RUHR-UNIVERSITÄT BOCHUM
INESC-ID Lisboa

CHiME Challenge:

Approaches to Robustness using Beamforming and Uncertainty-of-Observation Techniques

Dorothea Kolossa [1], Ramón Fernandez Astudillo [2], Alberto Abad [2], Steffen Zeiler [1], Rahim Saeidi [3], Pejman Mowlaee [1], João Paulo da Silva Neto [2], Rainer Martin [1]

[1] Institute of Communication Acoustics (IKA) Ruhr-Universität Bochum
[2] Spoken Language Laboratory, INESC-ID, Lisbon
[3] School of Computing, University of Eastern Finland

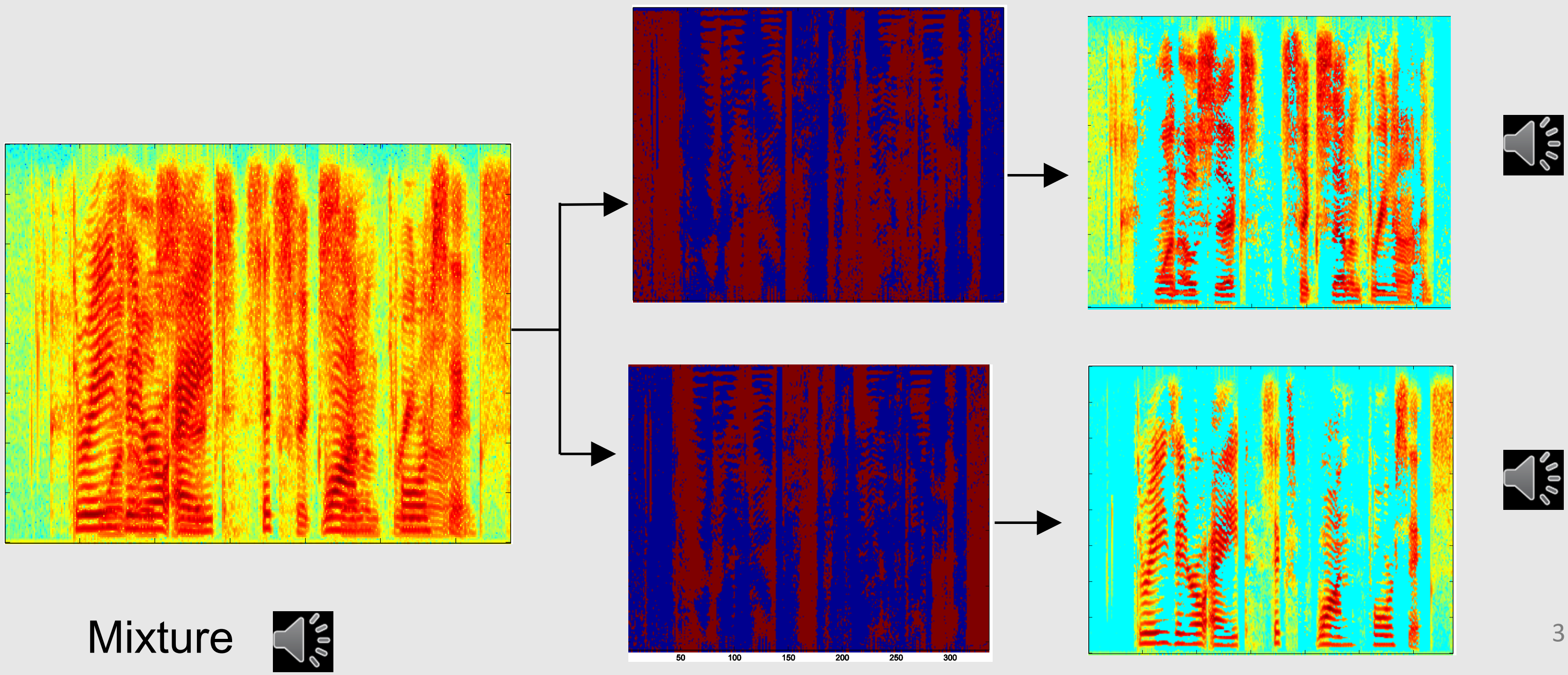
Overview

- Uncertainty-Based Approach to Robust ASR
- Uncertainty Estimation by Beamforming & Propagation
- Recognition under Uncertain Observations
- Further Improvements
  - Training: Full-covariance Mixture Splitting
  - Integration: Rover
- Results and Conclusions

Introduction: Uncertainty-Based Approach to ASR Robustness

- Speech enhancement in time-frequency-domain is often very effective.
- However, speech enhancement itself can neither
  - remove all distortions and sources of mismatch completely
  - nor can it avoid introducing artifacts itself

