Uncertainty Handling for Environment-Robust Speech Recognition

Introduction

Ramón Fernandez Astudillo - L2F INESC-ID Lisboa - ramon@astudillo.com

Li Deng - Microsoft Research, Redmond - deng@microsoft.com

Emmanuel Vincent - INRIA Rennes - emmanuel.vincent@inria.fr

Introduction Overview

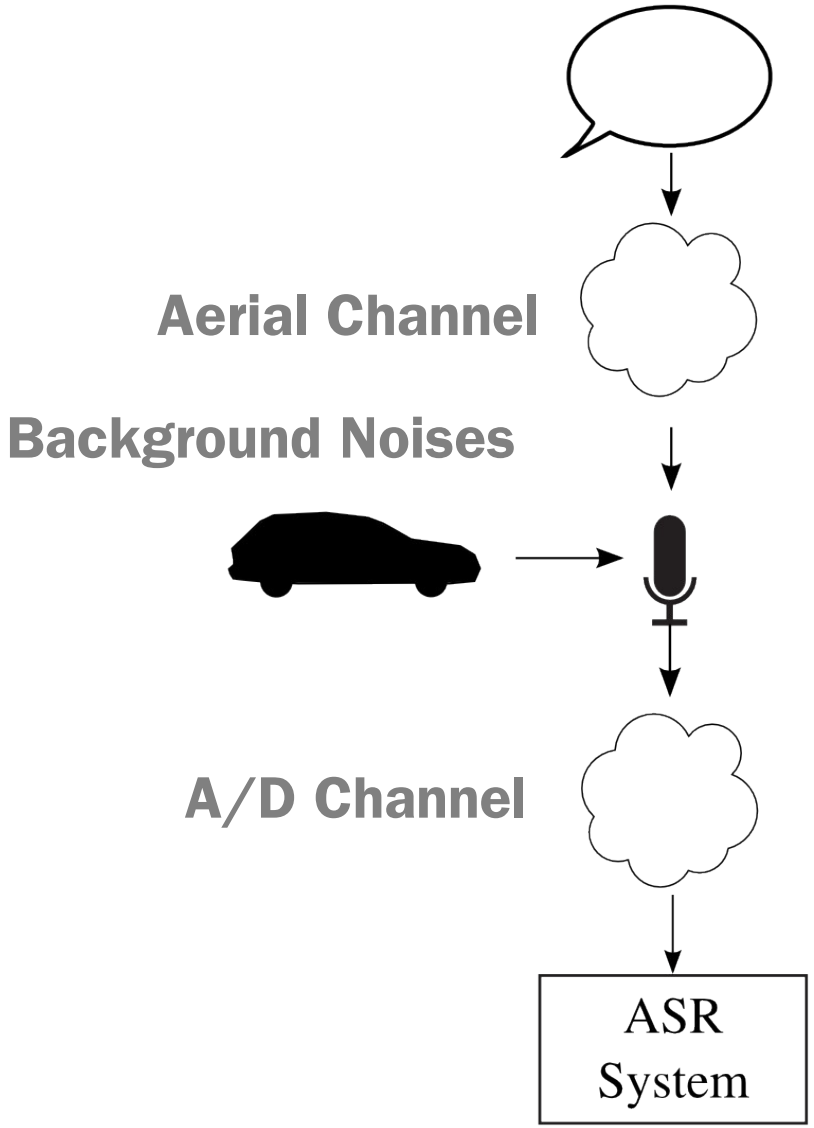
- Approaches to robust Automatic Speech Recognition (ASR)
- STFT-domain, log-feature-domain and notation
- Uncertainty Handling, brief tutorial overview

Automatic Speech Recognition (ASR)

> “Machine-based transcription of an acoustic speech signal into written words”

Ideal World ASR

- ASR system learns acoustic (and language) models from large corpora
- During recognition most probable sequence(s) of words acording to model provided
- This works assuming that input signal and trained models match



The Effect of the Environment

- Aerial channel

  Background noises
  Reverberation
  Gain oscilations

- A/D channel:

  Microphone characterictics
  Packet loss
  Signal processing artifacts

- Many unknown sorces of distortion!

Approaches to Robust ASR

- Speech (Feature) Domain Techniques

  Reseach field of its own (STFT domain speech enhancement)

  Decoupled from ASR system

  Low computational cost

  Limited improvements in performance

Approaches to Robust ASR

- Model Domain Techniques

  Compensate ASR model to better match corrupted signal

  Computational cost scales with size of ASR model

  High computational cost

  Bigger improvements in performance than feature compensation

Approaches to Robust ASR

- Mixed training (very large corpora)

  Include corrupted speech into training data

  Current corpora/model sizes allow partial modeling of corrupted acoustic space

  Feature and model compensation brings little additional improvement

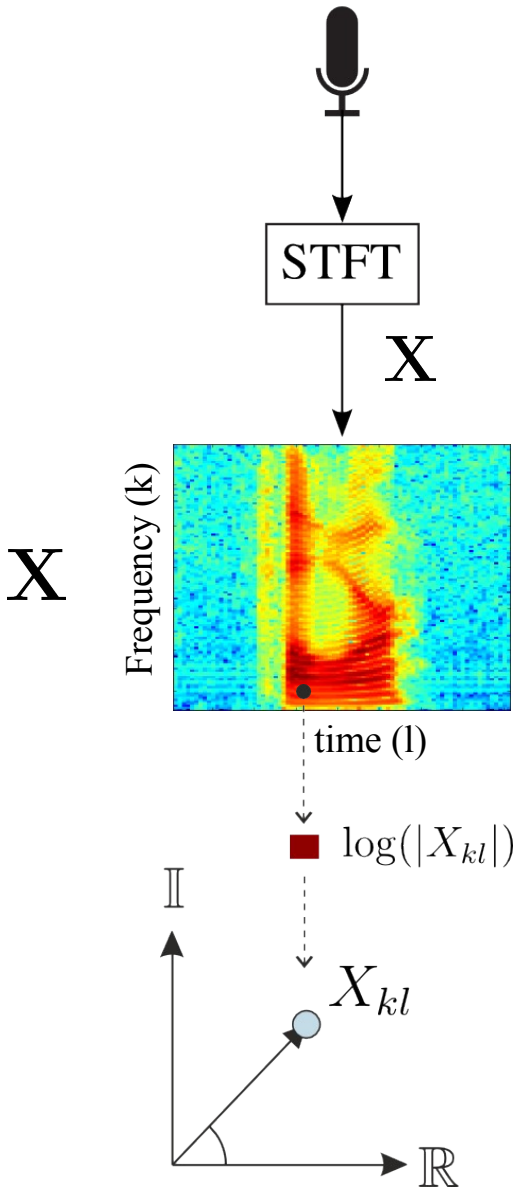
Approaches to Robust ASR

- Uncertainty Handling

  Methods with a common characteristic: Model how much do we know

  Includes both feature and model domain techniques

  Good trade-offs between computational complexity and robustness

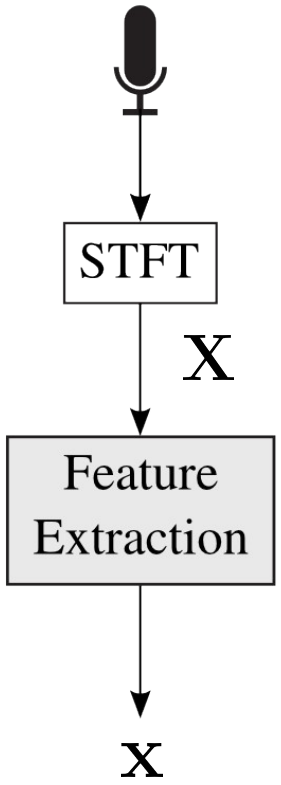


Short-Time Fourier transform (STFT):

- Input signal divided into overlapping analysis frames
- Discrete Fourier transform (DFT) computed for each frame

Properties

- Time-frequency representation
- Linear, invertible transform
- Convolution multiplicative
- Optimal domain for speech processing



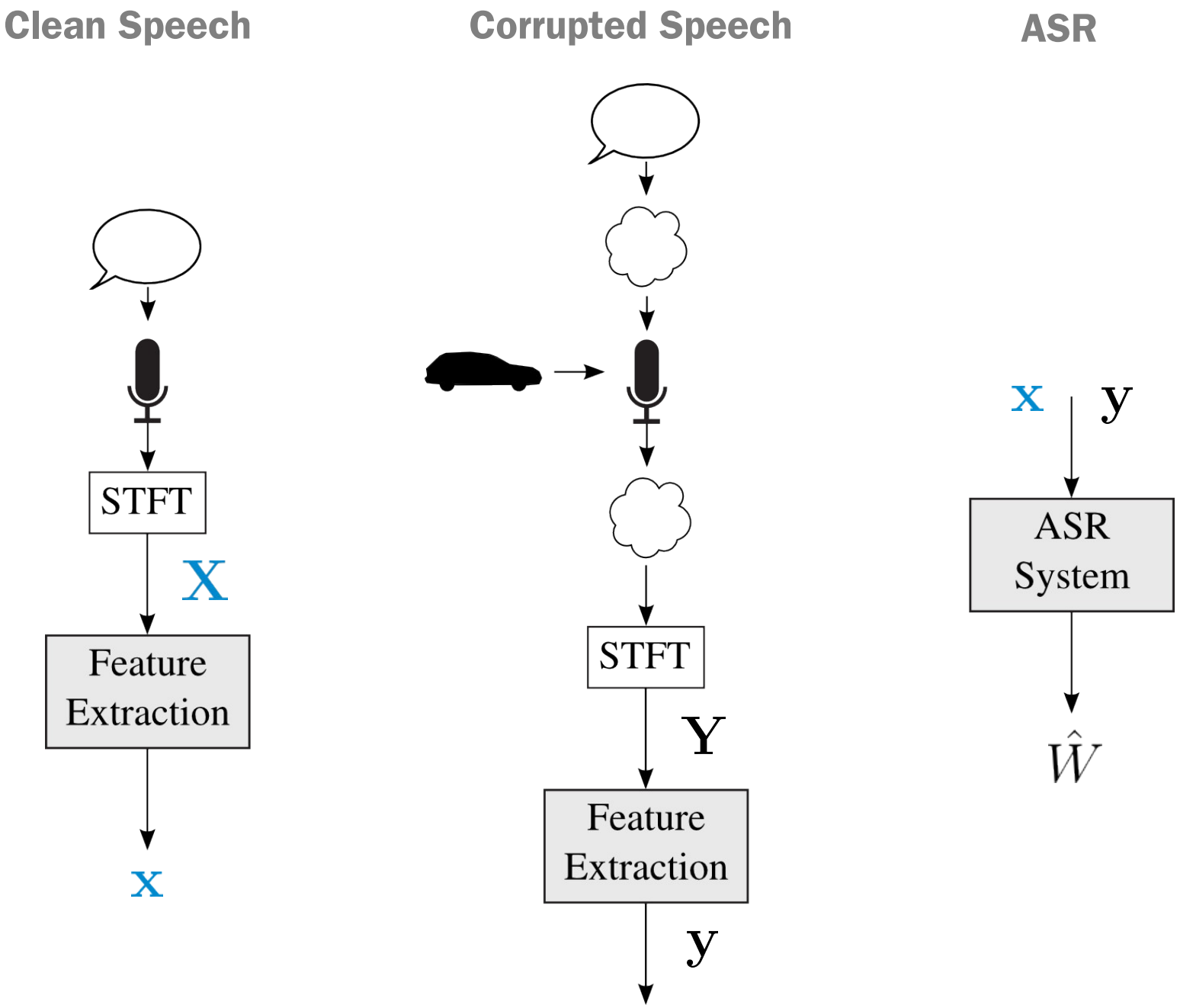
Non-linear (log) feature extractions:

- Inter/Intra-transformations of DFT frames
- MFCCs, RASTA-PLPs

Properties

- Often involve logarithm
- Often non-invertible
- Good compression of acoustic space
- Optimal domain for machine learning of speech

STFT-domain, log-feature-domain and notation



Uncertainty Handling, three main problems

1. Find a probabilistic relation between the unseen clean features $\mathbf{x}$ and the available information (features $\mathbf{y}$, STFT $\mathbf{Y}$, ...)

2. Update ASR model parameters (trained for $\mathbf{x}$) to correctly recognize $(\mathbf{y}, \mathbf{Y}, \ldots)$

3. Train models for clean speech $\mathbf{x}$ from the available information $(\mathbf{y}, \mathbf{Y}, \ldots)$

Well known example of 2, Uncertainty Decoding

- Conventional ASR: Find the most probable sequence of words given $\mathbf{x}$

$$\hat{\mathbf{W}} = \arg\max_{\mathbf{W}}\{p(\mathbf{W}|\mathbf{x})\} = \arg\max_{\mathbf{W}}\{p(\mathbf{x}|\mathbf{W})p(\mathbf{W})\}$$

- Where the acoustic model $p(\mathbf{x}|\mathbf{W})$ is a GMM-HMM and the likelihood of frame l on state q is

$$p(\mathbf{x}_l|q) = \mathcal{N}\left(\boldsymbol{\mu}_q, \boldsymbol{\Sigma}_q\right)$$

- The most probable sequence is then retrieved by the Viterby algorithm or token passing

Well known example of 2, Uncertainty Decoding

- Under observation uncertainty we do not know the clean features $\mathbf{x}$, but have a probabilistic description of its value (obtained through 1)

$$p(\mathbf{x}_l|\mathbf{y}_l) = \mathcal{N}\left(\boldsymbol{\mu}_l^x, \boldsymbol{\Sigma}_l^x\right)$$

- A solution for inference can be attained by integrating out the unseen $\mathbf{x}$

$$\hat{\mathbf{W}} = \arg\max_{\mathbf{W}} \{p(\mathbf{W}|\mathbf{y})\} = \arg\max_{\mathbf{W}} \left\{ \int_{\mathbb{R}^{I \cdot L}} p(\mathbf{W}|\mathbf{x}) p(\mathbf{x}|\mathbf{y}) d\mathbf{x} \right\}$$

- After some approximations this leads to the modified state likelihood

$$\widehat{p(\mathbf{x}_l|q)} = \mathcal{N}\left(\boldsymbol{\mu}_q, \boldsymbol{\Sigma}_q + \boldsymbol{\Sigma}_l^x\right)$$

Tutorial Overview

Log-Feature and Model Domain Approaches to Uncertainty Handling in ASR

[Li Deng, 1h]

Classification of many environment-robust ASR techniques (for handling uncertainty)

(Log) Feature- vs model-domain approaches
Structured vs unstructured approaches
Hybrid feature- and model-domain approaches

Example of structured approach in log-feature and in model domains: Use of a phase sensitive model of speech distortion (1)

Example of hybrid approach in feature-model domain: Noise adaptive training (NAT)

Other uncertainty-handling techniques: NAT extensions

Tutorial Overview

Linear-STFT Domain Approaches to Uncertainty Handling in ASR

[R. F. Astudillo, 1h]

Speech enhancement and uncertainty estimation, in STFT domain (1)

STFT Uncertainty Propagation (1), for

- MFCC features
- RASTA-PLP features
- MLP features

Integration of Speech Enhancement and ASR (2)

Tutorial Overview

Learning from Noisy data

[E. Vincent, 30min]

Bayesian uncertainty estimation for STFT-domain enhancement (1)

Expectation maximization training of acoustic models with unreliable input features (3)

Wrap-up and Perspectives

[E. Vincent, 15min]

Microsoft Research

Log-Feature and Model Domain Approaches to Uncertainty Handling in ASR

Li Deng

Microsoft Research, Redmond

Interspeech Tutorial (Part II; 1hr)

September 2012

Microsoft Research

Outline

- Classification of many environment-robust ASR techniques (for handling uncertainty)
  - (Log) Feature- vs model-domain approaches
  - Structured vs unstructured approaches
- Hybrid feature- and model-domain approaches
- Example of structured approaches: Use of a phase sensitive model of speech distortion
- Example of hybrid approaches: Noise adaptive training (NAT)
- Uncertainty-handling techniques: NAT extensions

Ref: **Li Deng, Front-End, Back-End, and Hybrid Techniques to Noise-Robust Speech Recognition, in D. Kolossa and R. Hab-Umbach (eds.) Robust Speech Recognition of Uncertain Data, pp. 67-99, Springer Verlag, 2011**

Microsoft Research

Taxonomy of Uncertainty Handling Techniques

	Feature Domain	Model Domain	Hybrid
Un-structured	Class F1	Class M1	Class H1
Structured	Class F2	Class M2	Class H2

A summary and classification of noise-robust speech recognition techniques

- Class F1: SPLICE, spectral subtraction, Wiener filter, HMM, MMSE, MMSE-Cep, CMN, CVN, CHN, RASTA, mod spectra;
- Class F2: VTS, Algonquin, phase-model;
- Class M1: MLLR, MAP, C-MLLR, N-CMLLR, multi-style training;
- Class M2: PMC, VTS, phase-model;
- Class H1: NAT-SS, NAT-SPLICE, JAT (NAT-LR), IVN; and
- Class H2: NAT-VTS, UD, JUD.

