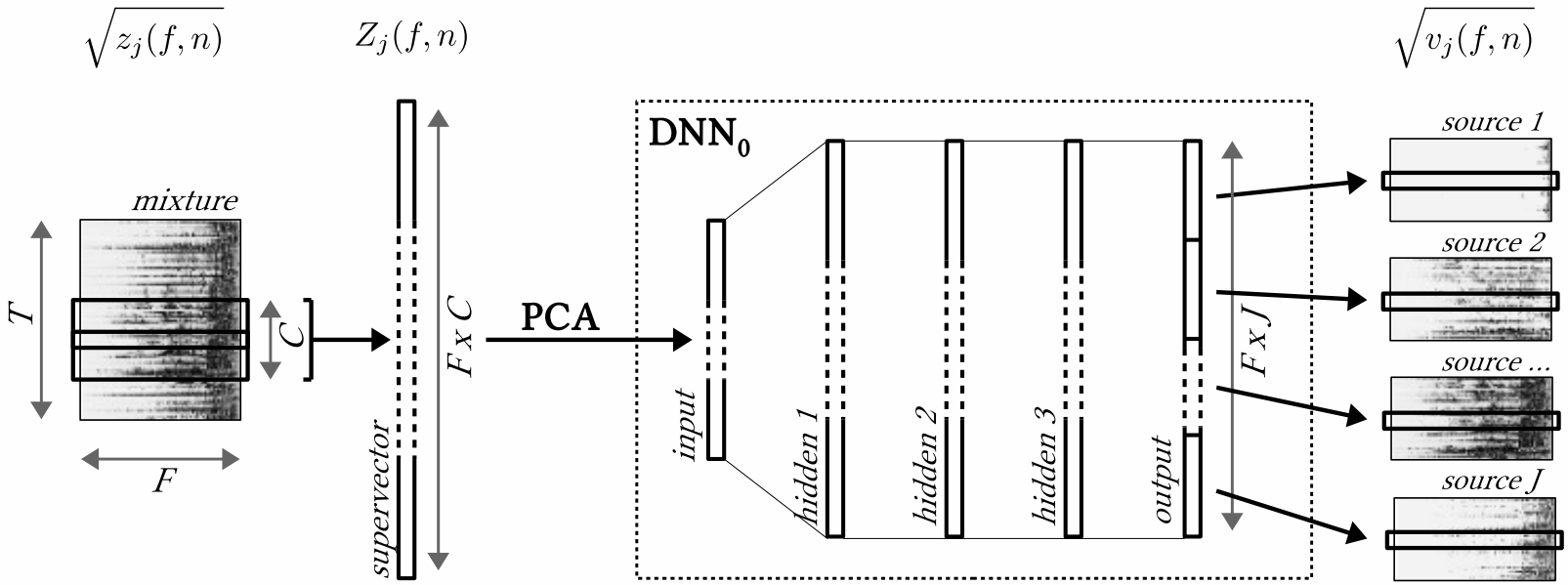
DEEP LEARNING BASED SPEECH ENHANCEMENT

Single-channel enhancement/separation

Spectral filtering achieved via time-frequency masking.



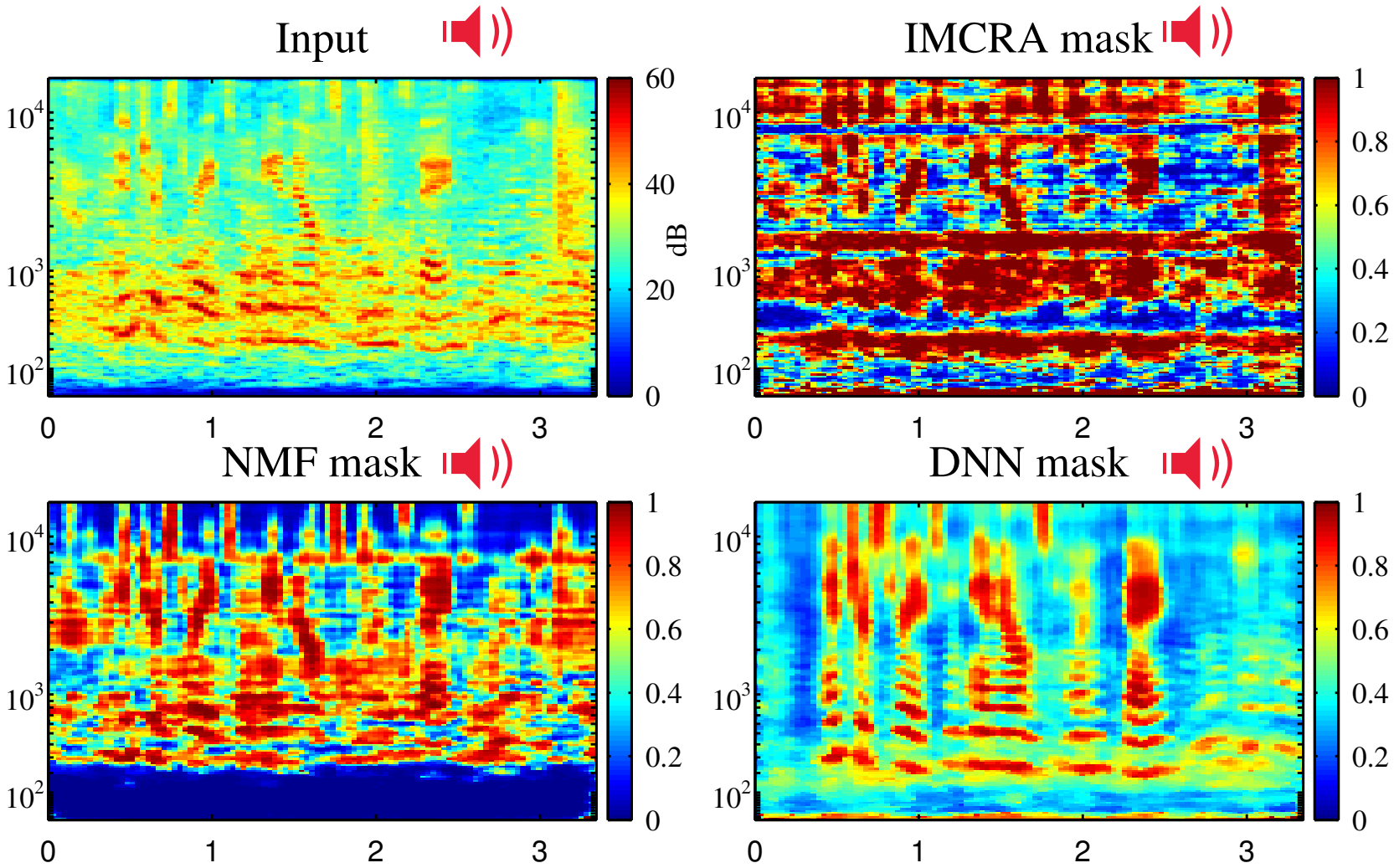
Using DNNs for single-channel separation