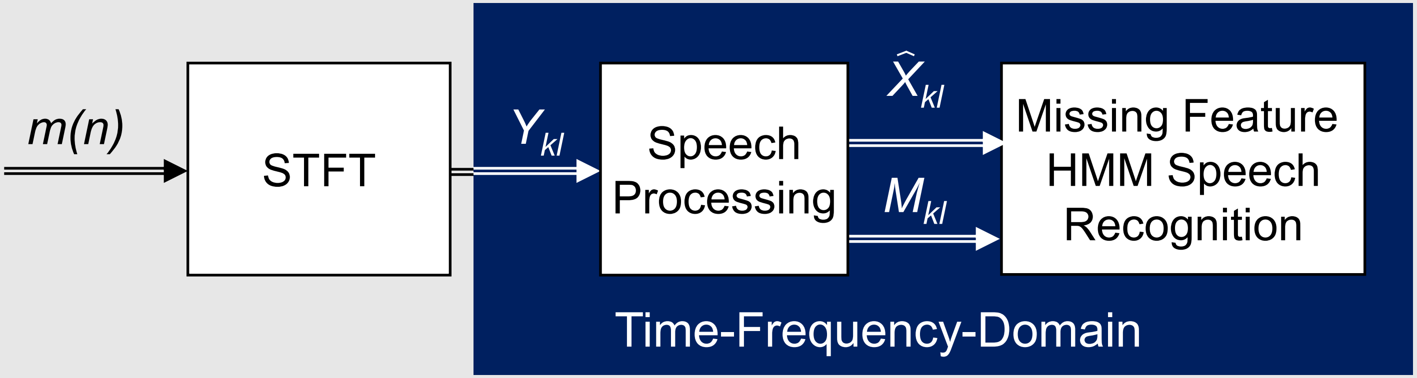
Simple example: Time-Frequency Masking



Introduction: Uncertainty-Based Approach to ASR Robustness

How can decoder handle such artificially distorted signals?

One possible compromise:

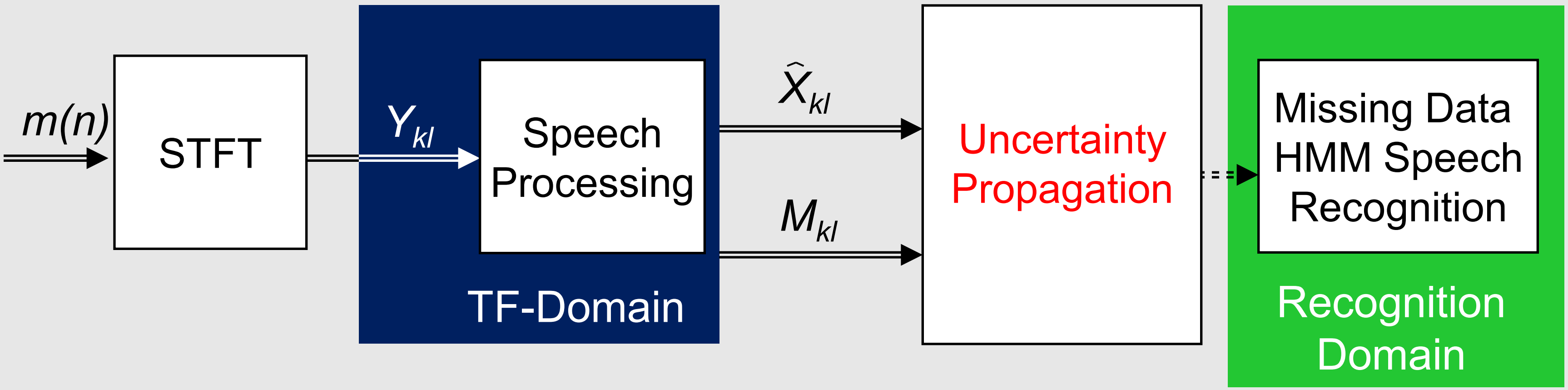


Problem: Recognition performs significantly better in other domains, such that missing feature approach may perform worse than feature reconstruction [1].

[1] B. Raj and R. Stern: „Reconstruction of Missing Features for Robust Speech Recognition“, Speech Communication 43, pp. 275-296, 2004.

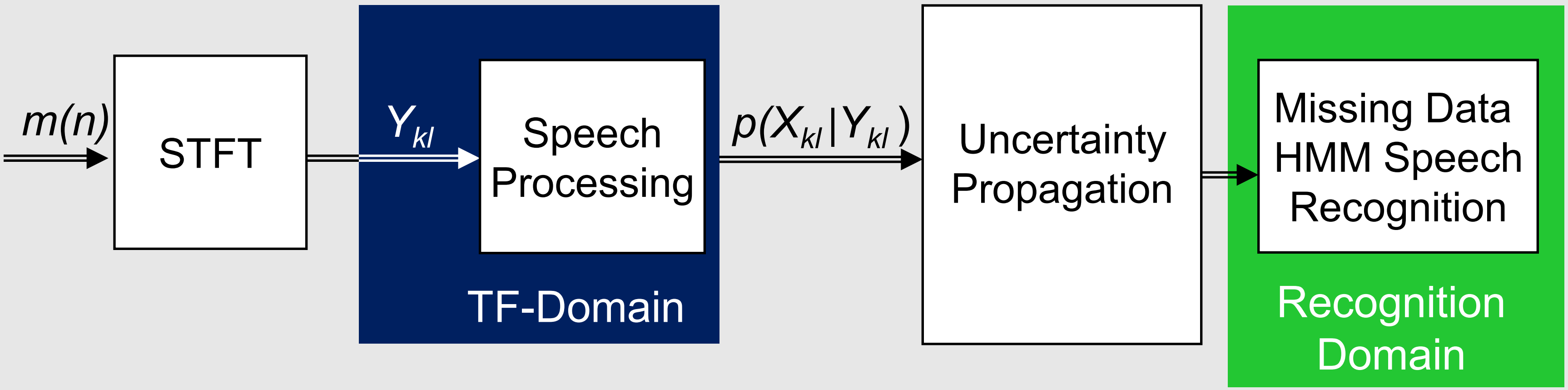
Introduction: Uncertainty-Based Approach to ASR Robustness