Feature-Domain vs Model-domain

- Feature-domain approach: feature enhancement (independent of speech classes)
- Model-domain approach: HMM adaptation to the noisy condition
- Hybrid: HMM adaptation to the feature enhanced condition; noise adaptive training; Aurora evaluation paradigm

Microsoft Research

Structured vs Unstructured

- Structured approach: Use of a parametric model of speech distortion
- Distortion model can apply to either feature- or model-domain
- Prominent examples of speech distortion models:
  - vector Taylor series (VTS)
  - Algonquin model
  - phase-sensitive model
- Unstructured approach: no use of speech-distortion models
- Prominent Examples:
  - SPLICE, spectral subtraction (feature enhancement)
  - MLLR, MAP, multi-style training (model adaptation

Example of structured approach:

Use of phase-sensitive model of speech distortion for

1) feature enhancement (log features)

Ref: **Li Deng, Jasha Droppo, and Alex Acero, Enhancement of log Mel power spectra of speech using a phase-sensitive model of the acoustic environment and sequential estimation of the corrupting noise, *IEEE Transactions on Speech and Audio Processing*, vol. 12, no. 2, pp. 133–143, March 2004**

2) model adaptation

Ref: **Jinyu Li, Dong Yu, Li Deng, Yifan Gong, and Alex Acero, A unified framework of HMM adaptation with joint compensation of additive and convolutive distortions, *Computer Speech and Language*, vol. 23, pp. 389-405, 2009**

Microsoft Research

A Phase-Sensitive Model for Speech Distortion

- Clean-speech=x; noise=n; channel=h; noisy-speech=y
- relationship in waveform-sample and DFT:

$$y[t] = x[t] * h[t] + n[t],$$

$$Y[k] = X[k]H[k] + N[k],$$

Instantaneous mixing phase

Relationship in power-spectrum:

$$|Y[k]|^2 = |X[k]|^2|H[k]|^2 + |N[k]|^2 + 2|X[k]H[k]||N[k]|cos\theta_k,$$

- **The last term was usually assumed zero (phase-insensitive), which is correct only in expected sense**

Microsoft Research

Phase-Sensitive Model (cont'd)

- relationship in Mel-filter power spectrum:

$$\sum_k W_k^{(l)} |Y[k]|^2 = \sum_k W_k^{(l)} |X[k]|^2 |H[k]|^2 + \sum_k W_k^{(l)} |N[k]|^2 + 2 \sum_k W_k^{(l)} |X[k]H[k]||N[k]| cos\theta_k,$$

or $$|\tilde{Y}^{(l)}|^2 = |\tilde{X}^{(l)}|^2 |\tilde{H}^{(l)}|^2 + |\tilde{N}^{(l)}|^2 + 2\alpha^{(l)} |\tilde{X}^{(l)}||\tilde{H}^{(l)}||\tilde{N}^{(l)}|,$$

$$\alpha^{(l)} \equiv \frac{\sum_k W_k^{(l)} |X[k]H[k]||N[k]| cos\theta_k}{|\tilde{X}^{(l)}||\tilde{H}^{(l)}||\tilde{N}^{(l)}|}.$$

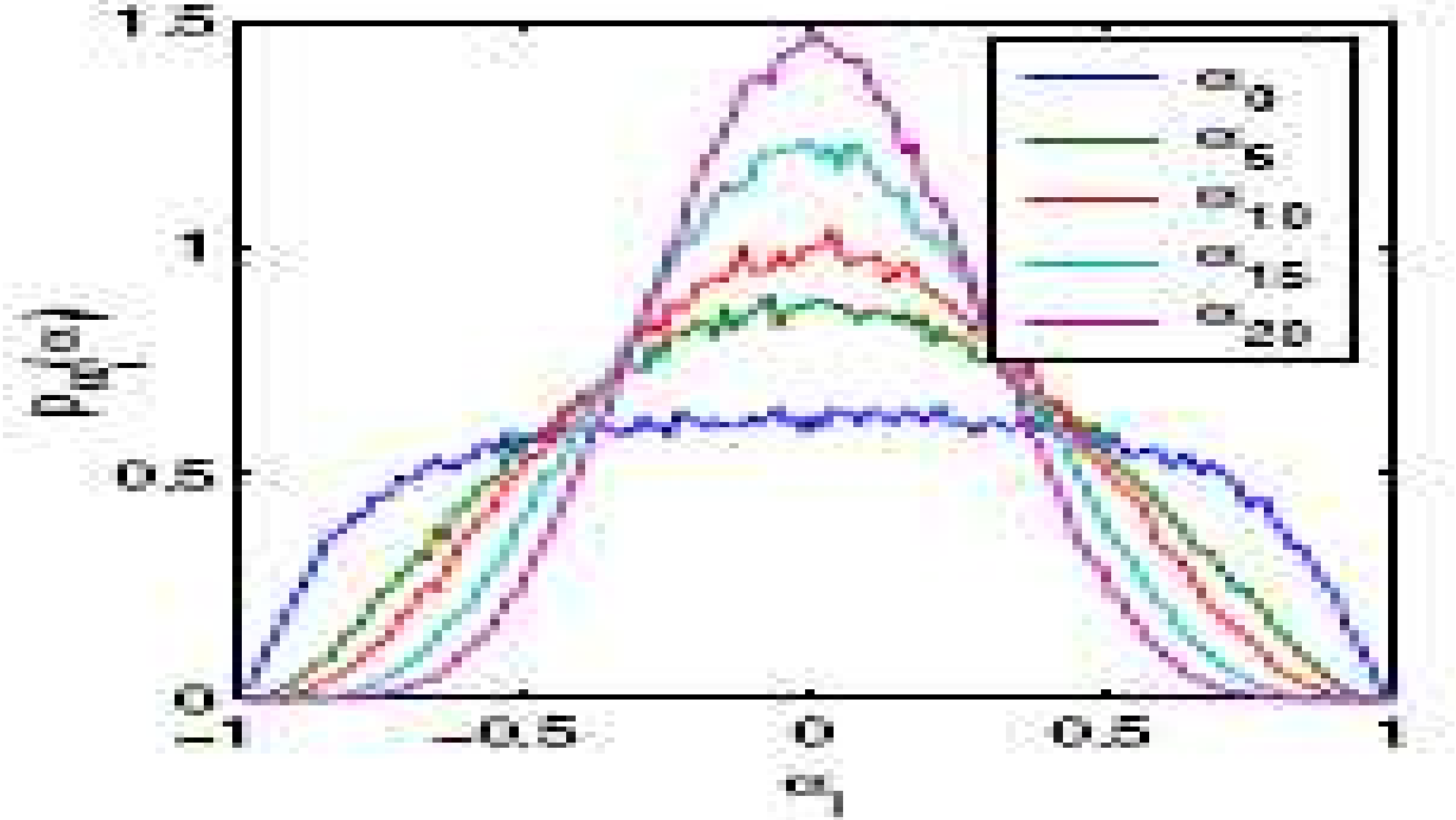
Microsoft Research

Distribution of Phase Factor

(Droppo, Acero, Deng, ICASSP2002)

-- Sum of many uniformly distributed random variables (filter banks)

-- Central limit theorem at work



Microsoft Research

Phase-Sensitive Model (cont'd)

- relationship in log-power-spectrum:

Define log-power-spectrum vectors:

$$\mathbf{y} = \begin{bmatrix} \log|\tilde{Y}^{(1)}|^2 \\ \log|\tilde{Y}^{(2)}|^2 \\ \cdots \\ \log|\tilde{Y}^{(l)}|^2 \\ \cdots \\ \log|\tilde{Y}^{(L)}|^2 \end{bmatrix}, \quad \mathbf{x} = \begin{bmatrix} \log|\tilde{X}^{(1)}|^2 \\ \log|\tilde{X}^{(2)}|^2 \\ \cdots \\ \log|\tilde{X}^{(l)}|^2 \\ \cdots \\ \log|\tilde{X}^{(L)}|^2 \end{bmatrix}, \quad \mathbf{n} = \begin{bmatrix} \log|\tilde{N}^{(1)}|^2 \\ \log|\tilde{N}^{(2)}|^2 \\ \cdots \\ \log|\tilde{N}^{(l)}|^2 \\ \cdots \\ \log|\tilde{N}^{(L)}|^2 \end{bmatrix}, \quad \mathbf{h} = \begin{bmatrix} \log|\tilde{H}^{(1)}|^2 \\ \log|\tilde{H}^{(2)}|^2 \\ \cdots \\ \log|\tilde{H}^{(l)}|^2 \\ \cdots \\ \log|\tilde{H}^{(L)}|^2 \end{bmatrix},$$

then:

$$e^{\mathbf{y}} = e^{\mathbf{x}} \bullet e^{\mathbf{h}} + e^{\mathbf{n}} + 2\alpha \bullet e^{\mathbf{x}/2} \bullet e^{\mathbf{h}/2} \bullet e^{\mathbf{n}/2} = e^{\mathbf{x}+\mathbf{h}} + e^{\mathbf{n}} + 2\alpha \bullet e^{(\mathbf{x}+\mathbf{h}+\mathbf{n})/2}, \quad \textbf{or}$$

$$\mathbf{y} = \log\left[e^{\mathbf{x}+\mathbf{h}} \bullet \left(1 + e^{\mathbf{n}-\mathbf{x}-\mathbf{h}} + 2\alpha \bullet e^{\frac{\mathbf{x}+\mathbf{h}+\mathbf{n}}{2}-\mathbf{x}-\mathbf{h}}\right)\right] = \mathbf{x} + \mathbf{h} + \log\left[1 + e^{\mathbf{n}-\mathbf{x}-\mathbf{h}} + 2\alpha \bullet e^{\frac{\mathbf{n}-\mathbf{x}-\mathbf{h}}{2}}\right]$$

Phase-Sensitive Model (cont'd)

- Gaussian assumption for phase factor

$$p(\alpha^{(l)}) = \mathcal{N}(\alpha^{(l)}; 0, \Sigma_{\alpha}^{(l)}),$$

- Computing conditional prob.:

$$p_y(\mathbf{y}|\mathbf{x}, \mathbf{n}, \mathbf{h}) = |J_{\boldsymbol{\alpha}}(\mathbf{y})|\ p_{\alpha}(\boldsymbol{\alpha}|\mathbf{x}, \mathbf{n}, \mathbf{h}),$$

- Jacobian computation:

$$\mathrm{diag}\left(\frac{\partial \mathbf{y}}{\partial \boldsymbol{\alpha}}\right) = \frac{2e^{\frac{\mathbf{n}-\mathbf{x}-\mathbf{h}}{2}}}{1 + e^{\mathbf{n}-\mathbf{x}-\mathbf{h}} + 2\boldsymbol{\alpha} \bullet e^{\frac{\mathbf{n}-\mathbf{x}-\mathbf{h}}{2}}} = \frac{2e^{\frac{\mathbf{n}+\mathbf{x}+\mathbf{h}}{2}}}{e^{\mathbf{x}+\mathbf{h}} + e^{\mathbf{n}} + 2\boldsymbol{\alpha} \bullet e^{\frac{\mathbf{n}+\mathbf{x}+\mathbf{h}}{2}}} = 2\, e^{\frac{\mathbf{n}+\mathbf{x}+\mathbf{h}}{2}-\mathbf{y}}.$$

- Final result for conditional dependency:

$$p_y(\mathbf{y}|\mathbf{x}, \mathbf{n}, \mathbf{h}) = \frac{1}{2}\ \left|\ \mathrm{diag}\left(e^{\mathbf{y}-\frac{\mathbf{n}+\mathbf{x}+\mathbf{h}}{2}}\right)\right|\ \mathcal{N}\left[\frac{1}{2}(e^{\mathbf{y}-\frac{\mathbf{n}+\mathbf{x}+\mathbf{h}}{2}} - e^{\frac{\mathbf{n}-\mathbf{x}-\mathbf{h}}{2}} - e^{-\frac{\mathbf{n}-\mathbf{x}-\mathbf{h}}{2}}); \mathbf{0}, \boldsymbol{\Sigma}_{\alpha}\right].$$

Microsoft Research

Speech Enhancement Using Phase Modeling (F2)

- After specifying conditional dependency, carry out estimation and inference
- Inference on the clean-speech layer in the Bayes net → speech feature enhancement
- Results (iterative enhancement algorithm):

$$\hat{x} \approx \sum_{m=1}^{M} \gamma_m(x_0, \bar{n}) \left(x_0 - \frac{b_m^{(1)}(x_0, \bar{n})}{b_m^{(2)}(x_0, \bar{n})} \right)$$

(using 2nd-order Taylor series expansion)

Microsoft Research

Noisy Speech Recognition Experiments

(Deng, Droppo, Acero, 2004)

- Aurora 2 noisy speech data
- Using power of true noise (i.e., no est. error)
- Recognition accuracy (%) using enhanced features:

L	1	2	4	7	12
SetA	94.12	96.75	97.96	98.11	98.12
SetB	94.80	97.29	98.10	98.48	98.55
SetC	91.00	94.50	96.50	97.86	98.00
Ave.	93.77	96.52	97.72	98.21	98.27

Microsoft Research

Experiments (cont'd)

Recognition Accuracy	Automatic noise est. algorithm	Assuming no noise est. errors
no phase info (low-fidelity)	**84.80%**	95.90%
phase info (high-fidelity)	**85.74%**	98.27%

--- Much lower relative error reduction when noise estimation errors are introduced
--- Why?

$$|Y[k]|^2 = |X[k]|^2|H[k]|^2 + |N[k]|^2 + 2|X[k]H[k]||N[k]|cos\theta_k,$$

Microsoft Research

Model Adaptation Using Phase Model (M2)

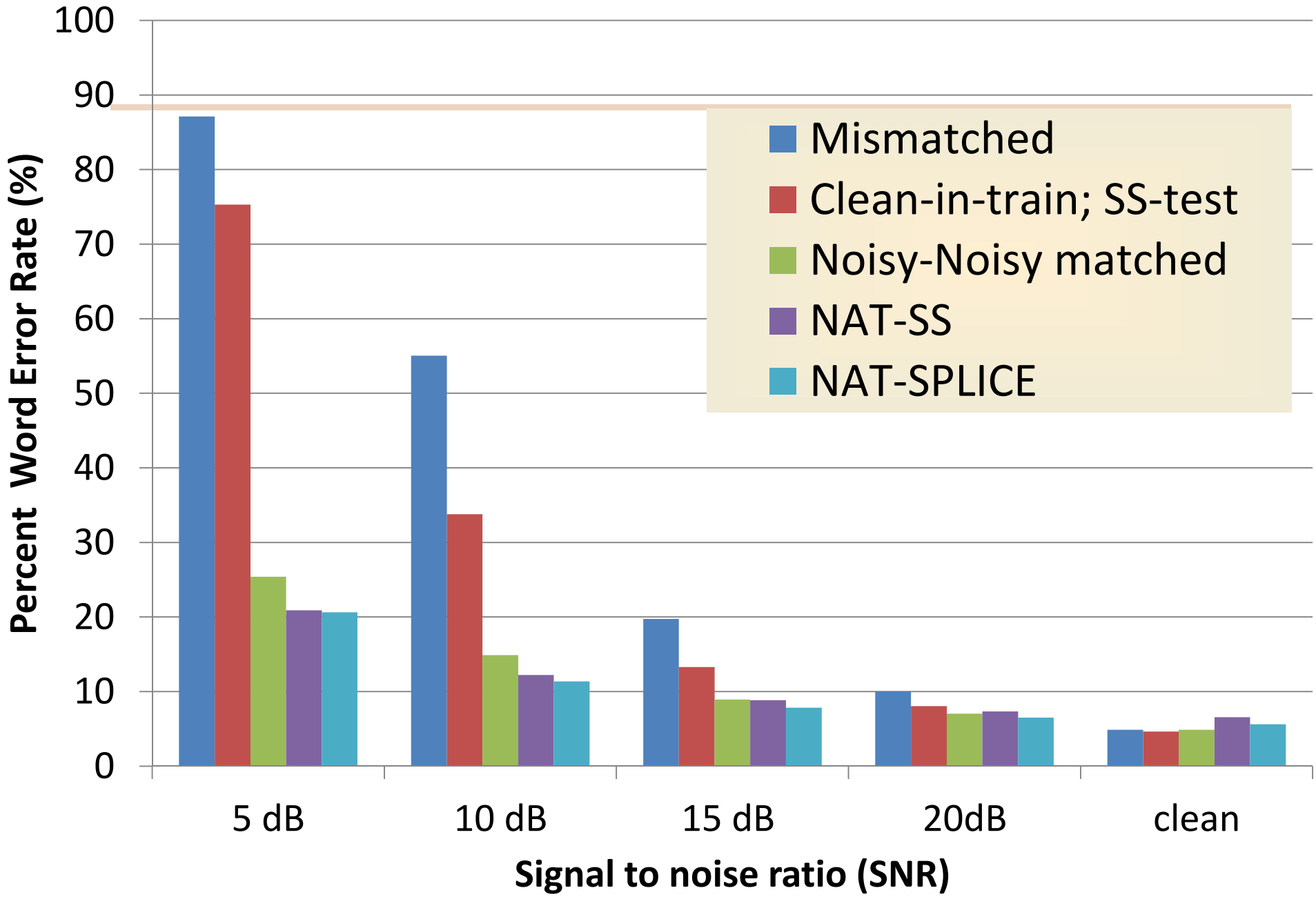
Recognition Accuracy	Automatic noise Est. algorithm	Assuming no noise Est. errors	HMM Adapt (better noise est.)
no phase info	84.80%	95.90%	**91.70%**
phase info	85.74%	98.27%	**93.32%**

Microsoft Research

Noise Adaptive Training

- Prominent example of Hybrid approach
- It performs feature enhancement
- It also modifies HMM parameters using enhanced features
- High performance