- Training data: simulated mixtures of speech and noise
- Test data: time-frequency mask computed as

$$\frac{v_{\text{speech}}(f,n)}{v_{\text{speech}}(f,n) + v_{\text{noise}}(f,n)}$$

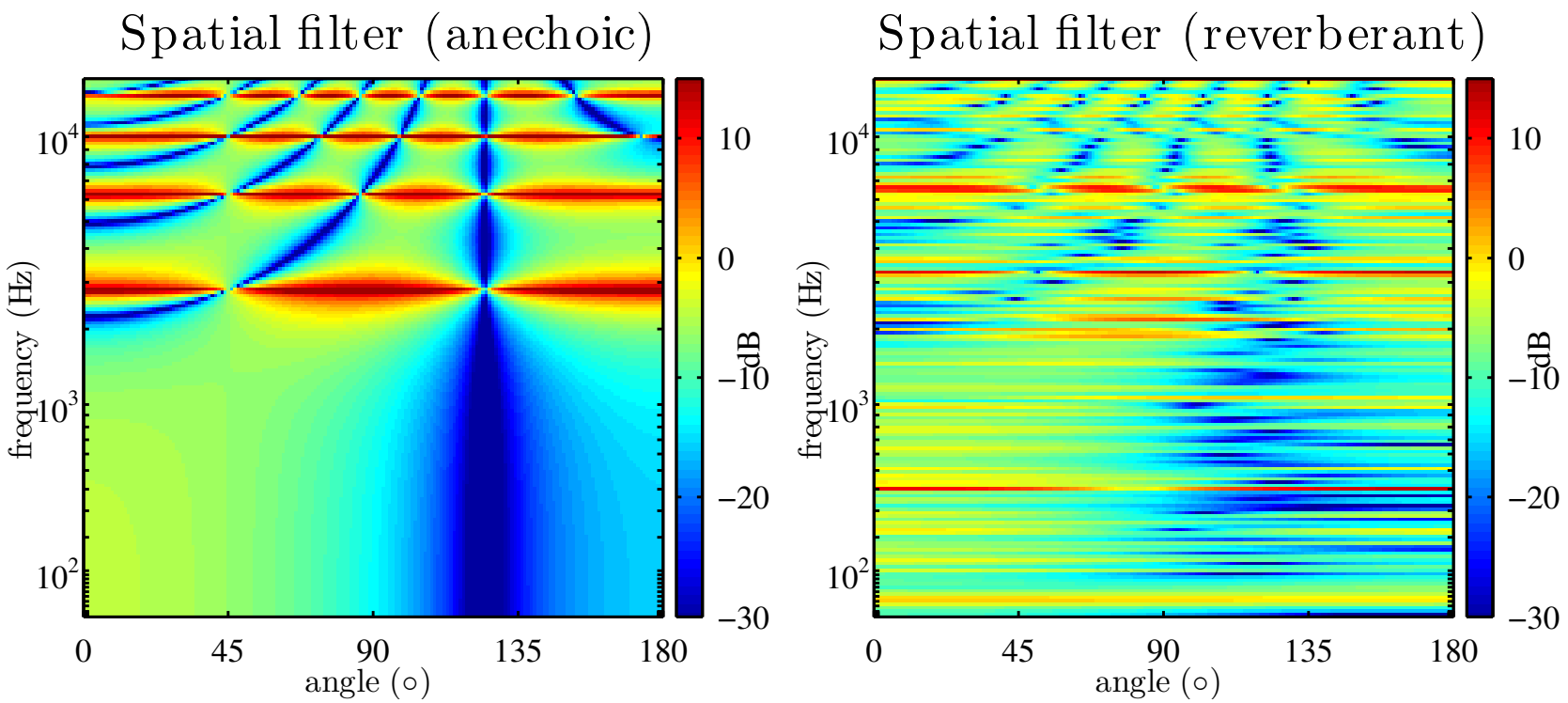
Results



CHiME-3: speech recorded in a café. Single-channel enhancement by Wiener mask. NMF training: noise context. DNN training: bus + café + pedestrian area + street.