Solution used here:
Transform uncertain features to desired domain of recognition



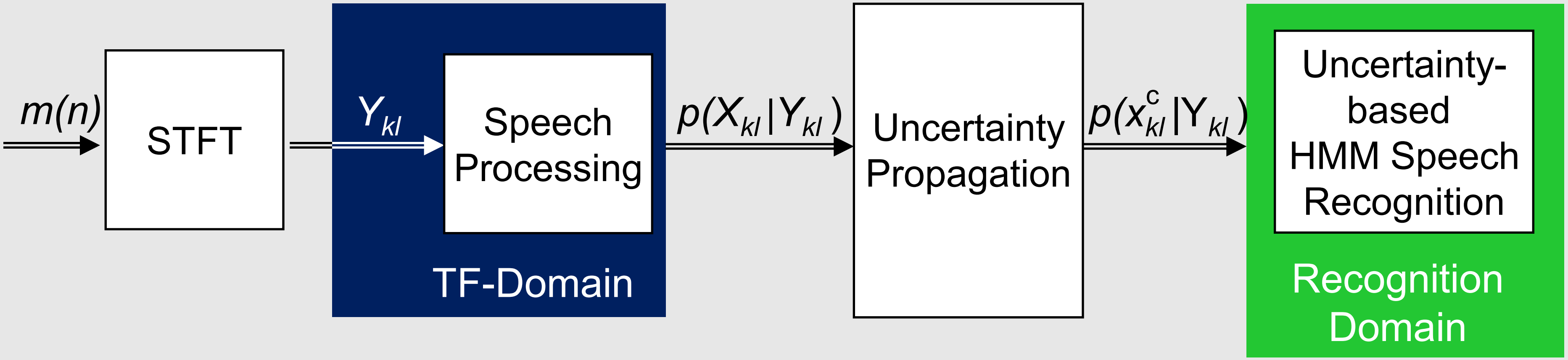
Introduction: Uncertainty-Based Approach to ASR Robustness

Solution used here:
Transform uncertain features to desired domain of recognition

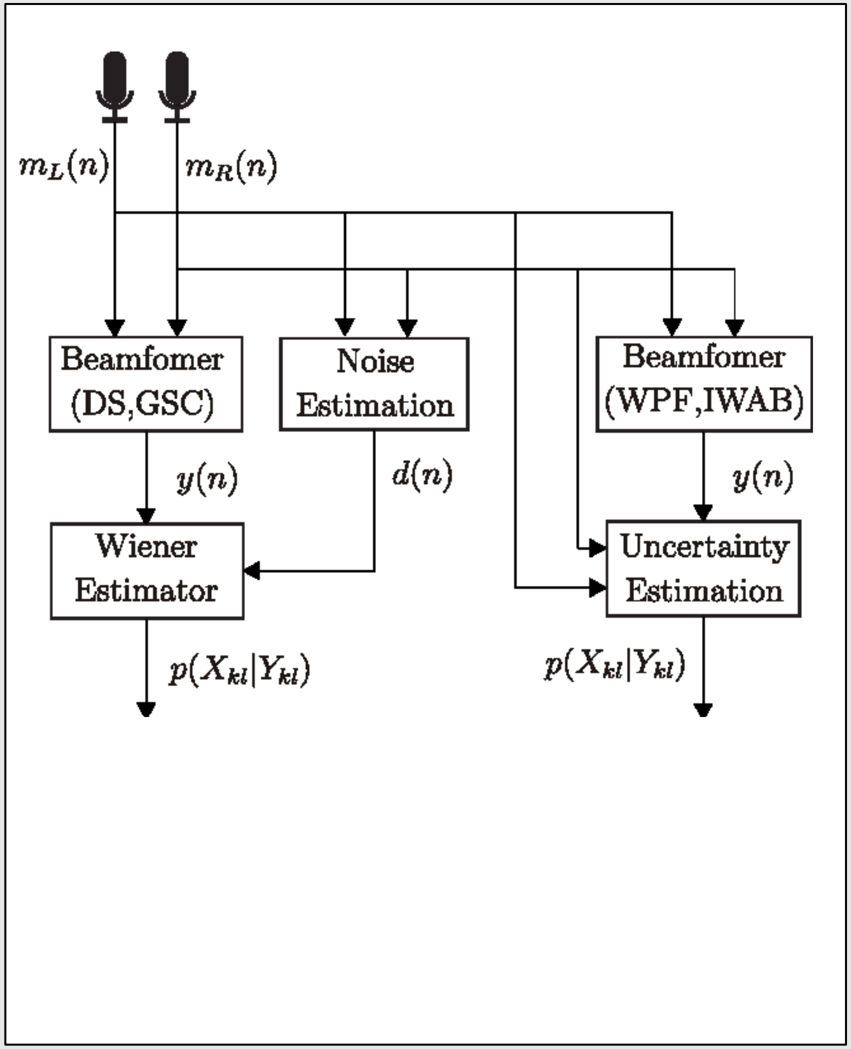


Introduction: Uncertainty-Based Approach to ASR Robustness

Solution used here:
Transform uncertain features to desired domain of recognition