Microsoft Research



NAT Varieties & Extensions

- Unstructured (H1): NAT-SS, NAT-SPLICE, NAT-LR, IVN
- Structured (H2): NAT-VTS, uncertainty decoding, joint UD

Summary

- Hundreds of techniques for noise-robust ASR can be classified into 6 classes
- Using two axes: Structured or otherwise; feature, model, or hybrid domains
- H1 or H2 gives best performance, exemplified by noise adaptive training (NAT) with various extensions

Linear-STFT Domain Approaches to Uncertainty Handling in ASR

Ramón Fernandez Astudillo - Laboratorio de Lengua Falada (L2f) INESC-ID Lisboa

ramon@astudillo.com

This work was partially funded by the Portuguese Foundation for Science and Technology (FCT) through the grant number SFRH/BPD/68428/2010 and the project PEst-OE/EEI/LA0021/2011

Overview

- STFT-Speech enhancement and residual uncertainty
  - The complex Gaussian uncertainty model
  - Residual uncertainty estimation (Empirical/MSE)

- STFT Uncertainty Propagation
  - Mel-Frequency Cepstral Coefficients
  - RASTA-Perceptual-Linear-Prediction
  - Multi-Layer Perceptron

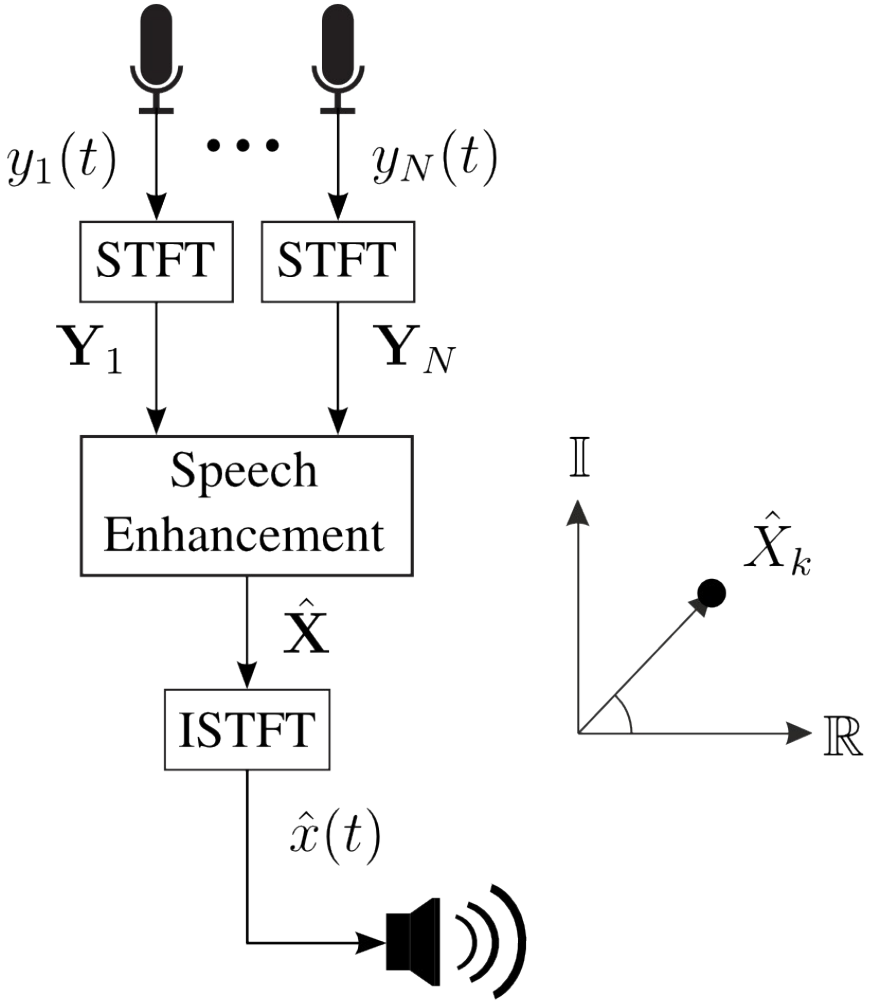
- Integration of STFT speech enhancement and robust ASR
  - Uncertainty Propagation as MMSE estimator
  - Uncertainty Propagation & Decoding
  - Experiments and Results

Overview

- **STFT-Speech enhancement and residual uncertainty**
  - The complex Gaussian uncertainty model
  - Residual uncertainty estimation (Empirical/MSE)

- STFT Uncertainty Propagation
  - Mel-Frequency Cepstral Coefficients
  - RASTA-Perceptual-Linear-Prediction
  - Multi-Layer Perceptron

- Integration of STFT speech enhancement and robust ASR
  - Uncertainty Propagation as MMSE estimator
  - Uncertainty Propagation & Decoding
  - Experiments and Results

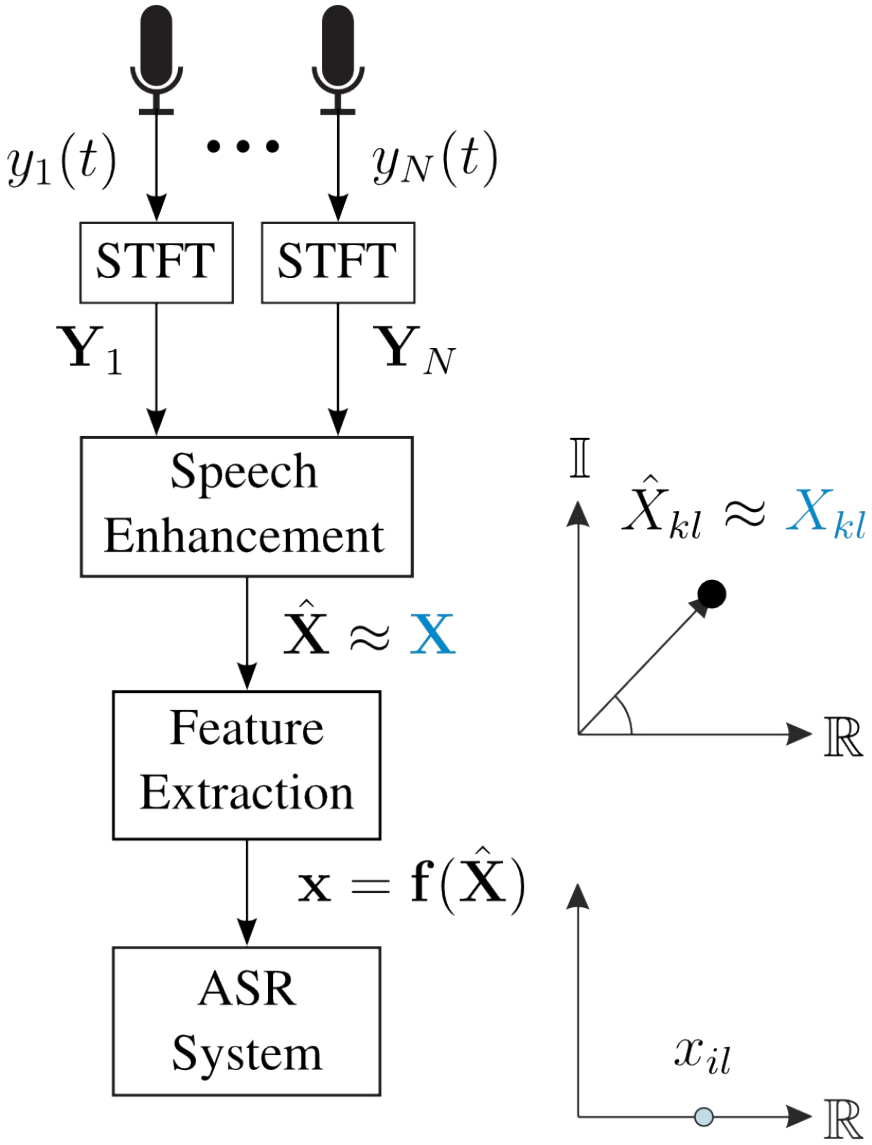


STFT Domain Speech Enhancement:

- Time (l)-frequency (k) representation of speech
- Speech corruption easy to model e.g.

  Aditivity of sources
  Reverberation (early, late)
  Cocktail Party Problem

- Active research field e.g hearing aids, telecommunications
- Targets humans, but usable for ASR



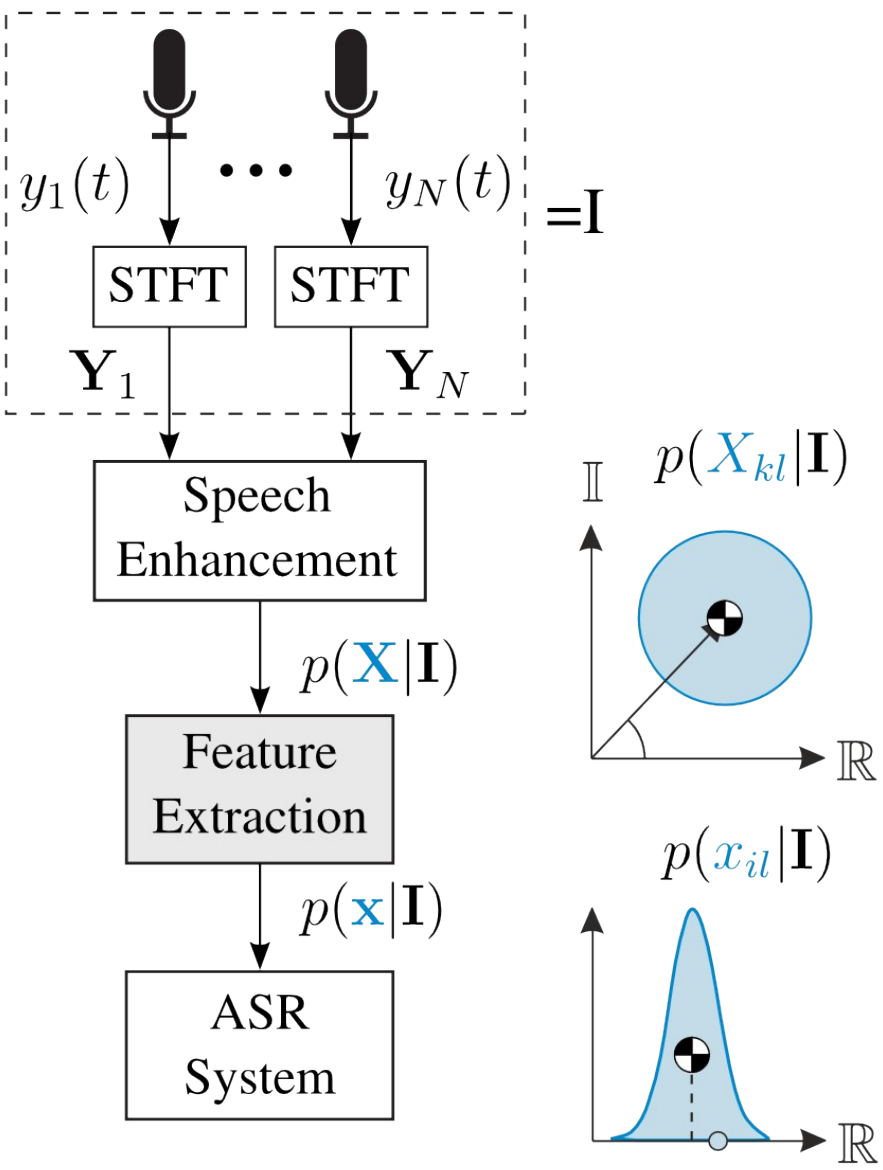
STFT Domain Speech Enhancement:

- Provides a wide range of methods for robust ASR

  Single Channel
  - Additive noise suppression
  - Late reverberation supression
  - ...

  Multichannel
  - Blind source separation
  - Beamforming
  - ...

- But integration poor (only point-estimate passed)

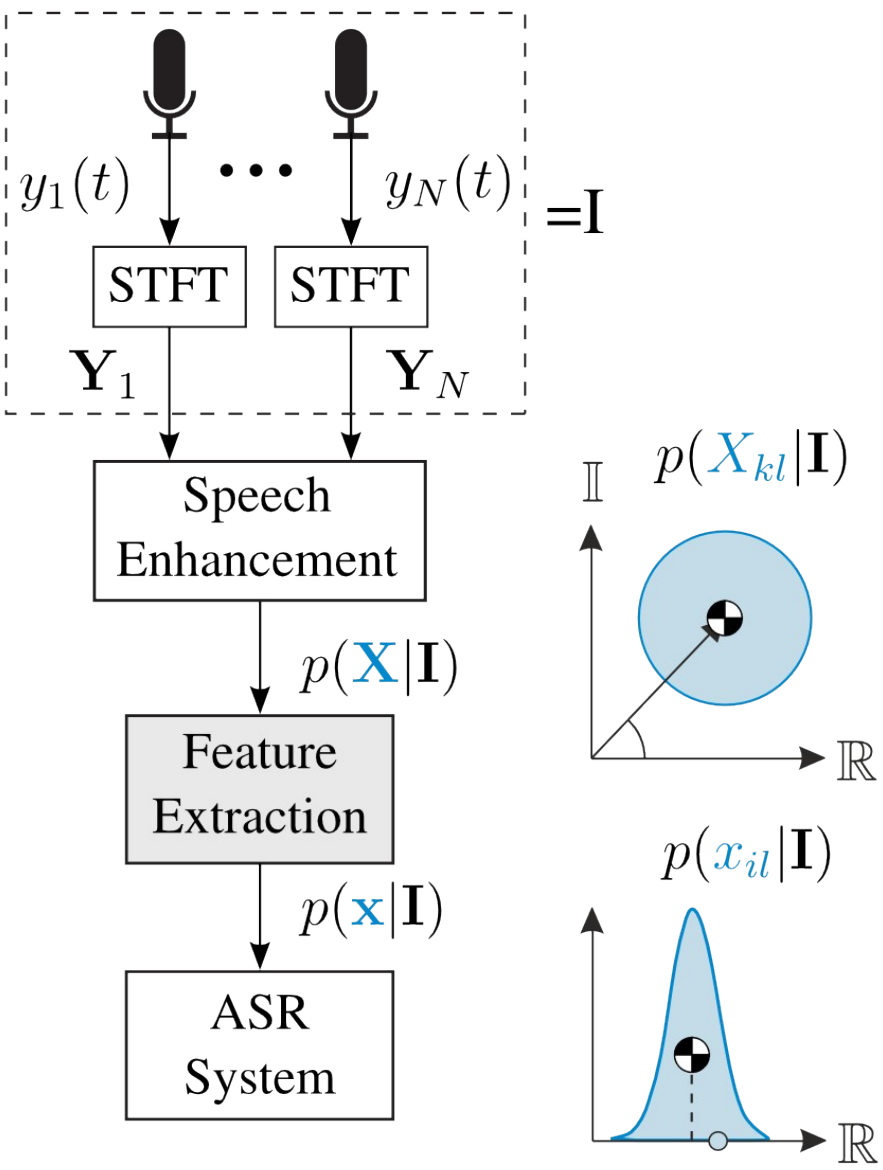


STFT Uncertainty Propagation:

- Uncertainty of estimation:

  “It is not possible to completely determine the clean signal from **I**”

- Only model possible is probabilitic
- Posterior distribution of each clean Fourier coefficient given **I**
- Feature extraction results in a posterior of the clean features given **I**



STFT Uncertainty Propagation:

- STFT-Uncertainty Propagation

  “Transform an uncertain description of the STFT into feature domain”

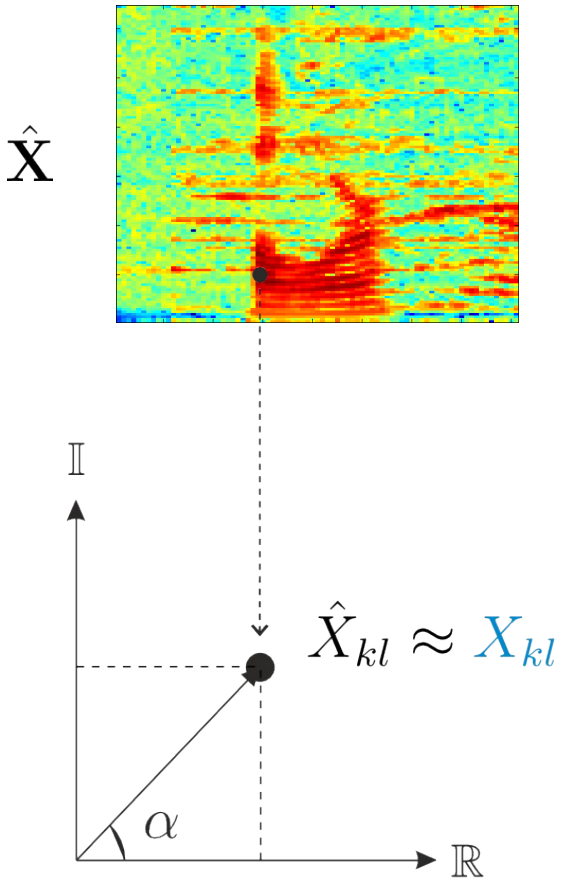
- Antecedents

  Constrained Gaussian model
  [Kolossa 2005]

  GMM model of spectral amplitude
  [Srinivasan 2006]

- Complex Gaussian Uncertainty Model
  [Astudillo 2010c, Astudillo 2011]