Multichannel enhancement/separation

Combination of spatial and spectral filtering.



Using DNNs for multichannel separation in an EM fashion

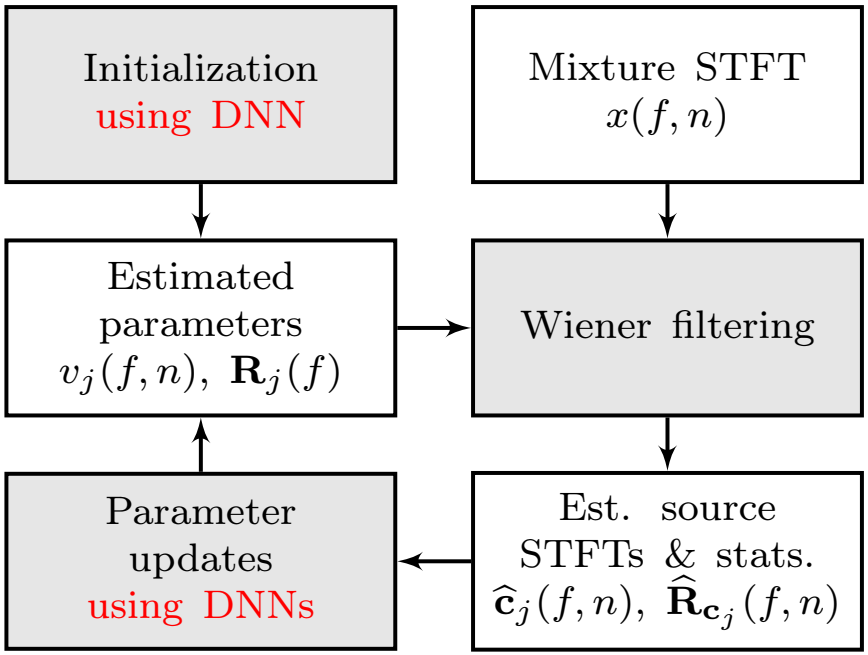
$$\mathbf{x}(f,n) = \sum_j \mathbf{c}_j(f,n) \qquad \mathbf{c}_j(f,n) \sim \mathcal{N}(\mathbf{0}, v_j(f,n)\,\mathbf{R}_j(f))$$

noisy speech

spatial image of j-th source

power spectrum

spatial covariance matrix



Parameter updates:

- Update spatial covariance matrix:

$$\mathbf{R}_j(f) = \frac{1}{N}\sum_{n=1}^{N} \frac{1}{v_j(f,n)} \widehat{\mathbf{R}}_{\mathbf{c}_j}(f,n)$$

- Compute unconstrained spectrogram:

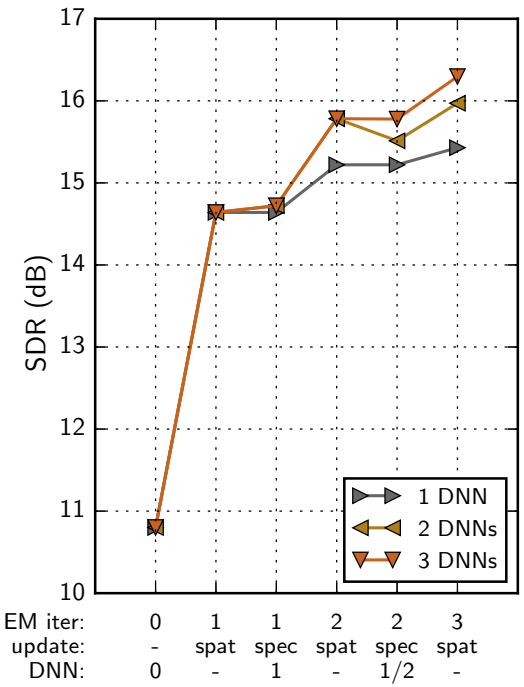
$$z_j(f,n) = \frac{1}{I}\mathrm{tr}\left(\mathbf{R}_j(f)^{-1}\widehat{\mathbf{R}}_{\mathbf{c}_j}(f,n)\right)$$

- Update spectrogram given **DNN**:

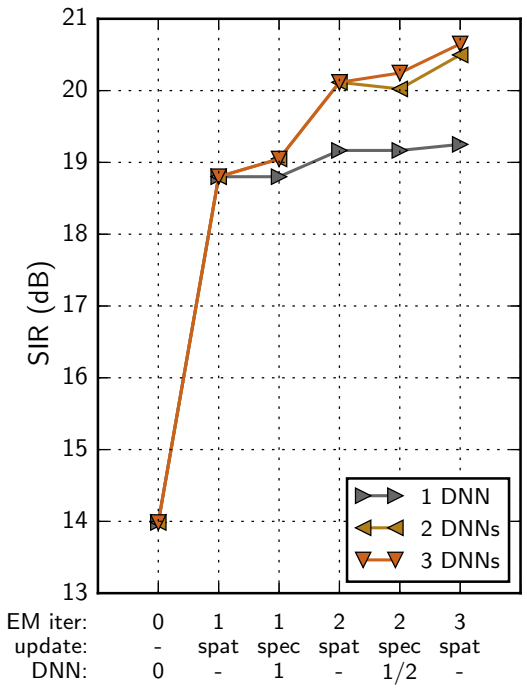
$$v_j(f,n) \leftarrow \mathrm{DNN}\left(z_j(f,n)\right)$$