Uncertainty Estimation & Propagation

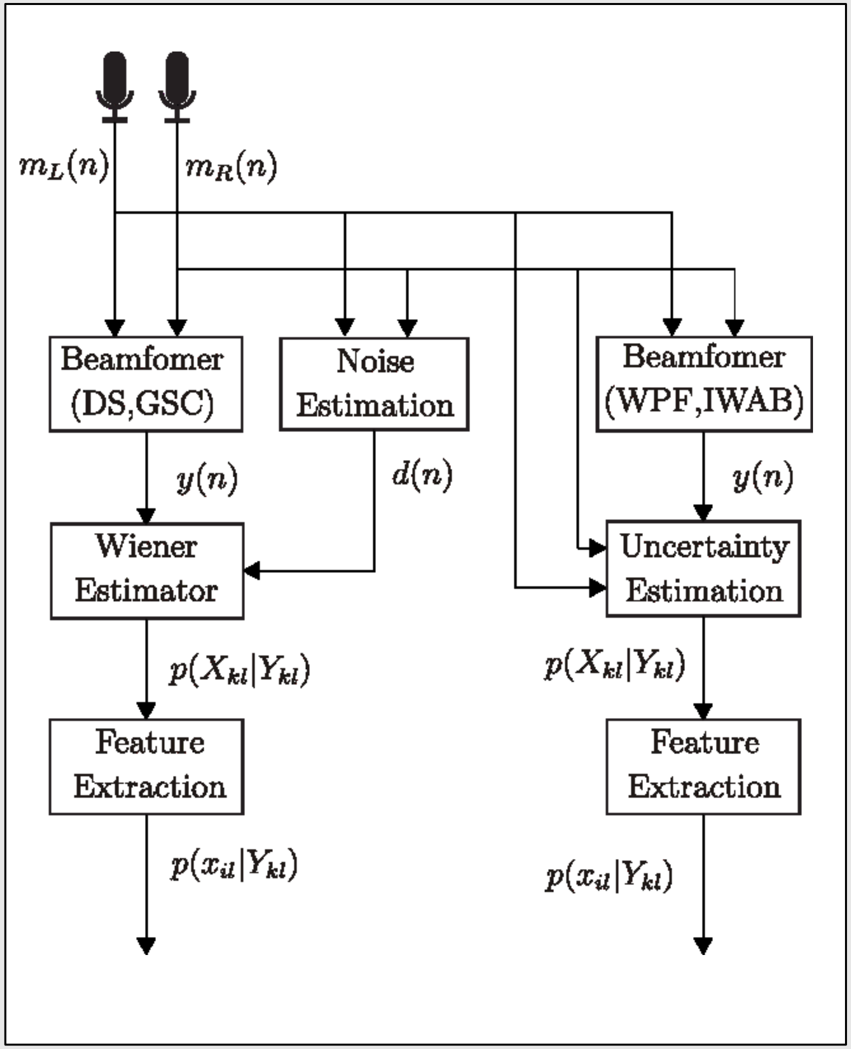


- Posterior estimation here is performed by using one of four beamformers:
  - Delay and Sum (DS)
  - Generalized Sidelobe Canceller (GSC) [2]
  - Multichannel Wiener Filter (WPF)
  - Integrated Wiener Filtering with Adaptive Beamformer (IWAB) [3]

[2] O. Hoshuyama, A. Sugiyama, and A. Hirano, “A robust adaptive beamformer for microphone arrays with a blocking matrix using constrained adaptive filters,” IEEE Trans. Signal Processing, vol. 47, no. 10, pp. 2677 –2684, 1999.

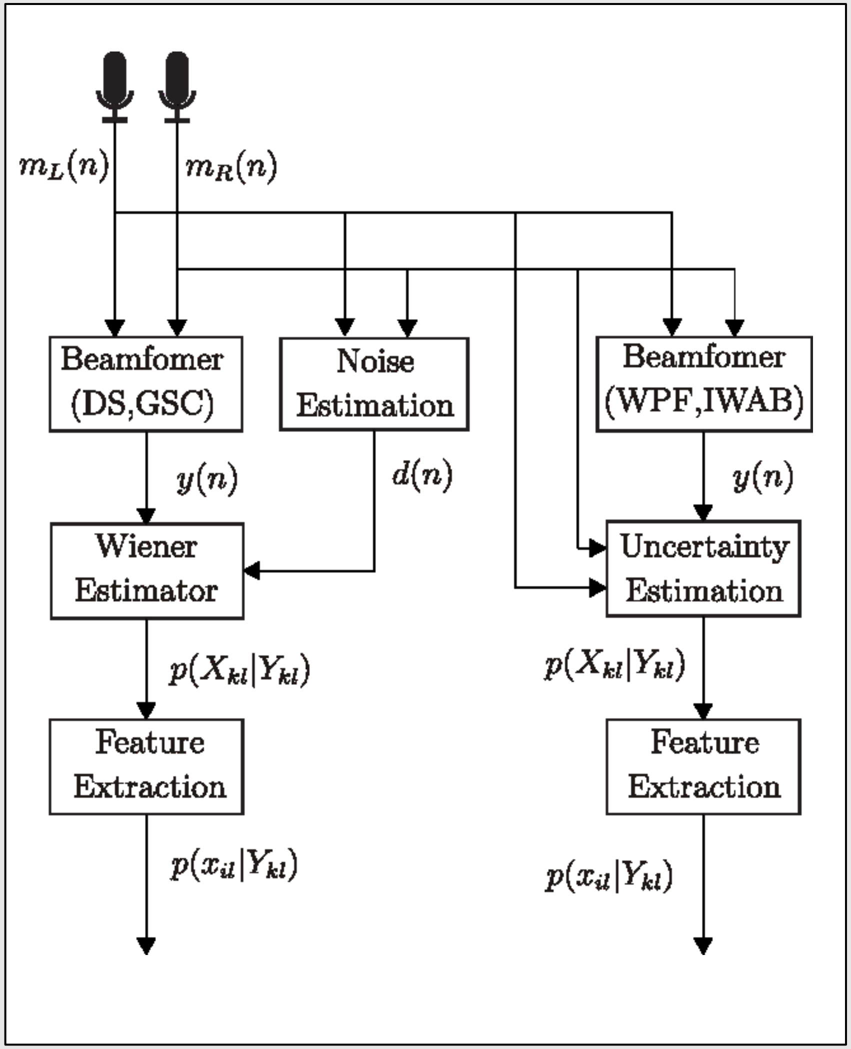
[3] A. Abad and J. Hernando, “Speech enhancement and recognition by integrating adaptive beamforming and Wiener filtering,” in Proc. 8th International Conference on Spoken Language Processing (ICSLP), 2004, pp. 2657–2660.