Estimated Clean Spectrum



Circularly symmetric Complex Gaussian Model

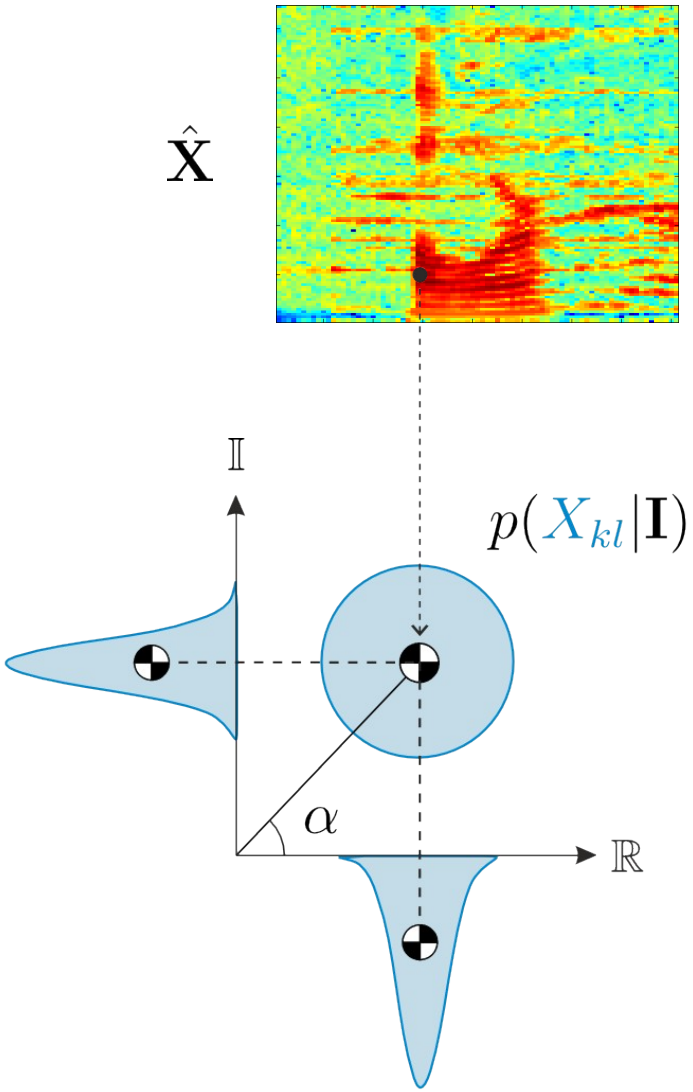
- Real, imaginary components

  Gaussian distributed
  Uncorrelated
  Same variance

$$p(X_{kl}|\mathbf{I}) = \mathcal{N}_{\mathbb{C}}\left(\hat{X}_{kl}, \lambda_{kl}\right)$$

- Mean equal to speech enhancement estimation
- Variance equal to uncertainty
- But how to compute uncertainty?

Uncertain Clean Spectrum



Circularly symmetric Complex Gaussian Model

- Real, imaginary components

  Gaussian distributed
  Uncorrelated
  Same variance

$$p(X_{kl}|\mathbf{I}) = \mathcal{N}_{\mathbb{C}}\left(\hat{X}_{kl}, \lambda_{kl}\right)$$

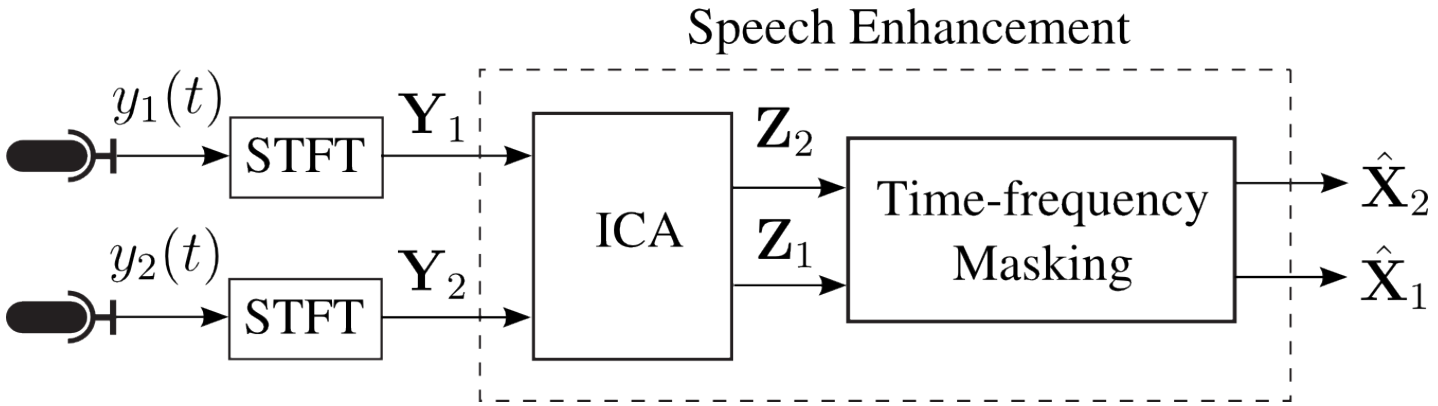
- Mean equal to speech enhancement estimation
- Variance equal to uncertainty
- But how to compute uncertainty?

Empirical Uncertainty Estimation

- Uncertainty as function of the amount of change at given signal processing stage
- The bigger the change, the bigger the uncertainty
- *Ad hoc* solution, not very elegant but effective
- Valid with any method, does not depend on complexity

Empirical Uncertainty Estimation

- Example, post-processing of blind source separation [Kolossa 2010]



- Uncertainty assumed proportional to change at time-frequency masking step

$$\lambda_{kl}^1 = \alpha \left| |Z_{kl}^1| - |\hat{X}_{kl}^1| \right|^2$$

- Additional parameter trained from examples

Residual Uncertainty in Minimum Mean Square Error (MMSE) speech enhancement methods

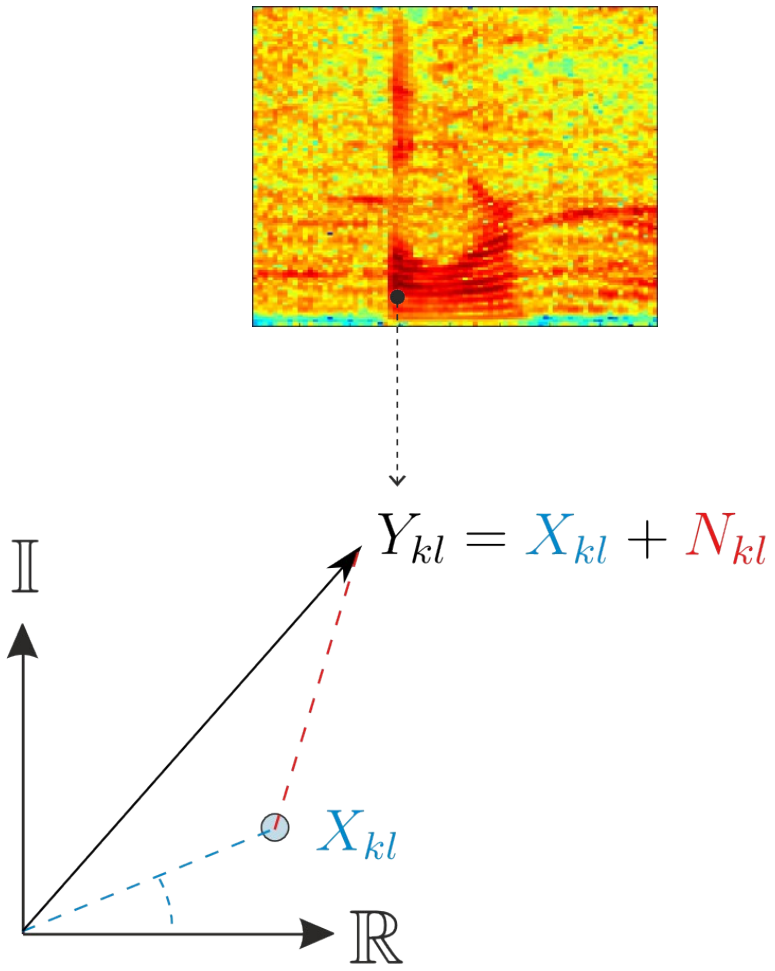
Well known methods

- Wiener filter (MMSE Fourier coefficient estimator)
- Ephraim-Malah filters (MMSE of amplitude, log-amplitude)
- ...

Wide range of supression techniques and implementations

- Additive noise
- Late reverberation
- Channel-decorrelated noise
- Post-Processing (NMF, ICA, Beamforming)
- ...

Complex Gaussian Model of Speech Distortion



- *A priori* models assumed for speech and distortion

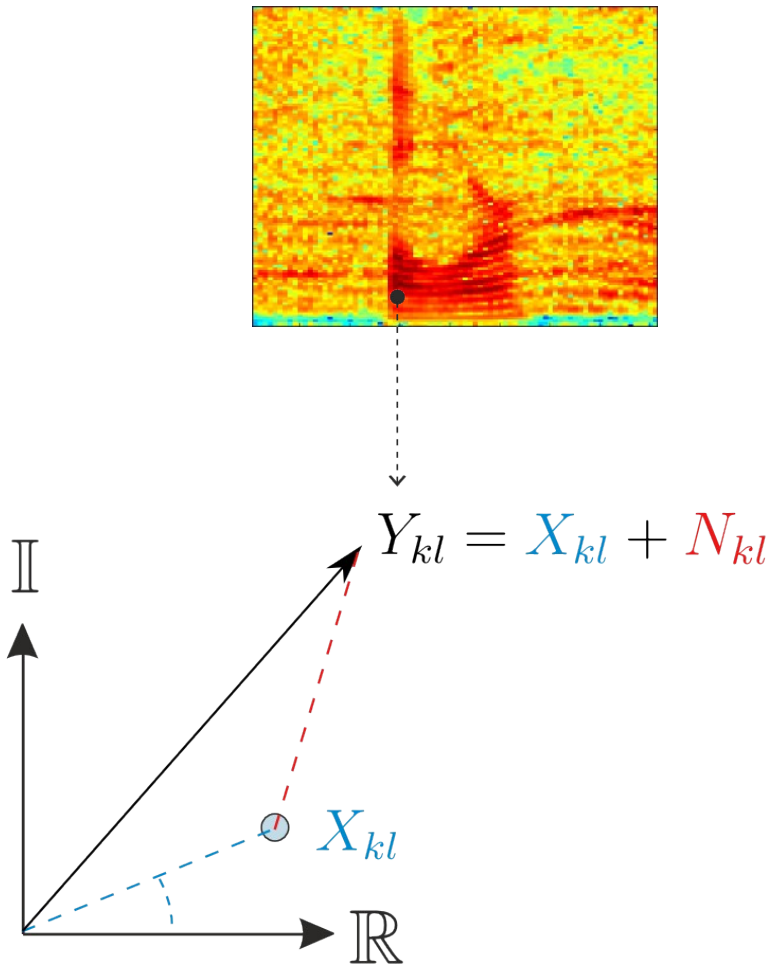
$$p(X_{kl}) = \mathcal{N}_{\mathbb{C}}\left(0, \hat{\lambda}_X\right)$$

$$p(N_{kl}) = \mathcal{N}_{\mathbb{C}}\left(0, \hat{\lambda}_D\right)$$

- This leads to the likelihood

$$p(Y_{kl}|X_{kl}) = \mathcal{N}_{\mathbb{C}}\left(X_{kl}, \hat{\lambda}_D\right)$$

Complex Gaussian Model of Speech Distortion



- *A priori* models assumed for speech and distortion

$$p(X_{kl}) = \mathcal{N}_{\mathbb{C}}\left(0, \hat{\lambda}_X\right)$$

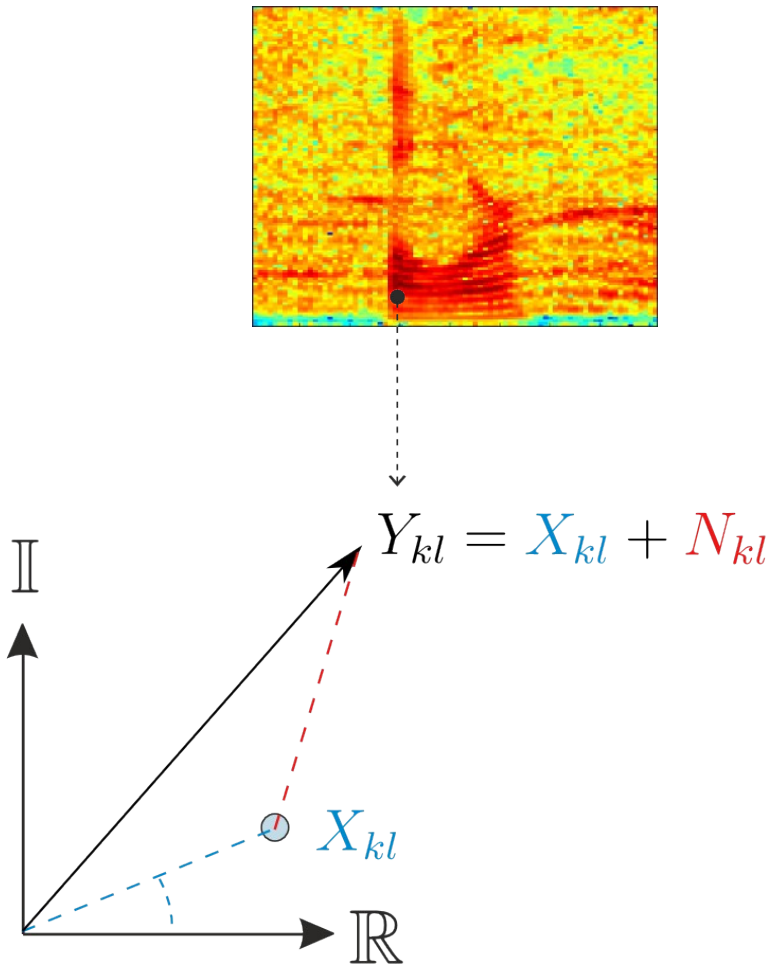
$$p(N_{kl}) = \mathcal{N}_{\mathbb{C}}\left(0, \hat{\lambda}_D\right)$$

- This leads to the likelihood

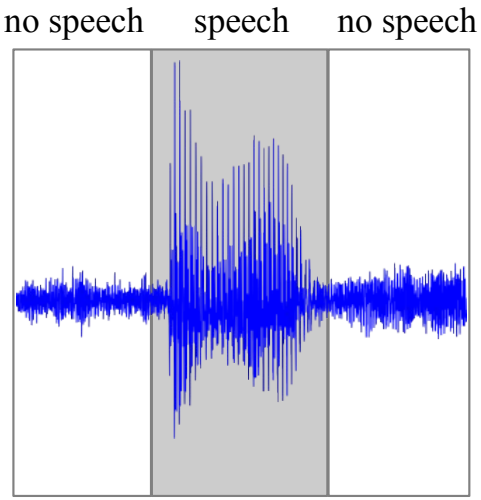
$$p(Y_{kl}|X_{kl}) = \mathcal{N}_{\mathbb{C}}\left(X_{kl}, \hat{\lambda}_D\right)$$

- But how to compute the variances?