Performance across iterations

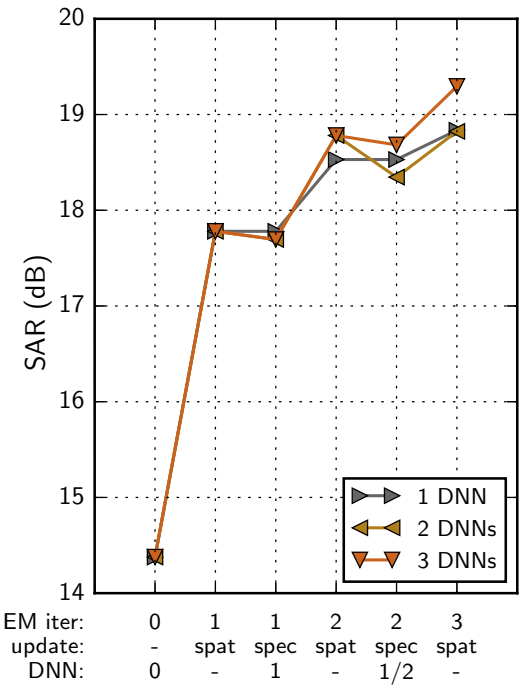
Spatial filtering translates into improved DNN inputs, which result in improved outputs.



SDR (⇒ *distortion*)



SIR (⇒*interference*)



SAR (⇒*artifact*)

Results

Evaluation in terms of word error rate (WER) with 6 mics.

Noisy		WER baseline
Single-channel DNN		no WER reduc.
Delay-and-sum		21% rel. WER reduc.
DNN post-filter		20% rel. WER reduc.
Multichannel DNN		39% rel. WER reduc.

CHiME-3: speech recorded in a bus/café. Single DNN iteration, no post-processing.

DEEP LEARNING BASED SPEECH RECOGNITION

DNN-based acoustic modeling

DNNs are also the state-of-the-art for acoustic modeling of speech

- input features: MFCCs, logmel, waveform
- outputs: phonetic classes, characters

Full speech recognition system

Robust automatic speech recognition (ASR) systems typically involve:

- multichannel speech enhancement
- robust features
  - auditory-inspired
  - feature adaptation (fMLLR)
- robust acoustic modeling
  - CNN, BLSTM
  - training data augmentation (more SNRs, enhanced data)
  - model adaptation
- system combination

CHiME-3/CHiME-4 benchmark

CHiME-3 dataset: WSJ0 utterances in four noise environments.

Original data (similar noises types and SNRs across all datasets):

- training: 7138 simulated utterances
- development: 1640 real + 1640 simulated utterances
- test: 1320 real utterances.

Baseline: feedforward DNN acoustic model on fMLLR features + 3-gram language model. No speech enhancement.