Uncertainty Estimation & Propagation



- Posterior of clean speech, $p(X_{kl}\,|Y_{kl})$, is then propagated into domain of ASR
- Feature Extraction
  - STSA-based MFCCs
  - CMS per utterance
  - possibly LDA

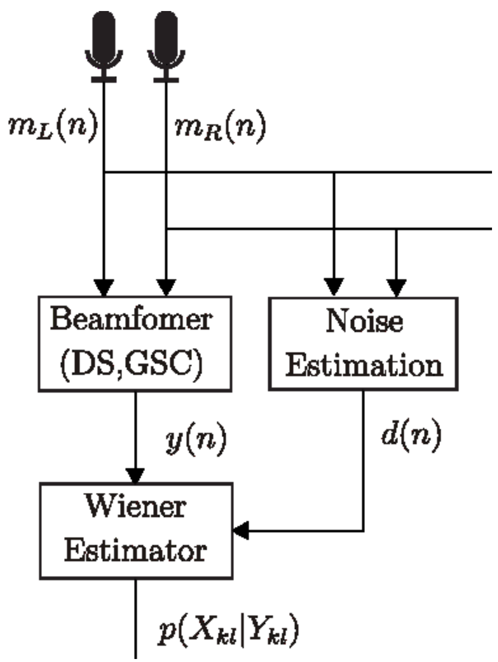
Uncertainty Estimation & Propagation



- Uncertainty model:

Complex Gaussian distribution

Uncertainty Estimation & Propagation



- Two uncertainty estimators:

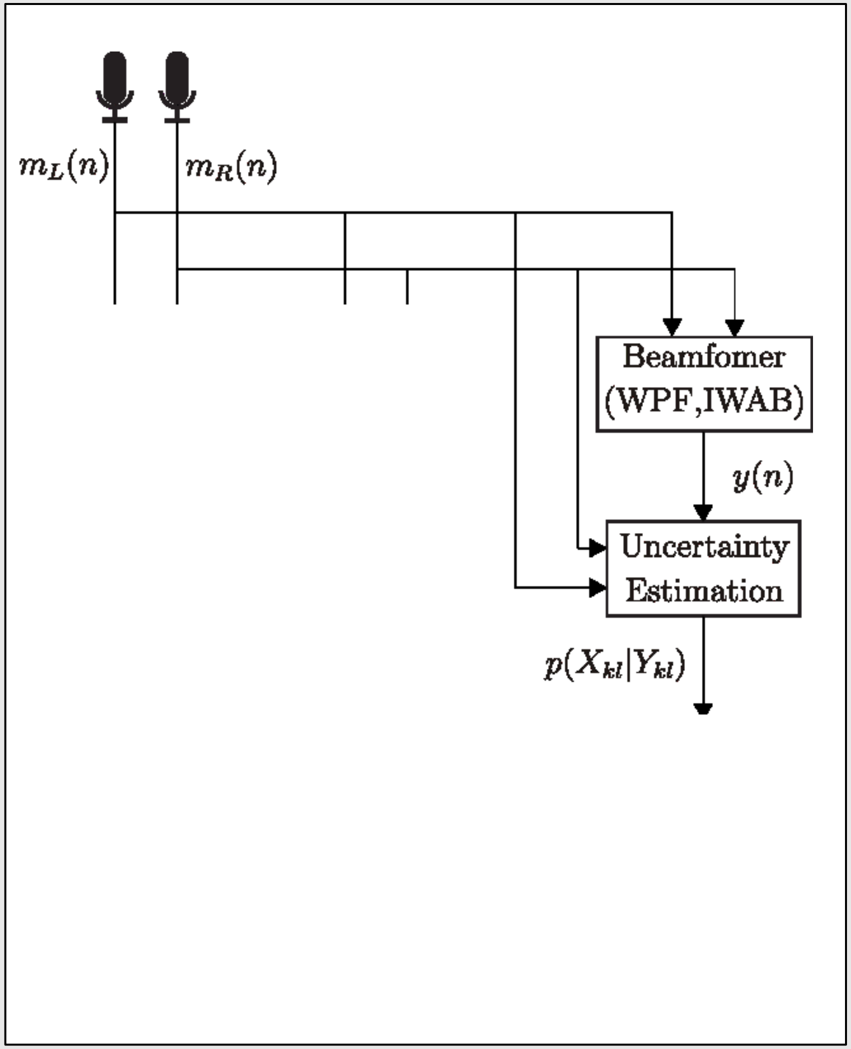
a) Channel Asymmetry Uncertainty Estimation
   - Beamformer output input to Wiener filter
   - Noise variance estimated as squared channel difference
   - Posterior directly obtainable for Wiener filter [4]:

$$\lambda_{\mathbf{D}} = \mathrm{DFT}\{(m_L(n) - m_R(n))^2\}$$

$$p(X_{kl}|Y_{kl}) = \mathcal{N}\left(\frac{\lambda_{X_{kl}}}{\lambda_{D_{kl}} + \lambda_{X_{kl}}} Y_{kl}; \frac{\lambda_{X_{kl}} \lambda_{D_{kl}}}{\lambda_{D_{kl}} + \lambda_{X_{kl}}}\right)$$