Complex Gaussian Model of Speech Distortion



- Variances determined e.g. from voice activity detection and stationarity



- Other alternatives

  Spatial information (beamforming)
  Statistical independence (ICA)
  Late reverberation models

Wiener filter: MMSE estimation of each Fourier coefficient

$$\hat{X}_{kl}^{\text{MMSE}} = \arg\min_{\hat{X}_{kl}} \left\{ E\left\{ \left\| X_{kl} - \hat{X}_{kl} \right\|^2 \right\} \right\}$$

The cost function is the expected error with respect to observable and hidden variables

$$E\left\{ \left\| X_{kl} - \hat{X}_{kl} \right\|^2 \right\} = \int_{\mathbb{C}} \int_{\mathbb{C}} \left\| X_{kl} - \hat{X}_{kl} \right\|^2 p(Y_{kl}, X_{kl}) dY_{kl} dX_{kl}$$

The solution is the expectation of the posterior distribution

$$\hat{X}_{kl}^{\text{MMSE}} = E\{X_{kl} | Y_{kl}\}$$

Wiener filter: MMSE estimation of each Fourier coefficient

$$\hat{X}_{kl}^{\text{MMSE}} = E\{X_{kl}|Y_{kl}\} = \int_{\mathbb{C}} X_{kl} p(X_{kl}|Y_{kl}) dX_{kl} = \frac{\hat{\lambda}_{X_{kl}}}{\hat{\lambda}_{X_{kl}} + \hat{\lambda}_{D_{kl}}} Y_{kl}$$

where the posterior is attained through Bayes

$$p(X_{kl}|Y_{kl}) = \frac{p(Y_{kl}|X_{kl})p(X_{kl})}{\int_{\mathbb{C}} p(Y_{kl}|X_{kl})p(X_{kl})dX_{kl}} = \frac{p(Y_{kl}|X_{kl})p(X_{kl})}{p(Y_{kl})}$$

Wiener filter: MMSE estimation of each Fourier coefficient

$$\hat{X}_{kl}^{\text{MMSE}} = E\{X_{kl}|Y_{kl}\} = \int_{\mathbb{C}} X_{kl} p(X_{kl}|Y_{kl}) dX_{kl} = \frac{\hat{\lambda}_{X_{kl}}}{\hat{\lambda}_{X_{kl}} + \hat{\lambda}_{D_{kl}}} Y_{kl}$$

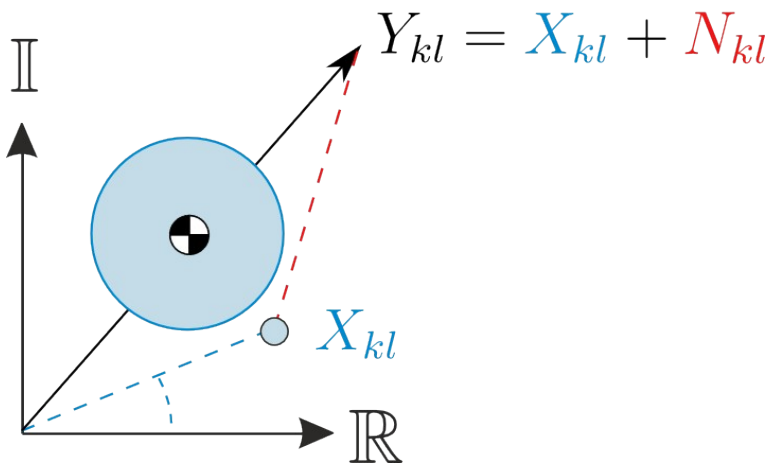
where the posterior is attained through Bayes

$$p(X_{kl}|Y_{kl}) = \frac{p(Y_{kl}|X_{kl})p(X_{kl})}{\int_{\mathbb{C}} p(Y_{kl}|X_{kl})p(X_{kl})dX_{kl}} = \frac{p(Y_{kl}|X_{kl})p(X_{kl})}{p(Y_{kl})}$$

We can use this posterior as an uncertain description of the signal [Astudillo 2009]

$$p(X_{kl}|Y_{kl}) = \mathcal{N}_{\mathbb{C}}\left(\hat{X}_{kl}^{\text{MMSE}}, \lambda_{kl}\right)$$

$$p(X_{kl}|Y_{kl})$$



- Same model as in the empirical uncertainty case
- Variance derived on solid mathematical grounds

The uncertainty in this model is equal to the residual MSE

$$\text{MSE} = \int_{\mathbb{C}} \int_{\mathbb{C}} \left\| X_{kl} - \hat{X}_{kl}^{\text{MMSE}} \right\|^2 p(X_{kl}, Y_{kl}) dX_{kl} dY_{kl} = \lambda_{kl} = \frac{\hat{\lambda}_{X_{kl}} \hat{\lambda}_{D_{kl}}}{\hat{\lambda}_{X_{kl}} + \hat{\lambda}_{D_{kl}}}$$

This error assumes *a priori* information is perfect (ignores variance errors)!

Overview

- STFT-Speech enhancement and residual uncertainty
  - The complex Gaussian uncertainty model
  - Residual uncertainty estimation (Empirical/MSE)

- **STFT Uncertainty Propagation**
  - Mel-Frequency Cepstral Coefficients
  - RASTA-Perceptual-Linear-Prediction
  - Multi-Layer Perceptron

- Integration of STFT speech enhancement and robust ASR
  - Uncertainty Propagation as MMSE estimator
  - Uncertainty Propagation & Decoding
  - Experiments and Results

Short-time Fourier Transform Uncertainty Propagation (STFT-UP)

> “Transforming a complex random matrix (uncertain spectrum) through a non-linear, multivariate, feature extraction”

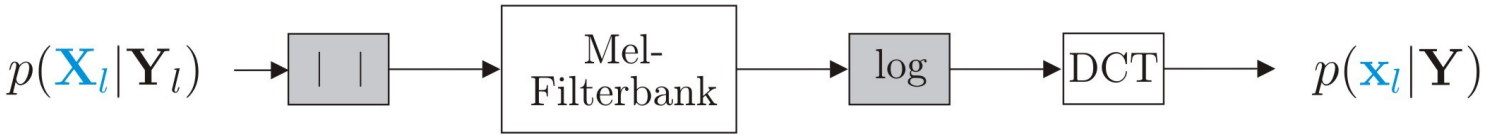
- Can be attained in multiple ways
  - Closed form (variable change)
  - Numerical approximation (e.g. Monte Carlo)
- Closed forms (if available) a better option for ASR (10ms ~ 256 random var.)

Short-time Fourier Transform Uncertainty Propagation (STFT-UP)

- Uses a combination of closed-form and pseudo-Monte Carlo methods
- Feature extraction divided into different steps (e.g amplitude, logarithm)
- Optimal method used for each step
- We need to know uncertainty distribution between steps
- Here recipes presented for

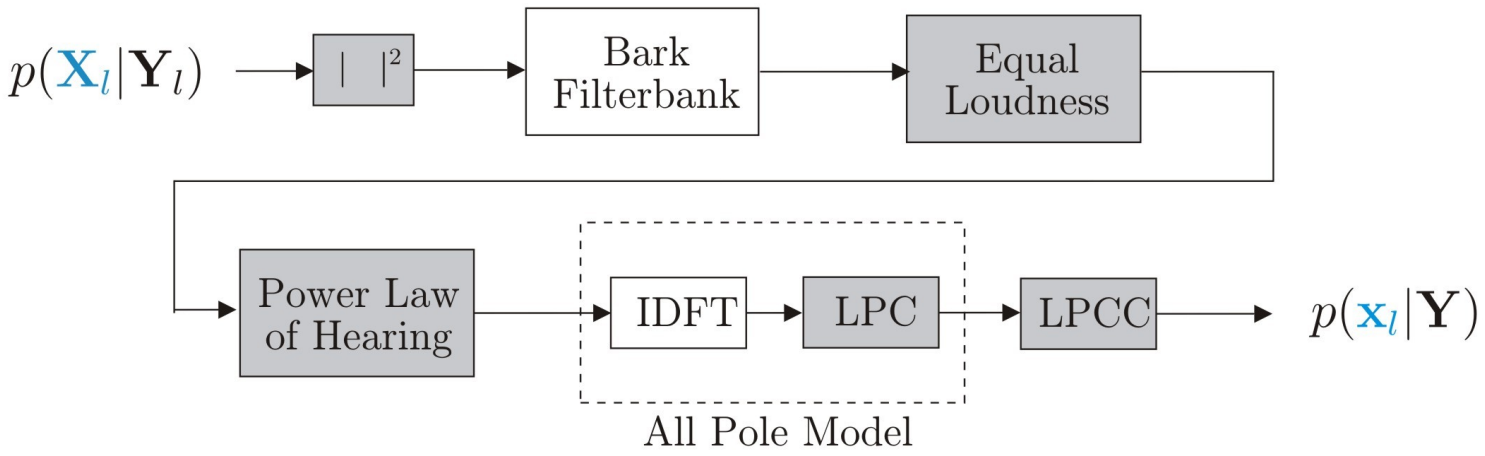
  Mel-Frequency Cepstral Coefficients
  RASTA-Perceptual-Linear-Prediction
  Multi-layer perceptron features

Mel-frequency Cepstral Coefficients (MFCCs)



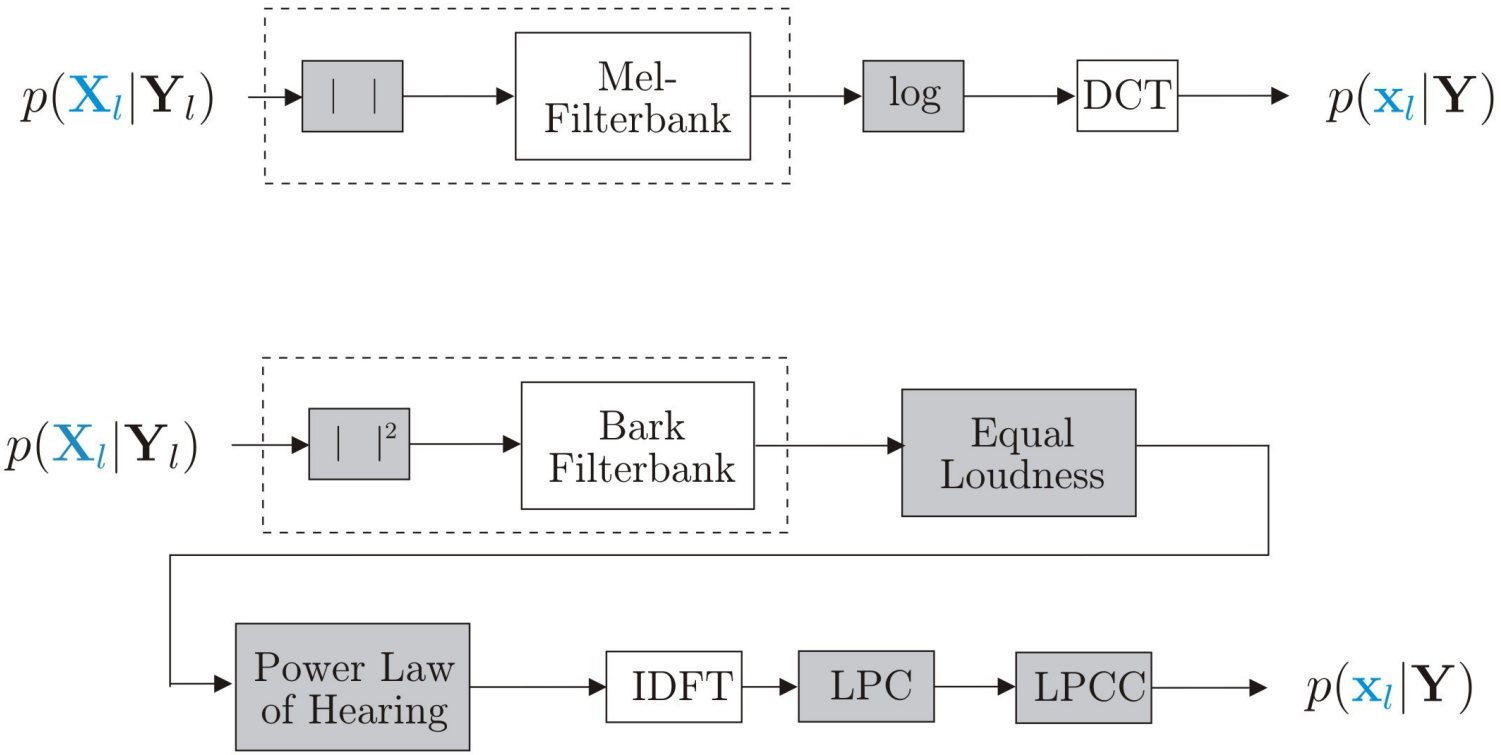
- Extract Amplitude from STFT
- Log-spaced filterbank, perceptually motivated (Linear, non-invertible)
- Logarithm
- Discrete cosine transform (Linear)

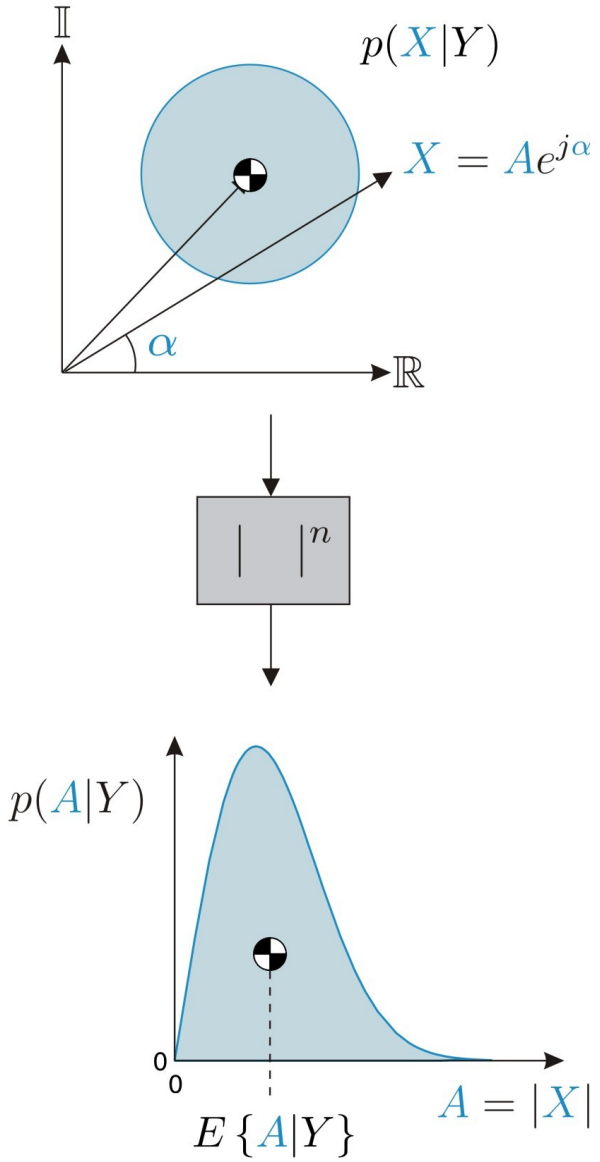
Perceptual Linear Prediction Cepstra (LPCCs)



- Squared amplitude from STFT
- Log-spaced filterbank, perceptually motivated (Linear, non-invertible)
- Equal loudness, power law of hearing (exponentiation)
- All-pole model (Levinson-Durbin recursion)
- LPCs to ceptral coefficients

Common step: N-th power of Magnitude + Linear filterbank





Powers of Magnitude

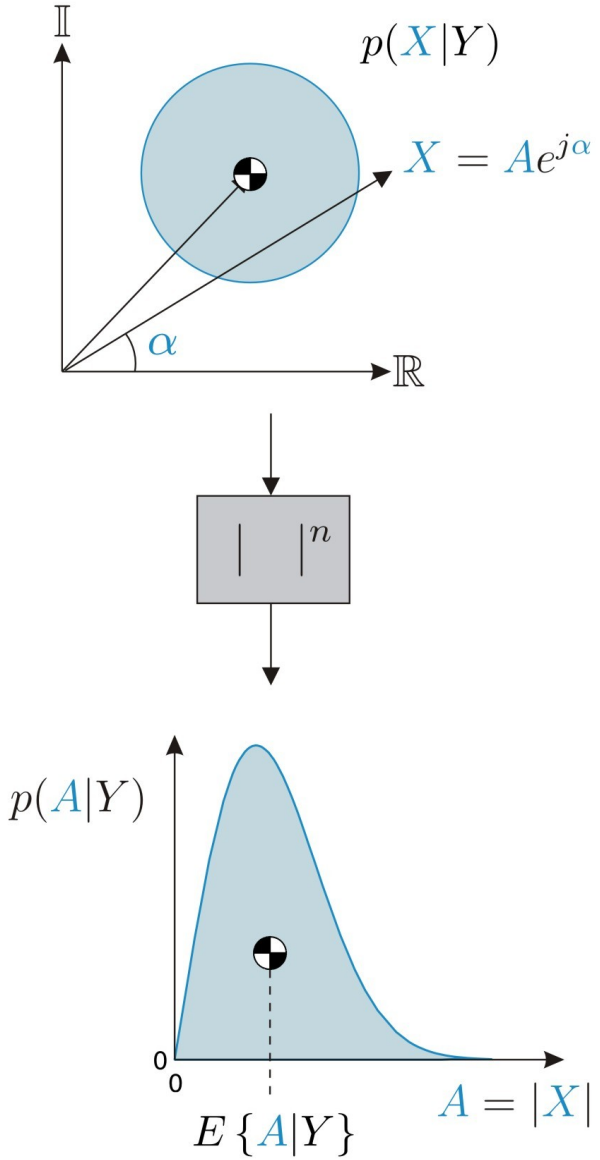
Uncertainty distribution in STFT domain

$$p(X_{kl}|Y_{kl}) = \mathcal{N}_{\mathbb{C}}\left(\hat{X}_{kl}, \lambda_{kl}\right)$$

We can integrate out the phase to get the amplitude

$$p(A|Y) = \int_0^{2\pi} p(Ae^{j\alpha}|Y)Ad\alpha$$

This results in a Rice distribution (n=1)



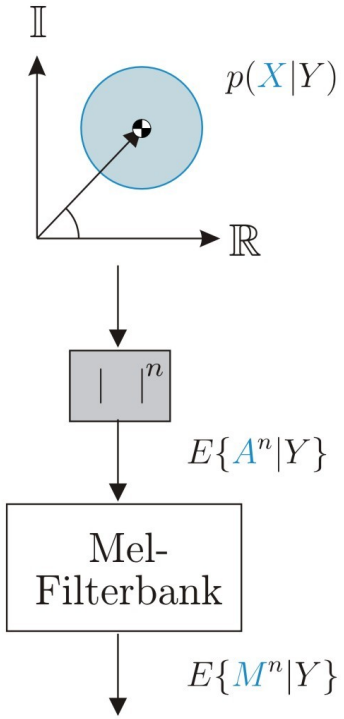
Powers of Magnitude

We can not compute the distribution, of the n-th power of the magnitude

Fortunately, the n-th moment of the amplitude can be computed

$$E\{A^n|Y\} = \int_0^\infty A^n p(A|Y)dA$$

$$= \Gamma\left(\frac{n}{2}+1\right)\lambda^{\frac{n}{2}}\Phi\left(-\frac{n}{2};1;-\frac{|\hat{X}|^2}{\lambda}\right)$$

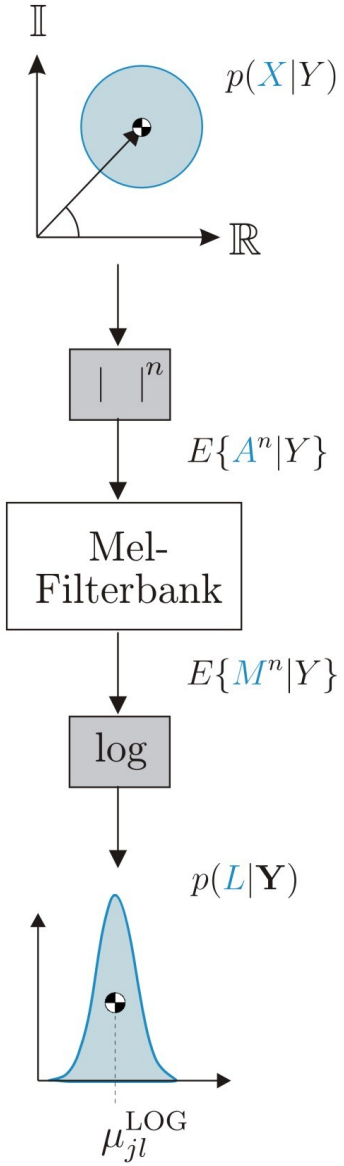


+Mel/Bark Filterbanks

- We can compute the nth moment
- Linear transform, no problem
- Dimension reduced ~10 times
- Induces feature correlation!