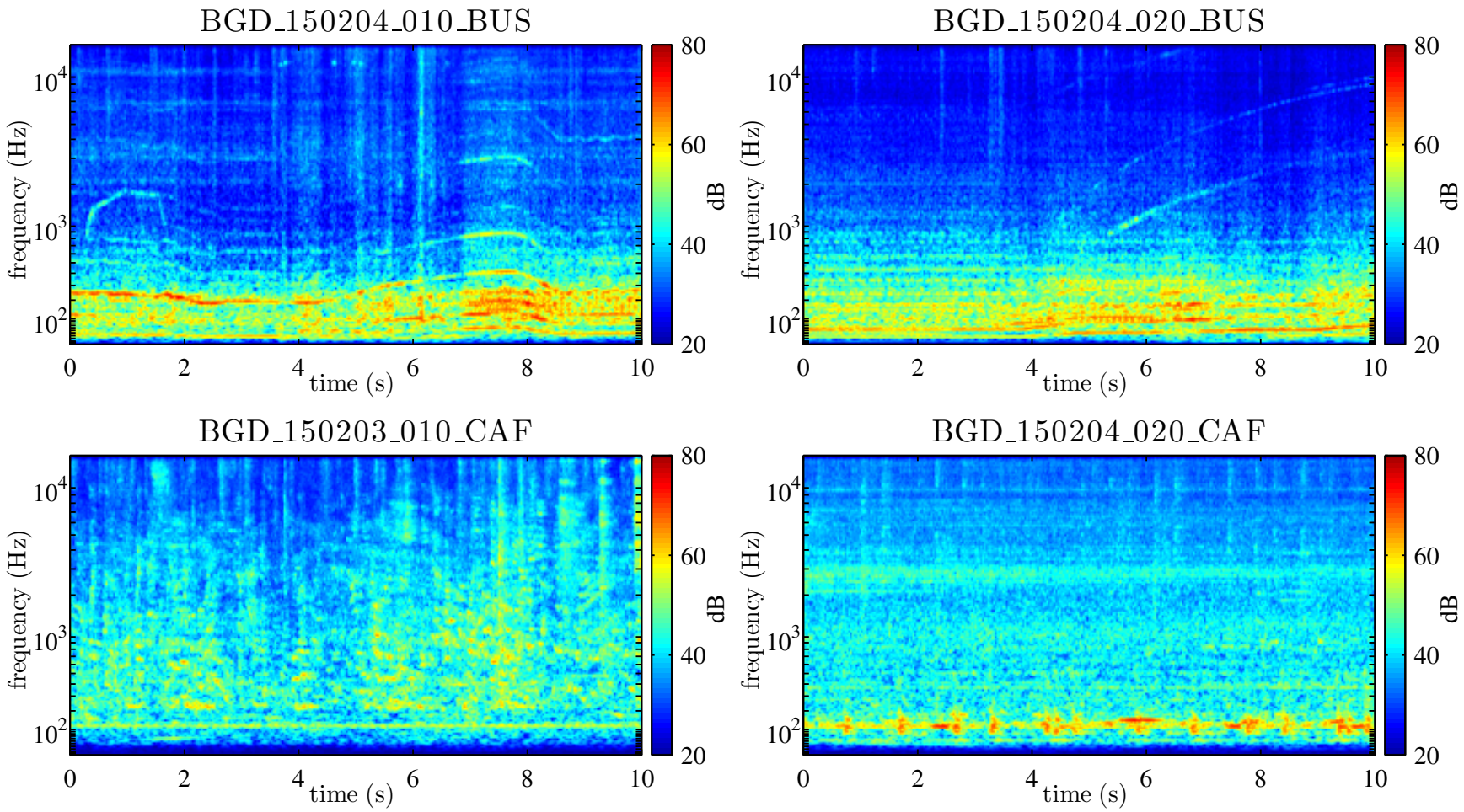
WER reduced from 33.4% (baseline) to 2.2% (best). Close to clean speech performance, possibly due to matched hardware and "all-inclusive" training data.

How much do environment or microphone mismatch affect enhancement and ASR performance?

IMPACT OF ENVIRONMENT AND MICROPHONE MISMATCHES

Environment mismatch

Noise characteristics vary within and across environments.



Environment mismatch — Impact on enhancement

WER achieved by multichannel enhancement (with baseline acoustic model)

Training (real)	Test (real) BUS	CAF	PED	STR	Avg.
BUS	21.03	13.06	17.92	9.28	15.32
CAF	31.48	13.15	16.95	8.78	17.59
PED	27.89	12.20	17.04	8.93	16.51
STR	24.30	11.80	16.42	8.48	15.25
1/4 of all	20.83	11.65	15.94	8.72	14.28
all but BUS	22.62	10.72	15.47	7.55	14.09
all but CAF	18.90	10.59	16.07	7.53	13.27
all but PED	18.56	10.76	14.93	8.09	13.08
all but STR	18.19	10.03	15.08	7.94	12.81
3/4 of all	18.84	10.98	15.41	7.79	13.26

1 training environment:
Multicondition: 14.28%
Matched: 14.93%
Mismatched: 16.58%

3 training environments:
Multicondition: 13.26%
Mismatched: 14.02%

$\Rightarrow$ multicondition training preferable to matched training

$\Rightarrow$ on average, performs well on environments not seen in training

Environment mismatch — Impact on ASR

WER achieved by acoustic model (no enhancement, RNN language model)

Training (real + sim)	Test (real) BUS	CAF	PED	STR	Avg.
BUS	45.56	33.34	26.53	17.71	30.78
CAF	44.33	23.22	18.78	16.88	25.80
PED	43.86	23.53	17.53	17.37	25.57
STR	40.31	28.63	22.27	16.14	26.83
1/4 of all	40.47	23.52	17.90	15.47	24.34
all but BUS	35.50	18.34	13.66	12.29	19.94
all but CAF	32.87	20.92	15.45	12.25	20.37
all but PED	32.62	20.64	15.66	12.33	20.31
all but STR	33.11	18.88	15.06	12.94	20.00
3/4 of all	32.75	19.41	13.45	12.40	19.50

1 training environment:
Multicondition: 24.34%
Matched: 25.61%
Mismatched: 27.80%