[4] R. F. Astudillo and R. Orglmeister, “A MMSE estimator in mel-cepstral domain for robust large vocabulary automatic speech recognition using uncertainty propagation,” in Proc. Interspeech, 2010, pp. 713–716.

Uncertainty Estimation & Propagation



- Two uncertainty estimators:

b) Equivalent Wiener variance

- Beamformer output directly passed to feature extraction

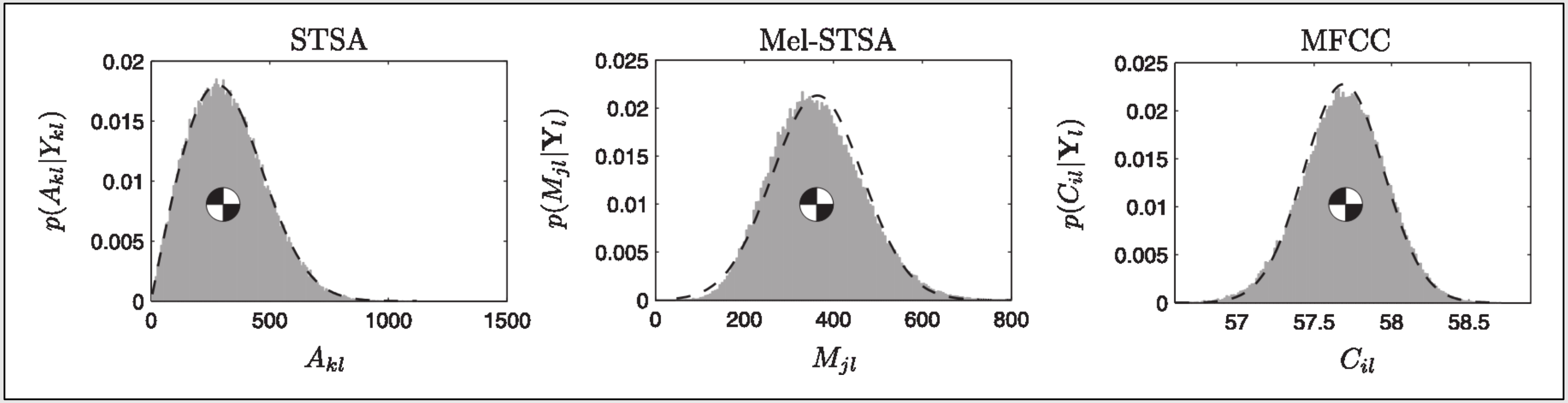
$$p(X_{kl}|Y_{kl}) = \mathcal{N}\left(Y_{kl}, \tilde{\lambda}_{kl}\right)$$

- Variance estimated using ratio of beamformer input and output, interpreted as Wiener gain

[4] R. F. Astudillo and R. Orglmeister, "A MMSE estimator in mel-cepstral domain for robust large vocabulary automatic speech recognition using uncertainty propagation," in Proc. Interspeech, 2010, pp. 713–716.

Uncertainty Propagation

- Uncertainty propagation from [5] was used
  - Propagation through absolute value yields MMSE-STSA
  - Independent log normal distributions after filterbank assumed



  - Posterior of clean speech in cepstrum domain assumed Gaussian
  - CMS and LDA transformations simple

[5] R. F. Astudillo, "Integration of short-time Fourier domain speech enhancement and observation uncertainty techniques for robust automatic speech recognition," Ph.D. thesis, Technical University Berlin, 2010.

Recognition under Uncertain Observations

- Standard observation likelihood for state q mixture m:

$$p(x \mid \mu_{q,m}, \Sigma_{q,m}) = N(x; \mu_{q,m}, \Sigma_{q,m})$$

- Uncertainty Decoding:

$$p(\mu_x \mid \mu_{q,m}, \Sigma_{q,m}, \Sigma_x) = N(\mu_x; \mu_{q,m}, \Sigma_{q,m} + \Sigma_x)$$

L. Deng, J. Droppo, and A. Acero, "Dynamic compensation of HMM variances using the feature enhancement uncertainty computed from a parametric model of speech distortion," IEEE Trans. Speech and Audio Processing, vol. 13, no. 3, pp. 412–421, May 2005.