Next transformations

- MFCC (logarithm)
- LPCC (log + various non-linear transf.)

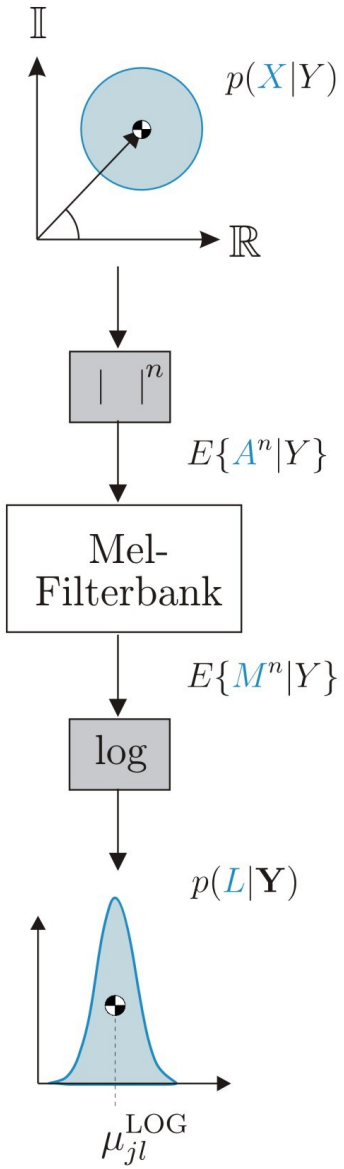


Logarithm

For n=2 (squared magnitude)

- Distribution approx. Log-normal
- Log-features then Gaussian
- Many feature extractions linear on log domain

DCT (completing MFCCs)
RASTA (IIR)
CMS
Deltas, Accelerations
Blind Equalization
LDA



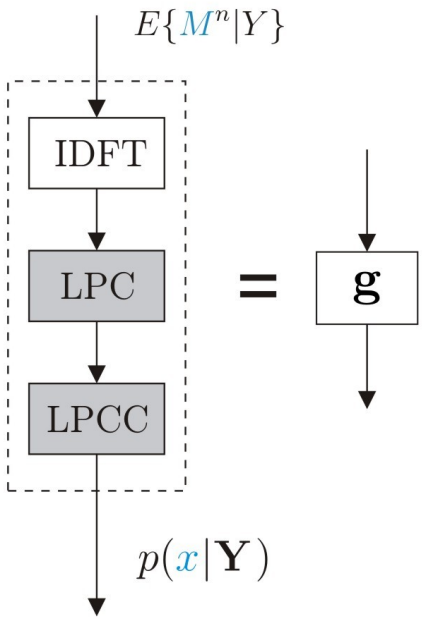
Logarithm

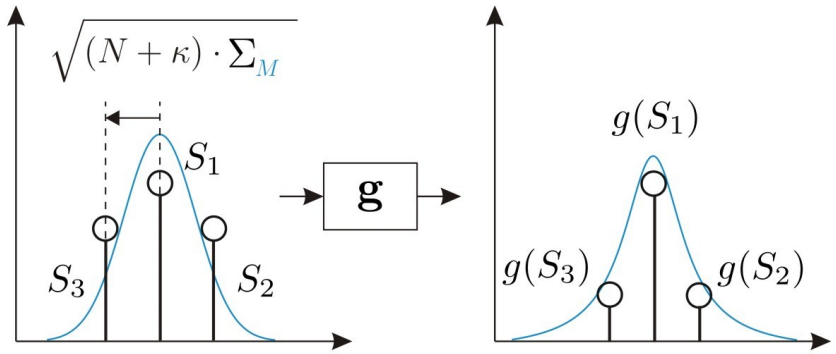
What if n not 2?

- Unscented Tranform, or
- Cumulant generating function (CGF)

$$\log(E\{\exp(nL)|Y\}) = \sum_{p=1}^{\infty} K_p^{L|Y} \frac{n^p}{p!}$$

- Taylor approximation p=2, has the same effect as log-normality
- Stronger assumption, no Gaussianity garanteed (DCT helps)

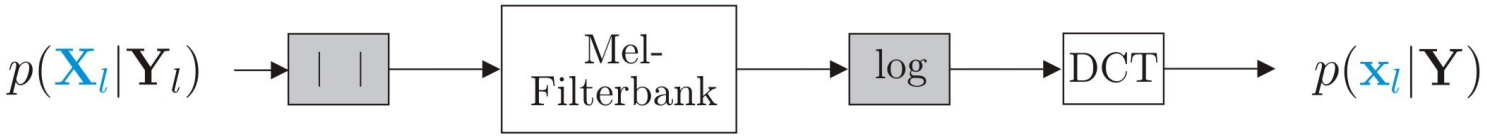




Generic Non-Linear Transforms

- Feature dimension (N) reduced after filterbank ~1 order of magnitude
- If uncertainty not too skewed...
- The unscented transform yields good results
- Approx. discrete distribution of 2N+1 points
- ~45 Transformations trough **g** needed

Example: MFCCs with STFT-UP



- Usually we would compute MFCCs as $x_{il} = \sum_{j=1}^{J} T_{ij} \log \left(\sum_{k=1}^{K} W_{jk} |X_{kl}|^2 \right)$
- But in the real world we only have $p(X_{kl}|Y_{kl}) = \mathcal{N}_{\mathbb{C}} \left(\hat{X}_{kl}, \lambda_{kl} \right)$
- Applying STFT-UP (diagonal cov.) we get a posterior $p(x_{il}|\mathbf{Y}) = \mathcal{N} \left(\mu_{il}^x, \Sigma_{iil}^x \right)$ with parameters:

Example: MFCCs with STFT-UP

$$\mu_{il}^{x} \approx \sum_{j=1}^{J} T_{ij} \log\left(\sum_{k=1}^{K} W_{jk}\left(|\hat{X}_{kl}|^2 + \lambda_{kl}\right)\right) - \sum_{j=1}^{J} T_{ij} K_2^{L_{jl}|\mathbf{Y}}$$

$$\Sigma_{iil}^{x} \approx \sum_{j=1}^{J} T_{ij}^2 K_2^{L_{jl}|\mathbf{Y}_l}$$

with

$$K_2^{L_{jl}|\mathbf{Y}} \approx \log\left(\sum_{k=1}^{K} W_{jk}^2\left(\lambda_{kl}^2 + |\hat{X}_{kl}|^4 + 4\lambda_{kl}|\hat{X}_{kl}|^2\right)\right) - 2\log\left(\sum_{k=1}^{K} W_{jk}\left(|\hat{X}_{kl}|^2 + \lambda_{kl}\right)\right)$$

Example: MFCCs with STFT-UP

$$\mu_{il}^{x} \approx \sum_{j=1}^{J} T_{ij} \log \left(\sum_{k=1}^{K} W_{jk} \left(|\hat{X}_{kl}|^2 + \lambda_{kl} \right) \right) - \sum_{j=1}^{J} T_{ij} K_2^{L_{jl}|\mathbf{Y}}$$

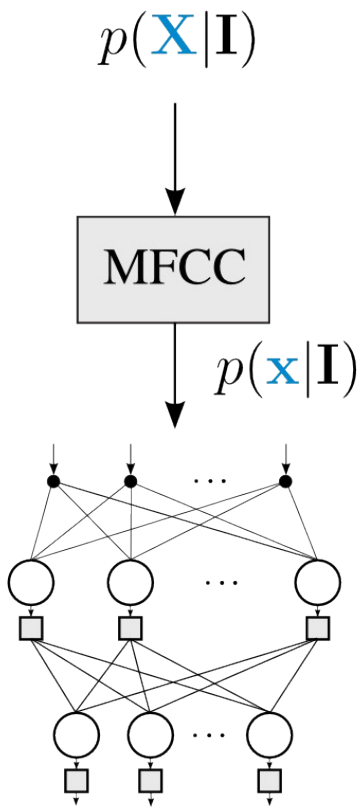
$$\Sigma_{iil}^{x} \approx \sum_{j=1}^{J} T_{ij}^2 K_2^{L_{jl}|\mathbf{Y}_l}$$

with

$$K_2^{L_{jl}|\mathbf{Y}} \approx \log \left(\sum_{k=1}^{K} W_{jk}^2 \left(\lambda_{kl}^2 + |\hat{X}_{kl}|^4 + 4\lambda_{kl}|\hat{X}_{kl}|^2 \right) \right) - 2 \log \left(\sum_{k=1}^{K} W_{jk} \left(|\hat{X}_{kl}|^2 + \lambda_{kl} \right) \right)$$

Computational cost around twice that of conventional MFCCs (plus noise estimation)

Multi-Layer Perceptrons (MLPs)



- As Feature Extraction (uncertainty Propagation)

  Bottleneck Features
  Multi-stream TANDEM approach

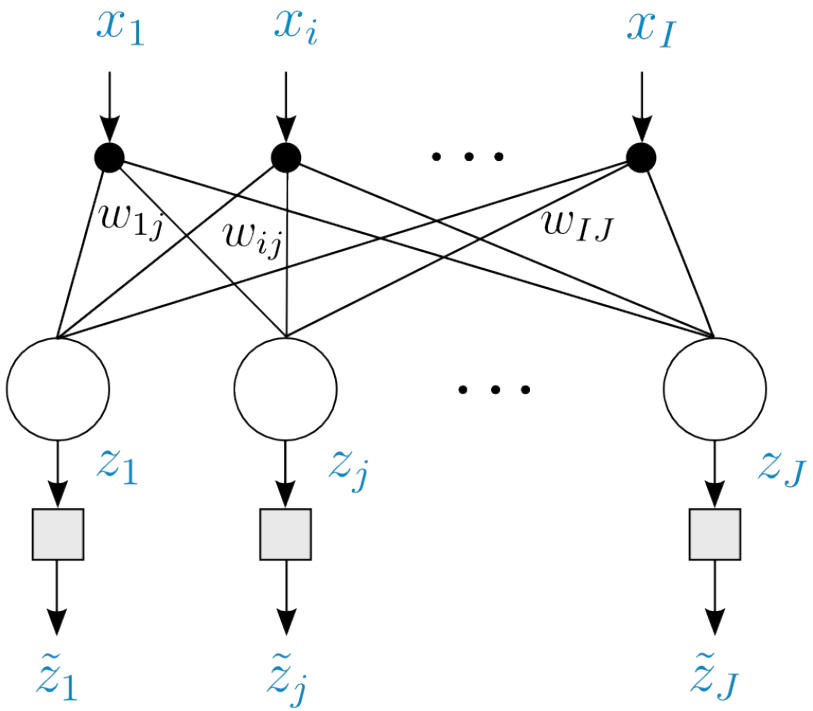
- As Acoustic Model (uncertainty decoding)

  ANN-HMM
  CD-DNN-HMM

- Basic structure: coupled layers of perceptrons

Multi-Layer Perceptrons (MLPs)

$$p(\mathbf{x}|\mathbf{I})$$



- Linear step

$$z_j = \sum_{i=1}^{I} w_{ij} x_i + b_j$$

- Non-linear step: Sigmoid

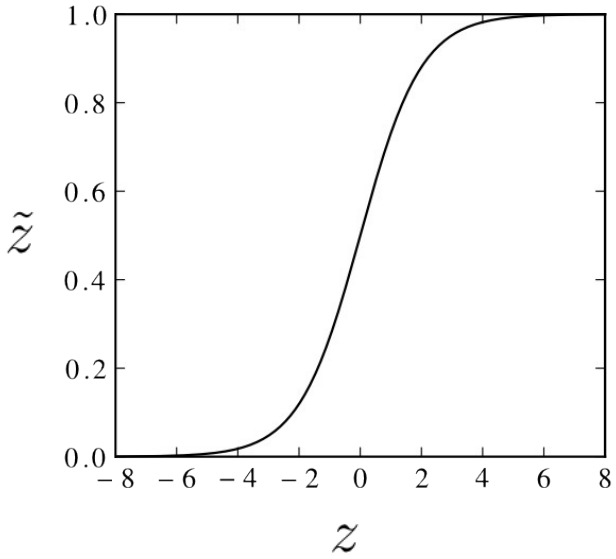
$$\tilde{z}_j = \frac{1}{1 + \exp(-z_j)}$$

- Assumption for propagation

  Output linear step Gaussian
  Outputs statistically independent

- Problem reduced to Gaussian propagation through Sigmoid

Multi-Layer Perceptrons (MLPs)

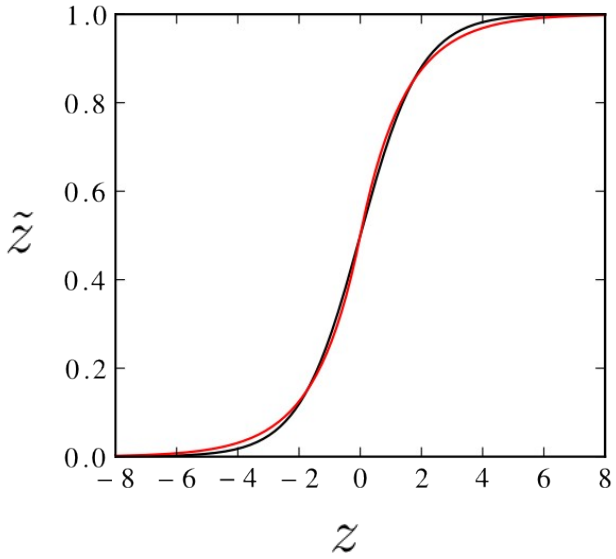


- Piecewise Sigmoid approximation

$$\tilde{z} = \frac{1}{1+e^{-z}} \approx \begin{cases} 2^{z-1} & \text{if} \quad z \leq 0 \\ 1 - 2^{-z-1} & \text{if} \quad z > 0 \end{cases}$$

- Exact propagation solutions exist for this approximation

Multi-Layer Perceptrons (MLPs)

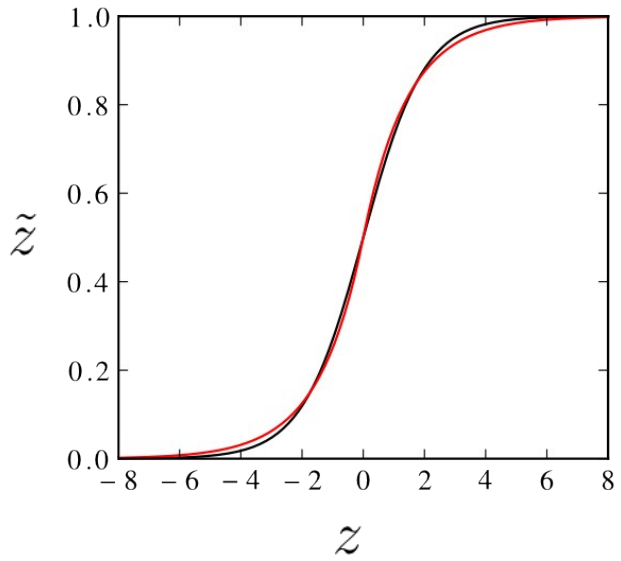


- Piecewise Sigmoid approximation

$$\tilde{z} = \frac{1}{1+e^{-z}} \approx \begin{cases} 2^{z-1} & \text{if} \quad z \leq 0 \\ 1 - 2^{-z-1} & \text{if} \quad z > 0 \end{cases}$$

- Exact propagation solutions exist for this approximation

Multi-Layer Perceptrons (MLPs)



- Piecewise Sigmoid approximation

$$\tilde{z} = \frac{1}{1+e^{-z}} \approx \begin{cases} 2^{z-1} & \text{if} \quad z \leq 0 \\ 1 - 2^{-z-1} & \text{if} \quad z > 0 \end{cases}$$

- Exact propagation solutions exist for this approximation

- The expected sigmoid output yields [Astudillo 2011b]

$$E\{\tilde{z}\} = 2^{(\mu_z + \frac{1}{2}\log(2)\sigma_z^2 - 1)}\Phi\left(-\frac{\mu_z}{\sigma_z} - \log(2)\sigma\right) - 2^{(-\mu_z + \frac{1}{2}\log(2)\sigma_z^2 - 1)}\Phi\left(\frac{\mu_z}{\sigma_z} - \log(2)\sigma\right) + \Phi\left(\frac{\mu_z}{\sigma_z}\right)$$