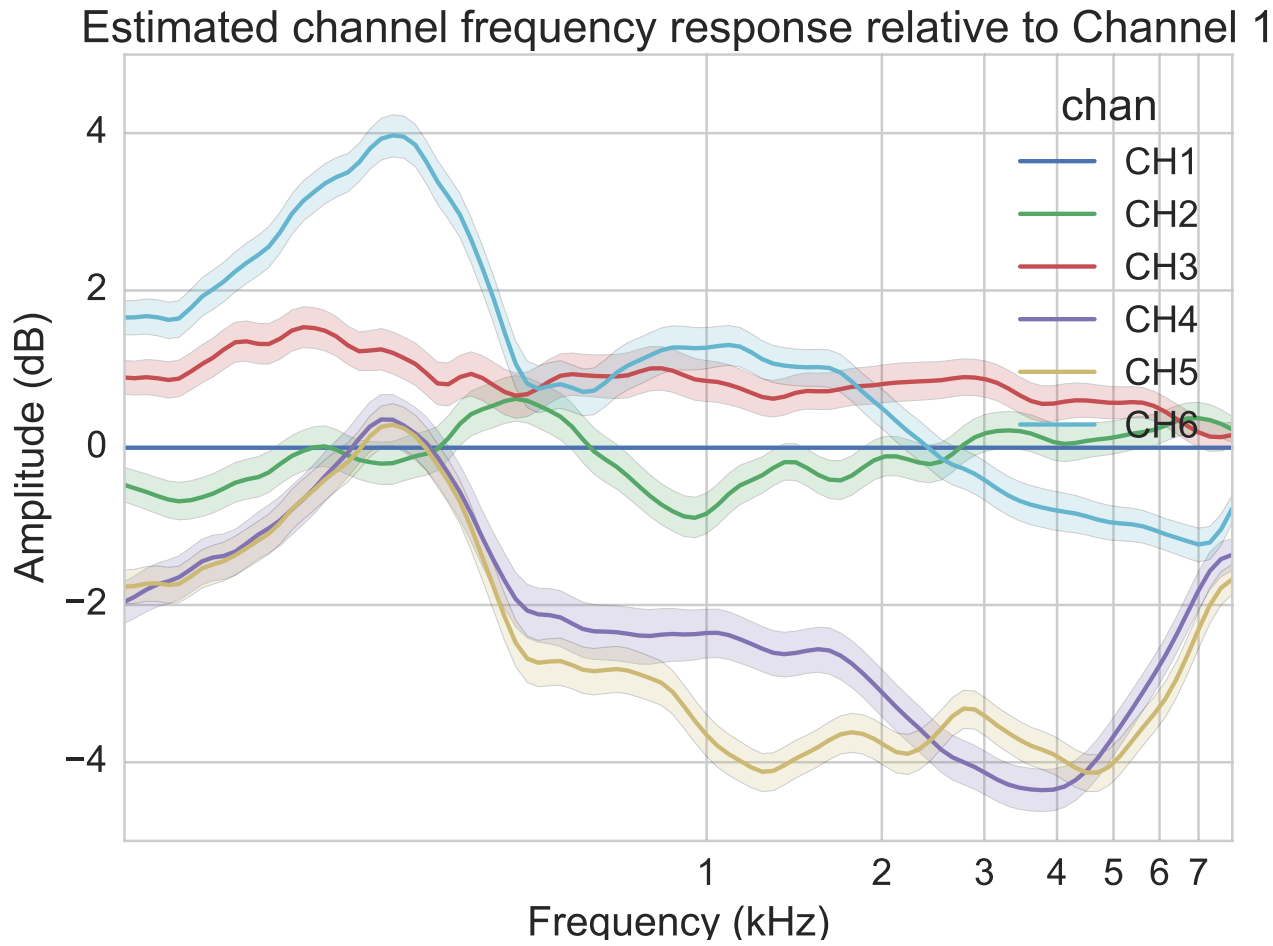
3 training environments:
Multicondition: 19.50%
Mismatched: 21.26%

⇒ similarly small impact on ASR as on enhancement

⇒ again, multicondition training performs best on average

Microphone response mismatch

Relative responses obtained by averaging over 1 min segments.



Microphone response mismatch — Impact on ASR

WER achieved by acoustic model (no enhancement, RNN language model)

Training (real + sim)	Test (real) ch1	ch2	ch3	ch4	ch5	ch6
ch1	31.65	83.01	35.20	30.62	27.07	30.76
ch2	32.43	71.36	35.85	32.44	29.12	31.77
ch3	30.86	83.04	34.94	30.20	26.26	30.91
ch4	31.65	82.74	35.64	28.74	25.13	29.27
ch5	33.60	84.05	37.70	30.25	25.84	30.73
ch6	31.33	81.65	35.18	28.25	24.72	28.04

Matched: 36.76%

Mismatched: 39.71%

$\Rightarrow$ matched training performs best on average

$\Rightarrow$ performs well on other mics, except backward mic (ch 2)

Is multicondition training the best one can do?

Environment mismatch does not improve performance

WER achieved by acoustic model (no enhancement, RNN language model)

Training (real + sim)	Test (real) BUS	CAF	PED	STR	Avg.
BUS	45.56	33.34	26.53	17.71	30.78
CAF	44.33	23.22	18.78	16.88	25.80
PED	43.86	23.53	17.53	17.37	25.57
STR	40.31	28.63	22.27	16.14	26.83
1/4 of all	40.47	23.52	17.90	15.47	24.34
all but BUS	35.50	18.34	13.66	12.29	19.94
all but CAF	32.87	20.92	15.45	12.25	20.37
all but PED	32.62	20.64	15.66	12.33	20.31
all but STR	33.11	18.88	15.06	12.94	20.00
3/4 of all	32.75	19.41	13.45	12.40	19.50

1 training environment:
Multicondition: 24.34%
Matched: 25.61%
Mismatched: 27.80%

3 training environments:
Multicondition: 19.50%
Mismatched: 21.26%
Better than multicondition (not significant)

$\Rightarrow$ excluding certain noises does not significantly improve performance

Microphone mismatch does not improve performance

WER achieved by acoustic model (no enhancement, RNN language model)

Training (real + sim)	Test (real) ch1	ch2	ch3	ch4	ch5	ch6
ch1	31.65	83.01	35.20	30.62	27.07	30.76
ch2	32.43	71.36	35.85	32.44	29.12	31.77
ch3	30.86	83.04	34.94	30.20	26.26	30.91
ch4	31.65	82.74	35.64	28.74	25.13	29.27
ch5	33.60	84.05	37.70	30.25	25.84	30.73
ch6	31.33	81.65	35.18	28.25	24.72	28.04

Matched: 36.76%

Mismatched: 39.71%

Better than matched (not significant)

$\Rightarrow$ training on another microphone does not significantly improve performance either

Data augmentation greatly improves performance

Simulated training data augmented 7 fold by changing the SNR by -15, -10, -5, $+5$, $+10$, and $+15$ dB relative wrt original.