- Modified Imputation:

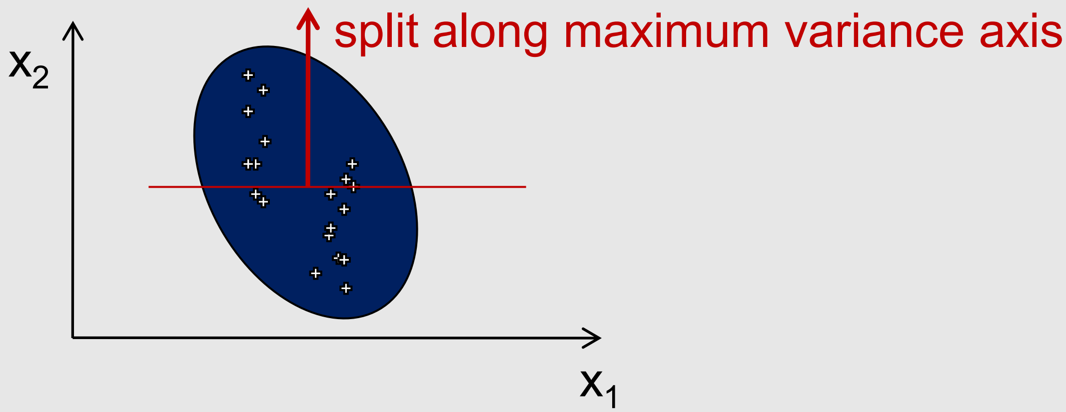
$$p(\mu_x \mid \mu_{q,m}, \Sigma_{q,m}, \Sigma_x) = \mathcal{N}(\hat{x}; \mu_{q,m}, \Sigma_{q,m})$$
$$\text{with } \hat{x} = (\Sigma_{q,m} + \Sigma_x)^{-1}(\Sigma_{q,m}\mu_x + \Sigma_x \mu_{q,m})$$

D. Kolossa, A. Klimas, and R. Orglmeister, "Separation and robust recognition of noisy, convolutive speech mixtures using time-frequency masking and missing data techniques," in Proc. Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA), Oct. 2005, pp. 82–85.

- Both uncertainty-of-observation techniques collapse to standard observation likelihood for $\Sigma_x = 0$.

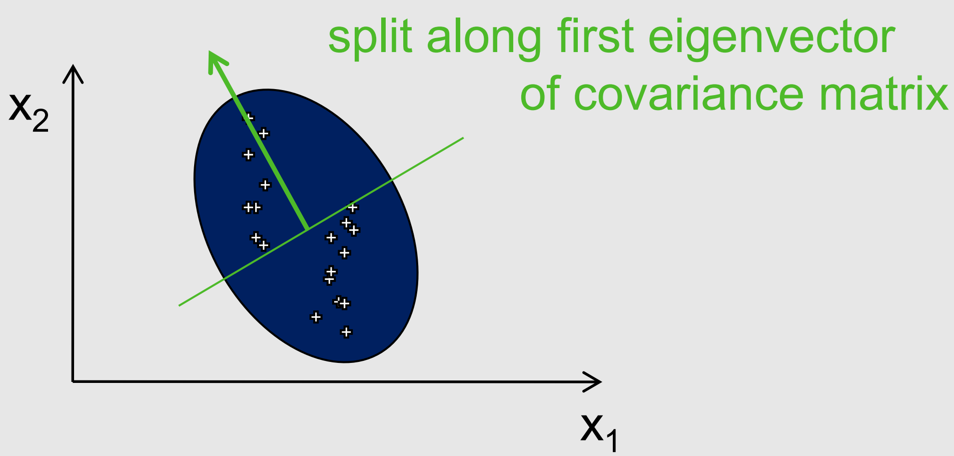
Further Improvements

- Training: Informed Mixture Splitting
  - Baum-Welch Training is only optimal locally -> good initialization and good split directions matter.
  - Therefore, considering covariance structure in mixture splitting is advantageous:



Further Improvements

- Training: Informed Mixture Splitting
  - Baum-Welch Training is only optimal locally -> good initialization and good split directions matter.
  - Therefore, considering covariance structure in mixture splitting is advantageous:



Further Improvements

- Integration: Recognizer output voting error reduction (ROVER)
  - Recognition outputs at word level are combined by dynamic programming on generated lattice, taking into account
    - the frequency of word labels and
    - the posterior word probabilities
  - We use ROVER on 3 jointly best systems selected on development set.

J. Fiscus, “A post-processing system to yield reduced word error rates: Recognizer output voting error reduction (ROVER),” in IEEE Workshop on Automatic Speech Recognition and Understanding, Dec. 1997, pp. 347 –354.

Results and Conclusions

- Evaluation:
  - Two scenarios are considered, clean training and multicondition (‚mixed') training.
  - In mixed training, *all* training data was used at *all* SNR levels, artifically adding *randomly* selected noise from noise-only recordings.
  - Results are determined on the development set first.
  - After selecting the best performing system on development data, final results are obtained as *keyword accuracies* on the *isolated sentences* of the *test set.*

Results and Conclusions

- JASPER Results after clean training

	-6dB	-3dB	0dB	3dB	6dB	9dB
Clean: Official Baseline	30.33	35.42	49.50	62.92	75.00	82.42
JASPER* Baseline	40.83	49.25	60.33	70.67	79.67	84.92

* JASPER uses full covariance training with MCE iteration control. Token passing is equivalent to HTK.