- Similar formulas exist for the variance and node covariance

Overview

- STFT-Speech enhancement and residual uncertainty
  - The complex Gaussian uncertainty model
  - Residual uncertainty estimation (Empirical/MSE)

- STFT Uncertainty Propagation
  - Mel-Frequency Cepstral Coefficients
  - RASTA-Perceptual-Linear-Prediction
  - Multi-Layer Perceptron

- **Integration of STFT speech enhancement and robust ASR**
  - Uncertainty Propagation as MMSE estimator
  - Uncertainty Propagation & Decoding
  - Experiments and Results

Computing MMSE estimates with STFT-UP

- We saw the MMSE estimator of the clean Fourier coefficient (Wiener filter)

$$\hat{X}^{\mathrm{MMSE}} = \arg\min_{\hat{X}} \left\{ E\left\{ \left\| X - \hat{X} \right\|^2 \right\} \right\} = E\{X|Y\}$$

- MMSE estimators of non-linear speech transformations are, however, better

- Non-linear estimators better related to percieved sound quality and ASR feature extractions

    - Amplitude, log-Amplitude (Ephraim-Malah filters)

    - MMSE-MFCC estimators [Yu 2008]

Computing MMSE estimates with STFT-UP

Fourier coeficient Amplitude A=|X|:

$$\hat{A}^{\text{MMSE}} = \arg\min_{\hat{A}} \left\{ E\left\{ \left\| A - \hat{A} \right\|^2 \right\} \right\} = E\{A|Y\} = E\{|X||Y\}$$

Fourier coeficient log-Amplitude log(|X|):

$$\widehat{\log(A)}^{\text{MMSE}} = \arg\min_{\widehat{\log(A)}} \left\{ E\left\{ \left\| \log(A) - \widehat{\log(A)} \right\|^2 \right\} \right\} = E\{\log(A)|Y\} = E\{\log(|X|)|Y\}$$

Computing MMSE estimates with STFT-UP

Fourier coeficient Amplitude A=|X|:

$$\hat{A}^{\text{MMSE}} = \arg\min_{\hat{A}} \left\{ E\left\{ \left\| A - \hat{A} \right\|^2 \right\} \right\} = E\{A|Y\} = E\{|X||Y\}$$

Fourier coeficient log-Amplitude log(|X|):

$$\widehat{\log(A)}^{\text{MMSE}} = \arg\min_{\widehat{\log(A)}} \left\{ E\left\{ \left\| \log(A) - \widehat{\log(A)} \right\|^2 \right\} \right\} = E\{\log(A)|Y\} = E\{\log(|X|)|Y\}$$

Generic MMSE estimator of non-linear feature extraction f

$$\widehat{f(X)}^{\text{MMSE}} = \arg\min_{\widehat{f(X)}} \left\{ E\left\{ \left\| f(X) - \widehat{f(X)} \right\|^2 \right\} \right\} = E\{f(X)|Y\}$$

Computing MMSE estimates with STFT-UP

- The solution is always the expectation of the Wiener filter posterior transformed through f

$$\widehat{f(X)}^{\text{MMSE}} = \arg\min_{\widehat{f(X)}} \left\{ E\left\{ \left\| f(X) - \widehat{f(X)} \right\|^2 \right\} \right\} = E\{f(X)|Y\}$$

- This is in fact what STFT-UP solves [Astudillo 2010]!

- Propagating the Wiener posterior yields MMSE estimates in MFCC, RASTA-LPCC, and MLP domain

- Also provides a variance that can be combined with observation uncertainty

Uncertainty Decoding and Propagation

- We derived UD from the modified Bayesian decoding rule (tutorial introduction)

$$\hat{\mathbf{W}} = \arg\max_{\mathbf{W}} \left\{ \int_{\mathbb{R}^{I \cdot L}} \frac{p(\mathbf{x}|\mathbf{W})}{p(\mathbf{x})} p(\mathbf{x}|\mathbf{y}) d\mathbf{x} \cdot p(\mathbf{W}) \right\}$$

- The same applies to the posterior attained from STFT-UP

$$\hat{\mathbf{W}} = \arg\max_{\mathbf{W}} \left\{ \int_{\mathbb{R}^{I \cdot L}} \frac{p(\mathbf{x}|\mathbf{W})}{p(\mathbf{x})} p(\mathbf{x}|\mathbf{Y}) d\mathbf{x} \cdot p(\mathbf{W}) \right\}$$

$$\approx \arg\max_{\mathbf{W}} \left\{ \int_{\mathbb{R}^{I \cdot L}} p(\mathbf{x}|\mathbf{W}) p(\mathbf{x}|\mathbf{Y}) d\mathbf{x} \cdot p(\mathbf{W}) \right\}$$

Uncertainty Decoding and Propagation

- We can also extend this easily to acoustic models based on neural networks.

- For each class q (diphone, senone) modeled by the multi-layer perceptron (MLP)

$$\int_{\mathbb{R}^I} p(\mathbf{x}_l|q)p(\mathbf{x}_l|\mathbf{Y})d\mathbf{x}_l = \int_{\mathbb{R}^I} \frac{\mathrm{MLP}(\mathbf{x}_l)}{p(q)} p(\mathbf{x}_l|\mathbf{Y}_l)d\mathbf{x}_l = \frac{E\{\mathrm{MLP}(\mathbf{x}_l)|\mathbf{Y}_l\}}{p(q)}$$

- The expected multi-layer perceptron output can be computed with the uncertainty propagation formulas here presented

Uncertainty Decoding and Propagation

- Another method that provides better results with STFT-UP is modified imputation [Kolossa 2005]
- State likelihood: $p(\mathbf{x}_l|q) = \mathcal{N}\left(\boldsymbol{\mu}_q, \boldsymbol{\Sigma}_q\right)$, Uncertain features: $p(\mathbf{x}_l|\mathbf{Y}) = \mathcal{N}\left(\boldsymbol{\mu}_l^x, \boldsymbol{\Sigma}_l^x\right)$
- The most likely feature value is obtained from

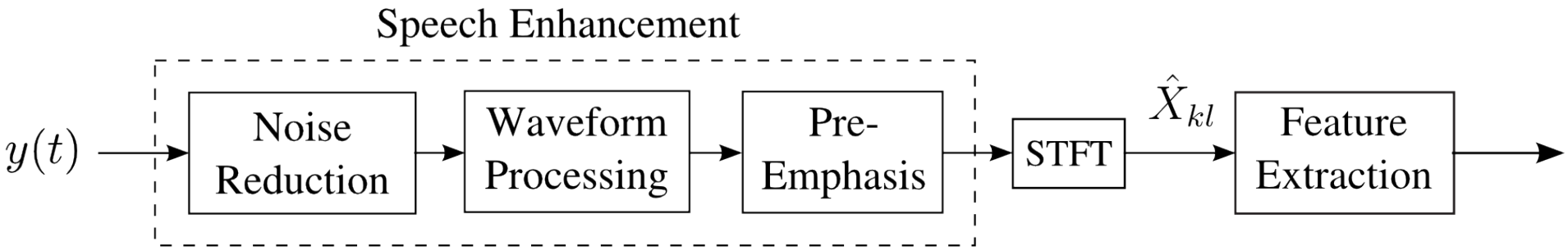
$$\hat{\mathbf{x}}_{ql}^{\mathrm{MI}} = \arg\max_{\hat{\mathbf{x}}_l} \{p(\hat{\mathbf{x}}_l|q, \mathbf{Y})\} = \arg\max_{\hat{\mathbf{x}}_l} \{p(\hat{\mathbf{x}}_l|q)p(\hat{\mathbf{x}}_l|\mathbf{Y})\}$$

- yielding

$$\hat{\mathbf{x}}_{qil}^{\mathrm{MI}} = \frac{\Sigma_{qii}}{\Sigma_{qii} + \Sigma_{ii}^x}\mu_{il}^x + \frac{\Sigma_{ii}^x}{\Sigma_{qii} + \Sigma_{ii}^x}\mu_{qi}$$

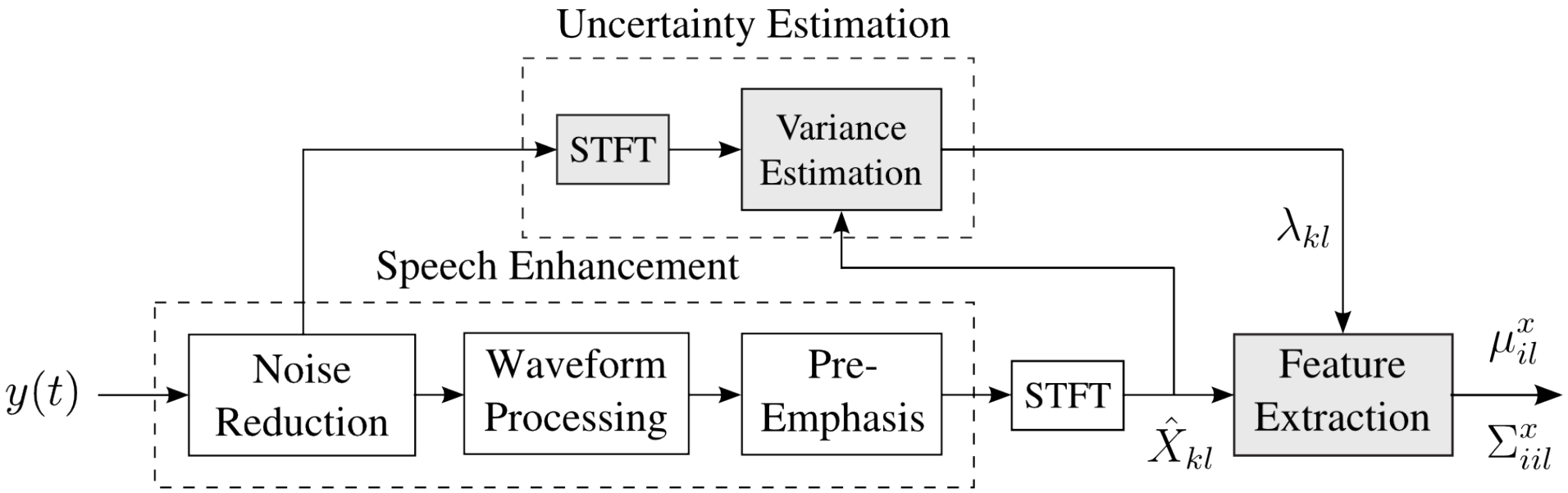
Improving the ETSI Advanced Front-End on the AURORA5 Task [Astudillo 2010b]

- Small Vocabulary (TI-Digits)
- Non-stationary additive noise, reveberant speech
- ETSI Advanced-Front-End features
- Empirical uncertainty estimation
- Tested: Uncertainty Decoding (UD) and Modified Imputation (MI)



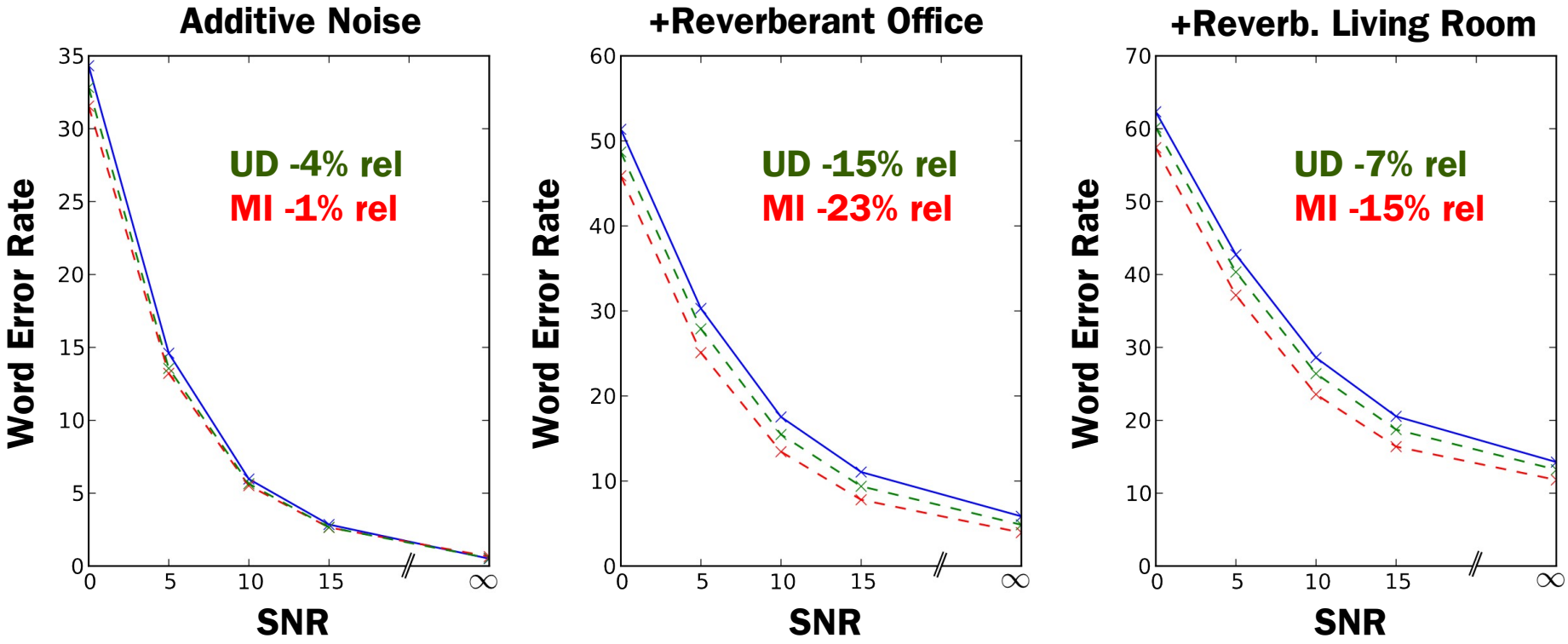
Improving the ETSI Advanced Front-End on the AURORA5 Task [Astudillo 2010b]

- Small Vocabulary (TI-Digits)
- Non-stationary additive noise, reveberant speech
- ETSI Advanced-Front-End features
- Empirical uncertainty estimation
- Tested: Uncertainty Decoding (UD) and Modified Imputation (MI)



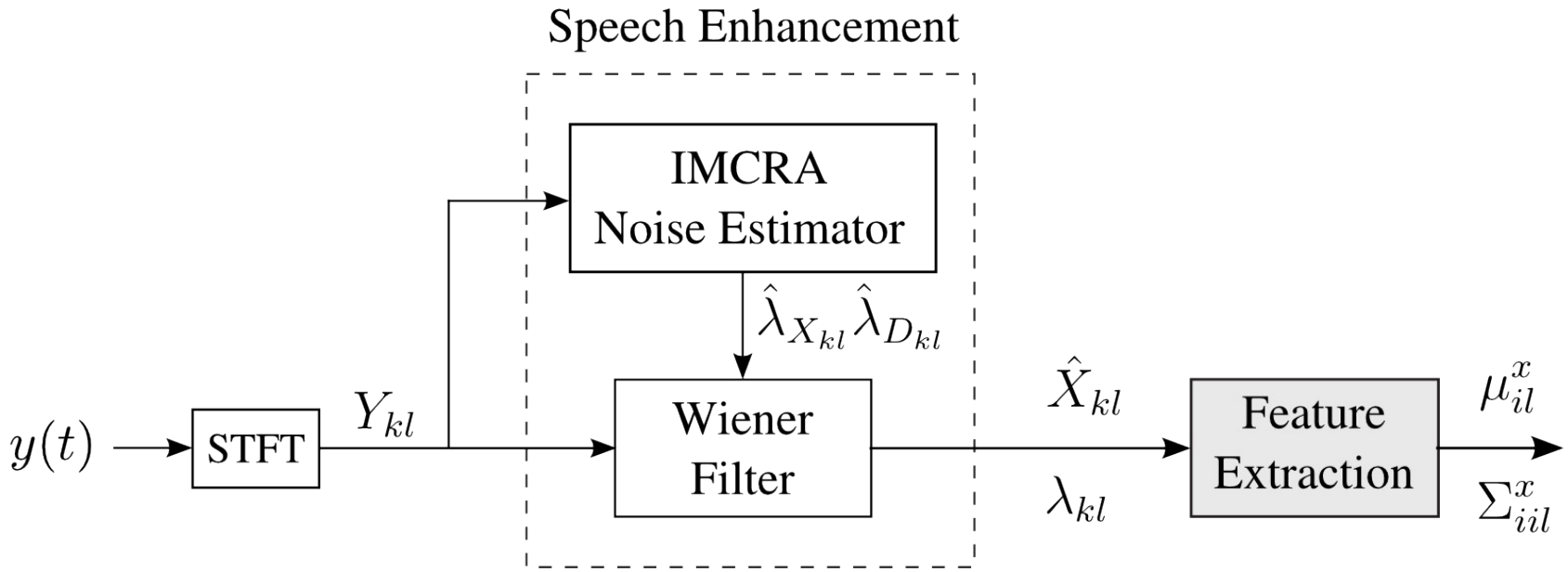
Improving the ETSI Advanced Front-End on the AURORA5 Task [Astudillo 2010b]

- Small Vocabulary (TI-Digits)
- Non-stationary additive noise, reveberant speech
- ETSI Advanced-Front-End features
- Empirical uncertainty estimation
- Tested: Uncertainty Decoding (**UD**) and Modified Imputation (**MI**)