Speech and noise signals unchanged.

Training dataset	**Development dataset**	**WER on test dataset (%)**				
		BUS	**CAF**	**PED**	**STR**	**Avg**
Original	Simu+real	50.08	27.27	20.37	18.51	29.05
Augment	Simu+real	36.85	26.84	20.80	15.22	**24.92**
Augment	Real	37.37	30.11	23.22	15.65	**26.59**

$\Rightarrow$ data augmentation improves WER by 8 or 14% relative (highly significant) despite increased SNR mismatch

Other practical evidence

Significant WER improvement also reported in the ASR literature when enhancing the test data, but not the training data.

T. Yoshioka, N. Ito, M. Delcroix, A. Ogawa, K. Kinoshita et al., "The NTT CHiME-3 system: Advances in speech enhancement and recognition for mobile multi-microphone devices," in Proc. ASRU, 2015.

See also more general proof in the machine learning literature.

C. R. González and Y. S. Abu-Mostafa, "Mismatched training and test distributions can outperform matched ones," Neural Computation, vol. 27, no. 2, pp. 365–387, Dec. 2015.

Optimizing the training set

What is the optimal training set given the task, the classifier, and the average test conditions (no adaptation)?

So far, tuned by trial and error.

Importance weighting

Optimization framework: importance weighting

- generate a large training set by applying all possibly relevant data augmentation techniques and parameters
- then weight every sample according to its "usefulness".

Conventional transfer learning: weight each training sample x by $p_{\text{test}}(x)/p_{\text{train}}(x)$ so that the training and test data are matched.

According to the above evidence, this is suboptimal $\Rightarrow$ discriminative approach required.

Idea: weight every training sample so as to minimize error on a development set.

DISCRIMINATIVE DATA AUGMENTATION

Classical DNN training objective

Classically, a DNN is trained to estimate the posterior $p_\theta(y|x)$ over labels (senones) y given inputs x.

The parameters θ are obtained by minimizing the average loss $\mathcal{L}$ on the training set:

$$\hat{\theta} = \arg\min_\theta E_{p_{\text{train}}(x)p(y|x)}[\mathcal{L}(p_\theta(y|x), y)].$$

e.g., using cross-entropy:

$$E\left[\mathcal{L}(p_\theta(y|x), y)\right] = -\frac{1}{N}\sum_{i=1}^{N} \log p_\theta(y_i|x_i).$$

Proposed discriminative importance weighting objective

Define a weighted loss function with importance weights $\omega_i \geq 0$, e.g., using cross-entropy:

$$E\left[\mathcal{L}_\omega(p_\theta(y|x), y, \omega)\right] = -\frac{\sum_{i=1}^{N} \omega_i \log p_\theta(y_i|x_i)}{\sum_{i=1}^{N} \omega_i}.$$

Solve the following nested optimization problem:

- find the DNN parameters $\hat{\theta}$ that minimize the weighted loss on the training set
- find the data weights $\hat{\omega}$ for which this DNN yields minimum average error rate $\mathcal{E}$ on the development set

$$\hat{\theta} = \arg\min_\theta E_{p_{\text{train}}(x)p(y|x)}\left[\mathcal{L}_\omega(p_\theta(y|x), y, \omega)\right]$$

$$\hat{\omega} = \arg\min_\omega E_{p_{\text{dev}}(x)p(y|x)}\left[\mathcal{E}(p_{\hat{\theta}}(y|x), y)\right].$$

Algorithm

Alternating optimization algorithm:

- update the parameters θ via one epoch of stochastic gradient descent (SGD) on the full set
  - compute gradient for each x_i by usual backpropagation
  - multiply it by $\omega_i / \sum_{i=1}^{N} \omega_i$ before summing over time
- update the weights ω using one step of gradient descent
  - for each weight ω_i, update the DNN via one step of SGD on a single sample x_i
  - compute resulting difference Δe_i in the classification error
  - update the weights as $\omega_i = \omega_i - \lambda \Delta e_i$ with a learning rate λ