Results and Conclusions

- JASPER Results after clean training

	-6dB	-3dB	0dB	3dB	6dB	9dB
Clean: Official Baseline	30.33	35.42	49.50	62.92	75.00	82.42
JASPER Baseline	40.83	49.25	60.33	70.67	79.67	84.92
JASPER + BF* + UP	54.50	61.33	72.92	82.17	87.42	90.83

* Best strategy here:

Delay and sum beamformer + noise estimation + modified imputation

Results and Conclusions

- HTK Results after clean training

	-6dB	-3dB	0dB	3dB	6dB	9dB
Clean: Official Baseline	30.33	35.42	49.50	62.92	75.00	82.42
HTK + BF* + UP	42.33	51.92	61.50	73.58	80.92	88.75

* Best strategy here:

Wiener post filter + uncertainty estimation

Results and Conclusions

- Results after clean training

	-6dB	-3dB	0dB	3dB	6dB	9dB
Clean: Official Baseline	30.33	35.42	49.50	62.92	75.00	82.42
HTK + BF + UP	42.33	51.92	61.50	73.58	80.92	88.75
HTK + BF* + UP + MLLR	54.83	65.17	74.25	82.67	87.25	91.33

* Best strategy here:

Delay and sum beamformer + noise estimation

Results and Conclusions

- Overall Results after clean training

	-6dB	-3dB	0dB	3dB	6dB	9dB
Clean: Official Baseline	30.33	35.42	49.50	62.92	75.00	82.42
JASPER Baseline	40.83	49.25	60.33	70.67	79.67	84.92
JASPER + BF + UP	54.50	61.33	72.92	82.17	87.42	90.83
HTK + BF + UP	42.33	51.92	61.50	73.58	80.92	88.75
HTK + BF + UP + MLLR	54.83	65.17	74.25	82.67	87.25	91.33
ROVER (JASPER + HTK)*	57.58	64.42	76.75	86.17	88.58	92.75

* (JASPER +DS + MI) & (HTK+GSC+NE) & (JASPER+WPF+MI)

Results and Conclusions

- JASPER Results after multicondition training

	-6dB	-3dB	0dB	3dB	6dB	9dB
Multicondition: HTK Baseline	63.00	72.67	79.50	85.25	89.75	93.58
JASPER Baseline	64.33	73.08	81.75	85.67	89.50	91.17

Results and Conclusions

- JASPER Results after multicondition training

	-6dB	-3dB	0dB	3dB	6dB	9dB
Multicondition: HTK Baseline	63.00	72.67	79.50	85.25	89.75	93.58
JASPER Baseline	64.33	73.08	81.75	85.67	89.50	91.17
JASPER + BF* + UP	73.92	79.08	86.25	89.83	91.08	93.00

* best JASPER setup here: Delay and sum beamformer + noise estimation + modified imputation + LDA to 37d

Results and Conclusions

- JASPER Results after multicondition training

	-6dB	-3dB	0dB	3dB	6dB	9dB
Multicondition: HTK Baseline	63.00	72.67	79.50	85.25	89.75	93.58
JASPER Baseline	64.33	73.08	81.75	85.67	89.50	91.17
JASPER + BF* + UP	73.92	79.08	86.25	89.83	91.08	93.00
as above, but 39d	+0.58%	-0.25%	-2.16%	-1.41%	-2.0%	-0.5%

* best JASPER setup here: Delay and sum beamformer + noise estimation + modified imputation + LDA to 37d

Results and Conclusions

- HTK Results after multicondition training

	-6dB	-3dB	0dB	3dB	6dB	9dB
Multicondition: HTK Baseline	63.00	72.67	79.50	85.25	89.75	93.58
HTK + BF* + UP	67.92	77.75	84.17	89.00	91.00	92.75

* best HTK setup here: Delay and sum beamformer + noise estimation

Results and Conclusions

- HTK Results after multicondition training

	-6dB	-3dB	0dB	3dB	6dB	9dB
Multicondition: HTK Baseline	63.00	72.67	79.50	85.25	89.75	93.58
HTK + BF + UP	67.92	77.75	84.17	89.00	91.00	92.75
HTK + BF* + UP + MLLR	68.25	79.75	84.67	89.58	91.25	92.92

* best HTK setup here: Delay and sum beamformer + noise estimation

Results and Conclusions

- Overall Results after multicondition training

	-6dB	-3dB	0dB	3dB	6dB	9dB
Multicondition: HTK Baseline	63.00	72.67	79.50	85.25	89.75	93.58
JASPER Baseline	64.33	73.08	81.75	85.67	89.50	91.17
JASPER + BF + UP	73.92	79.08	86.25	89.83	91.08	93.00
HTK + BF + UP	67.92	77.75	84.17	89.00	91.00	92.75
HTK + BF + UP + MLLR	68.25	79.75	84.67	89.58	91.25	92.92
ROVER (JASPER + HTK)*	74.58	80.58	87.92	90.83	92.75	94.17

* (JASPER +DS + MI + LDA) & (JASPER+WPF, no observation uncertainties) & (HTK+DS+NE)

Results and Conclusions

- Conclusions
  - Beamforming provides an opportunity to estimate not only the clean signal but also its standard error.
  - This error - the observation uncertainty - can be propagated to the MFCC domain or an other suitable domain for improving ASR by uncertainty-of-observation techniques.
  - Best results were attained for uncertainty propagation with modified imputation.
  - Training is critical, and despite strange philosophical implications, observation uncertainties improve the behaviour after multicondition training as well.
  - Strategy is simple & easily generalizes to LVCSR.

Thank you !

Further Improvements

- Training: MCE-Guided Training
  - Iteration and splitting control is done by minimum classification error (MCE) criterion on held-out dataset.
  - Algorithm for mixture splitting:
    - initialize split distance *d*
    - while *m* < *numMixtures*
      - split all mixtures by distance *d* along 1st eigenvector
      - carry out re-estimations until accuracy improves no more
      - if $acc_m >= acc_{m-1}$
        - *m* = *m*+1
      - else
        - go back to previous model
        - *d* = *d*/*f*