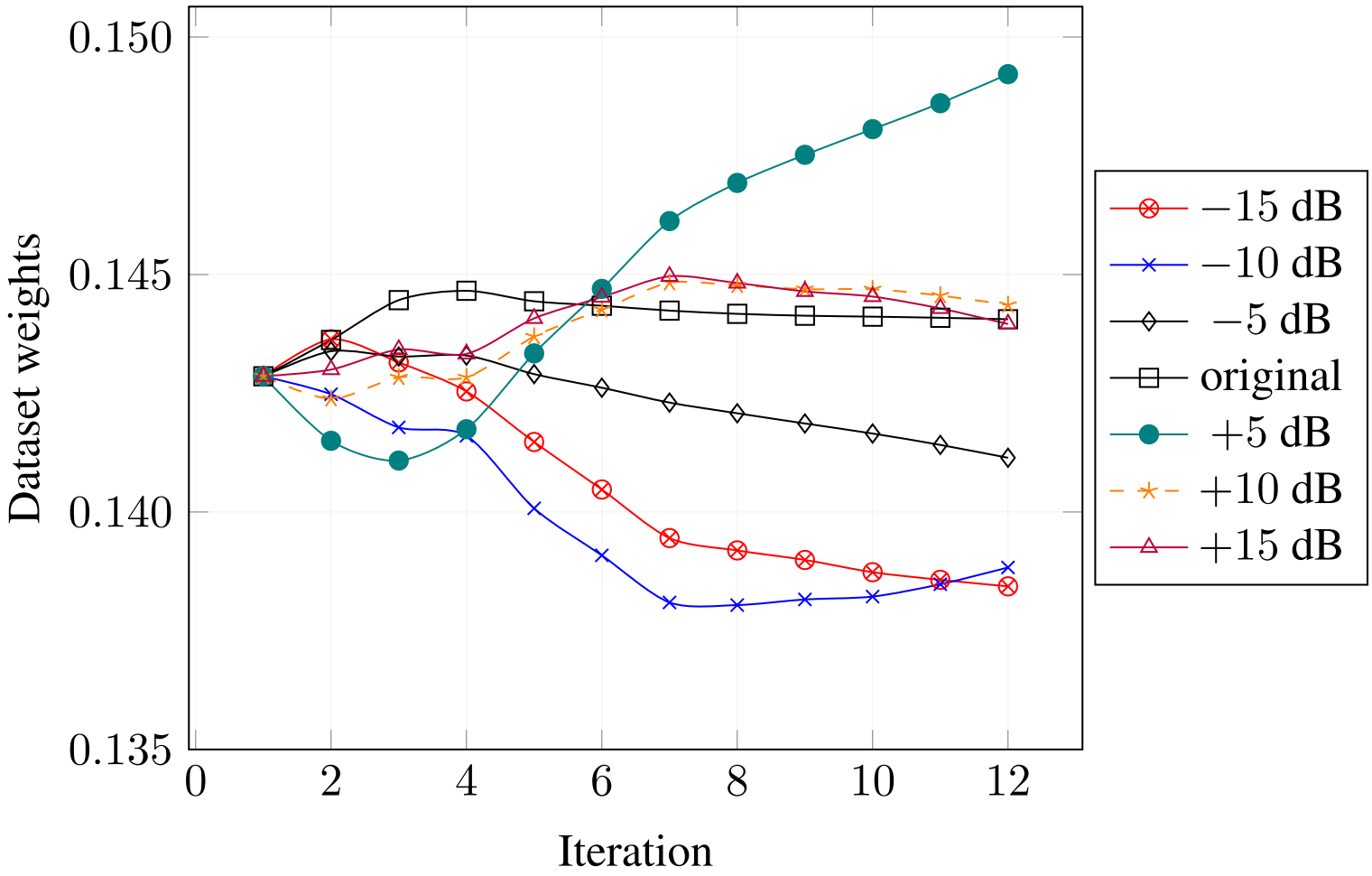
In practice, weights are tied across subsets of data.

Experimental setup

Training set augmented by changing the SNR by -15, -10, -5, $+5$, $+10$, and $+15$ dB relative for the same utterances and noises.

Weights tied across all utterances generated using the same relative SNR w.r.t. the original training set $\Rightarrow$ 7 weights in total.

Results



$\Rightarrow$ resulting SNR distribution $\neq$ original, hence $\neq$ test

Results

Training dataset	Development dataset	WER on test dataset (%)				
		BUS	CAF	PED	STR	Avg
Original	Simu+real	50.08	27.27	20.37	18.51	29.05
Augment	Simu+real	36.85	26.84	20.80	15.22	24.92
Augment+weight	Simu+real	34.23	23.53	19.64	13.82	**22.80**
Augment	Real	37.37	30.11	23.22	15.65	26.59
Augment+weight	Real	32.60	24.15	19.86	14.10	**22.68**

$\Rightarrow$ discriminative weighting further improves the WER

$\Rightarrow$ but weights not so contrasted...

CONCLUSION

Conclusion

DNNs for speech enhancement or ASR are surprisingly robust to mismatched training data.

SNR-augmented training data improves performance despite SNR mismatch. Not all mismatched training data do.

Discriminative weighting algorithm can potentially improve performance, but efficiency and understanding still to be improved.

References

D. Ribas, E. Vincent, J. R. Calvo, "A study of speech distortion conditions in real scenarios for speech processing applications", in *Proc. SLT*, 2016.

A. A. Nugraha, A. Liutkus, E. Vincent, "Multichannel audio source separation with deep neural networks", *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 2016.

E. Vincent, S. Watanabe, A. A. Nugraha, J. Barker, R. Marxer, "An analysis of environment, microphone and data simulation mismatches in robust speech recognition", *Computer Speech and Language*, 2017.

S. Sivasankaran, E. Vincent, I. Illina, "Discriminative importance weighting of augmented training data for acoustic model training", in *Proc. ICASSP*, 2017.