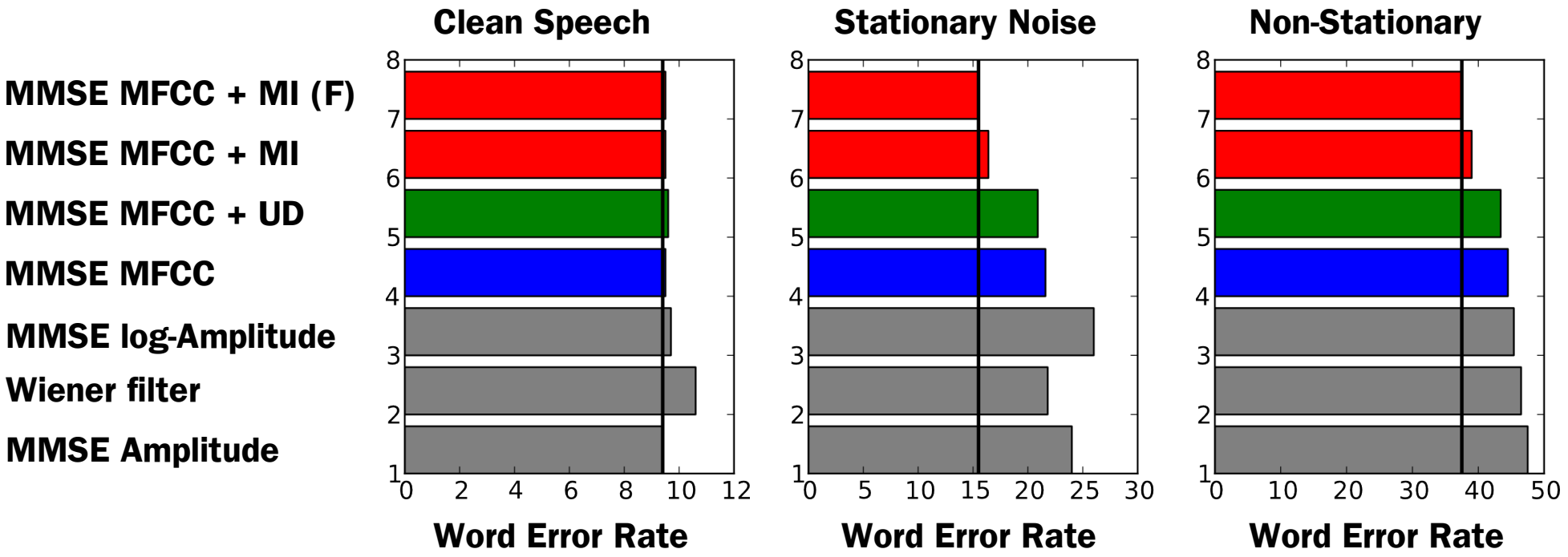
Results on AURORA4 Task against conventional STFT Speech Enhancement

- Not-so-small Vocabulary (Wall Street-Journal, 5K-Words)
- Non-stationary additive noise
- IMCRA noise estimator + decision directed method
- MFCC features, with deltas and delta-deltas (no delay otherwise)
- Compared with amplitude, log-amplitude and Wiener estimators



Results on AURORA4 Task against conventional STFT Speech Enhancement

- MMSE-MFCC outperforms all other MMSE estimators
- Use of **MI** achieves **-17% / -19%** relative WER reduction against best STFT MMSE
- Large reduction for stationary noise **-40%** rel. (**-29%** vs Wiener)
- Non-stationary noise reduction **-17%** rel. (affected by noise estimation errors)



Closing Remarks

- STFT Uncertainty Propagation allows for the integration of speech enhancement and ASR
- Exploits existing (and prospective) expertise in the speech enhancement field through empirical or MSE uncertainty propagation
- Recipes were presented to attain propagation through various feature extractions
- STFT-UP requires low computational costs and minimal modifications of ASR systems

Thank You!

Code for STFT-UP (including HTK patches for MI, UD) available from http://www.astudillo.com/ramon/research/stft-up/

Learning from Noisy Data

Emmanuel Vincent
INRIA Rennes – Bretagne Atlantique, France
emmanuel.vincent@inria.fr

Overview

- Bayesian uncertainty estimation for STFT-domain enhancement
  - Theoretical Bayesian uncertainty estimator
  - Variational Bayesian approximation
  - Example results
- Expectation maximization training of acoustic models with unreliable input features
  - EM training algorithm
  - Example results

Overview

- **Bayesian uncertainty estimation for STFT-domain enhancement**
  - Theoretical Bayesian uncertainty estimator
  - Variational Bayesian approximation
  - Example results
- Expectation maximization training of acoustic models with unreliable input features
  - EM training algorithm
  - Example results

Remember empirical uncertainty estimation and residual MSE?

Uncertainty estimators previously described in this tutorial:

- empirical nonlinear distortion estimator $\lambda_{kl} = \alpha \left| |Z_{kl}| - |\widehat{X}_{kl}| \right|^2$ with Z output of linear beamformer

  Ad hoc solution, not very elegant but effective

- residual MSE estimator $\lambda_{kl} = \dfrac{\widehat{\lambda}_{X_{kl}} \widehat{\lambda}_{D_{kl}}}{\widehat{\lambda}_{X_{kl}} + \widehat{\lambda}_{D_{kl}}}$ stemming from the Wiener filter

  Exact expression of the Wiener filter posterior given the estimated speech and distortion variances $\widehat{\lambda}_{X_{kl}}$ and $\widehat{\lambda}_{D_{kl}}$: $p(X_{kl}|Y_{kl}, \widehat{\lambda}_{X_{kl}}, \widehat{\lambda}_{D_{kl}}) = \mathcal{N}_{\mathbb{C}}(\widehat{X}_{kl}^{\mathrm{MMSE}}, \lambda_{kl})$

Wait, this does not account for the uncertainty about these variances!

What are the parameters of the speech enhancement method?

What's more, the variances $\lambda_{X_{kl}}$ and $\lambda_{D_{kl}}$ often derive from a set of parameters $\boldsymbol{\theta}$, which are themselves uncertain!!

Examples:

- steering/blocking vectors for beamforming, ICA,
- hidden states, initial/transition probabilities, and state means/variances for GMM, HMM,
- basis spectra and scaling coefficients for NMF, harmonic NMF, and variants thereof,
- decay parameters for late reverberation models.

Flexible FASST framework generalizing some of these enhancement methods.
Toolbox available from http://bass-db.gforge.inria.fr/fasst/

Theoretical Bayesian uncertainty estimator

The theoretical Bayesian uncertainty estimator is given by

$$p(\mathbf{X}|\mathbf{Y}) = \int p(\mathbf{X}, \boldsymbol{\theta}|\mathbf{Y})\, d\boldsymbol{\theta}$$

Problem: this integral typically involves thousands of dimensions!

Approximate variational Bayesian uncertainty estimator

Variational Bayes (VB): approximate the joint posterior $p(\mathbf{X}, \boldsymbol{\theta}|\mathbf{Y})$ by the closest distribution $q(\mathbf{X}, \boldsymbol{\theta})$ for which the integral is tractable.

When q is assumed to factor as $q(\mathbf{X}, \boldsymbol{\theta}) = \prod_{kl} q(X_{kl}) q(\boldsymbol{\theta})$, the posterior over X_{kl} is simply obtained as

$$p(X_{kl}|\mathbf{Y}) \approx q(X_{kl}).$$

VB inference: the theory

The closeness between p and q is measured via the Kullback-Leibler divergence

$$KL(q||p) = \int q(\mathbf{X}, \boldsymbol{\theta}) \log \frac{q(\mathbf{X}, \boldsymbol{\theta})}{p(\mathbf{X}, \boldsymbol{\theta}|\mathbf{Y})}.$$

Minimizing this quantity is equivalent to maximizing the so-called variational free energy

$$\mathscr{L}(q) = \int q(\mathbf{X}, \boldsymbol{\theta}) \log \frac{p(\mathbf{Y}, \mathbf{X}, \boldsymbol{\theta})}{q(\mathbf{X}, \boldsymbol{\theta})} \, d\mathbf{X} \, d\boldsymbol{\theta}$$

This function is sometimes not maximizable in closed form and minorization by a parametric bound $f(\mathbf{Y}, \mathbf{X}, \boldsymbol{\theta}, \boldsymbol{\Omega}) \leq p(\mathbf{Y}, \mathbf{X}, \boldsymbol{\theta})$ may be needed.

Assuming $q(\mathbf{X}, \boldsymbol{\theta}) = \prod_{kl} q(X_{kl}) \prod_i q(\theta_i)$, the solution is iteratively estimated by

1. tightening the bound w.r.t. the variational parameters $\boldsymbol{\Omega}$,
2. $q(\theta_i) \propto \exp[\mathbb{E}_{\mathbf{X}, \theta_{i' \neq i}} \log f(\mathbf{Y}, \mathbf{X}, \boldsymbol{\theta}, \boldsymbol{\Omega})]$
3. $q(X_{kl}) \propto \exp[\mathbb{E}_{X_{k'l' \neq kl}, \boldsymbol{\theta}} \log f(\mathbf{Y}, \mathbf{X}, \boldsymbol{\theta}, \boldsymbol{\Omega})]$

VB inference in practice

This results in a VB expectation-maximization (EM) algorithm where posterior distributions over the parameters are updated instead of deterministic parameter values as in the usual maximum likelihood (ML)-based EM algorithm.

Resulting approximating distributions for FASST given in [Adiloğlu 2012]:

- complex-valued Gaussian for X_{kl} and for the steering vectors,
- generalized inverse Gaussian for the NMF parameters (generalization of gamma and inverse-gamma distributions).

Speaker recognition benchmark

We here consider speaker recognition with a baseline classifier rather than ASR because it allows us to focus on the performance improvement due to acoustic modeling alone.

Data: 2011 PASCAL CHiME Speech Separation and Recognition Challenge
http://spandh.dcs.shef.ac.uk/projects/chime/challenge.html
Short spoken commands mixed with genuine noise backgrounds recorded in a family home.

Training: 20 clean utterances from each of 34 speakers
Test: 20 other utterances per speaker, each mixed at 6 different SNRs

Enhancement: multichannel NMF (ML or VB).

Features: static MFCCs (2 to 20), log-normal UP

Baseline classifier: 32-component GMMs with diagonal covariances, initialized by hierarchical K-means

Signal enhancement accuracy

Average SDR (dB) over the estimated target signals

	without UP
ML enhancement	1.58
VB enhancement	1.70

VB performs similarly to ML in terms of signal enhancement accuracy. . .

MMSE feature estimation accuracy

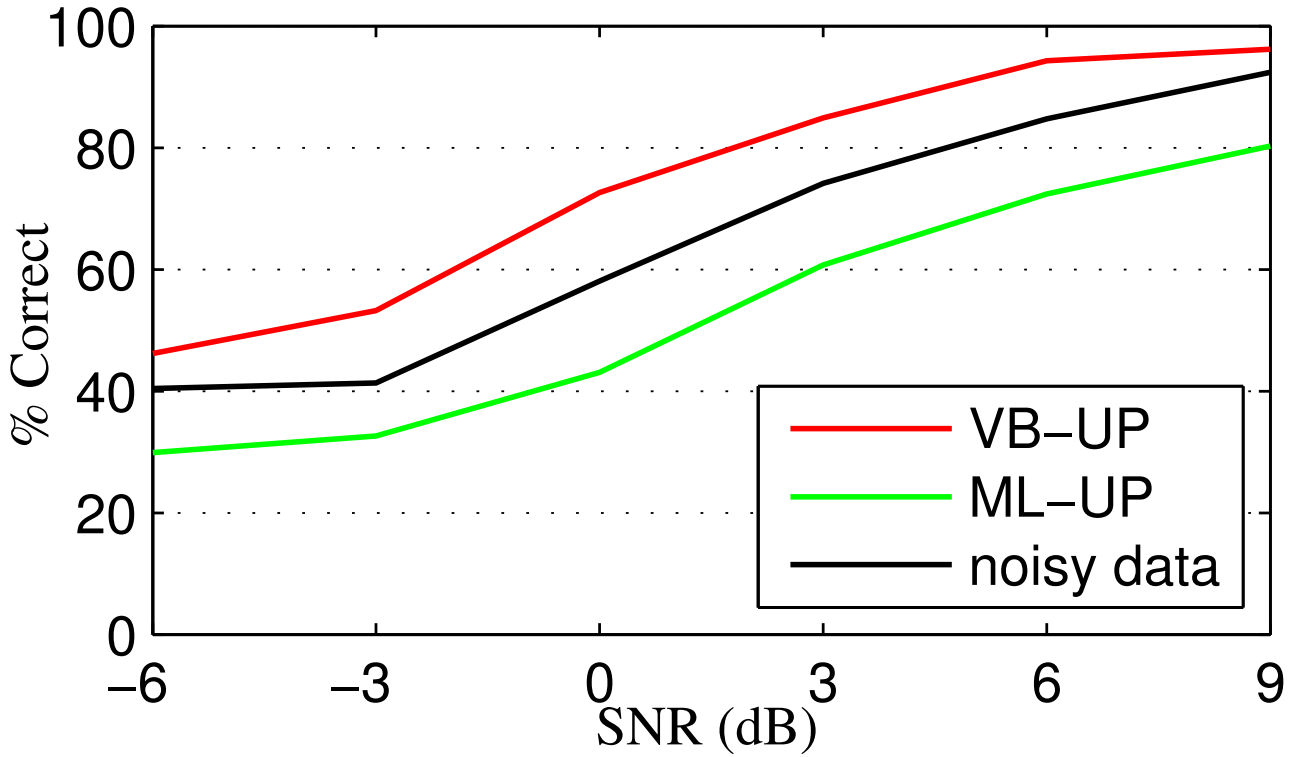
Average RMS error over the estimated MMSE MFCCs $\widehat{x}_{il}^{\mathrm{MMSE}}$

	without UP	with UP
ML enhancement	0.869	0.767
VB enhancement	0.862	0.763

. . . VB performs similarly to ML in terms of MMSE feature estimation accuracy. . .

Speaker recognition accuracy

. . . but it performs better in terms of speaker recognition accuracy



For the particular task, data, and enhancement method considered here, signal enhancement decreases classification accuracy and ML-UP makes it even worse (yes, this can happen!)

But VB-UP improves it by 9% absolute compared to using noisy data.

Overview

- Bayesian uncertainty estimation for STFT-domain enhancement
  - Theoretical Bayesian uncertainty estimator
  - Variational Bayesian approximation
  - Example results
- **Expectation maximization training of acoustic models with unreliable input features**
  - EM training algorithm
  - Example results

What about mixed training?

So far, we (and others) have assumed that the acoustic models have been trained on clean data but...

- speaker-dependent clean data may not be available,
- mixed training is an effective technique aside from feature and model compensation.

NAT is complementary to UD because

- the Gaussian parametric model (or other models) of uncertainty may not fit the actual distribution of uncertainty,
- even when it does, its covariance $\mathbf{\Sigma}_l^x$ (or their parameters) are never perfectly estimated.

NAT allows to learn from data the residual distortion that UP failed to represent.

Naive approach

Naive approach: conventional training on noisy or enhanced training data, followed by UD on the test data.

Problem: although this is sometimes effective, the model compensation is biased.

The variance of the distortion is counted twice:

- the distortion on the training data is accounted for by the parameters of the acoustic model,
- the distortion on the test data is accounted for by the uncertainty estimator,
- the variances of both distortions add up in the UD rule.

We don't want the model parameters to account for the distortion on the training data.

The model parameters should represent clean data only (and the small residual distortion over the training data not represented by UP).

EM training algorithm for GMMs (1)

Again, let us focus on GMM-based speaker recognition first for simplicity.

Notations:

- GMM parameters $\{\boldsymbol{\mu}_q, \boldsymbol{\Sigma}_q, \omega_q\}$ for each speaker class C,
- uncertain data $p(x_l|\mathbf{Y}) \sim \mathcal{N}(\boldsymbol{\mu}_l^x, \boldsymbol{\Sigma}_l^x)$.

Reminder: assuming $\{\boldsymbol{\mu}_q, \boldsymbol{\Sigma}_q, \omega_q\}$ have been trained on clean data, UD relies on the modified likelihood

$$p(\boldsymbol{\mu}_l^x, \boldsymbol{\Sigma}_l^x|C) = \prod_l \sum_q \omega_q \mathcal{N}(\boldsymbol{\mu}_l^x|\boldsymbol{\mu}_q, \boldsymbol{\Sigma}_q + \boldsymbol{\Sigma}_l^x)$$

In order to train on noisy data, we simply maximize the UD likelihood on the training data.

This can be achieved via the EM algorithm considering both the states q_l and the clean data $\mathbf{x}_l$ as hidden data [Ozerov 2012].

EM training algorithm for GMMs (2)

E-step: estimation of the underlying clean feature moments by Wiener filtering

$$\gamma_{q,l} \propto \omega_q \mathcal{N}(\boldsymbol{\mu}_l^x | \boldsymbol{\mu}_q, \boldsymbol{\Sigma}_q + \boldsymbol{\Sigma}_l^x),$$

$$\mathbf{W}_{q,l} = \boldsymbol{\Sigma}_q (\boldsymbol{\Sigma}_q + \boldsymbol{\Sigma}_l^x)^{-1},$$

$$\hat{\mathbf{x}}_{q,l} = \boldsymbol{\mu}_q + \mathbf{W}_{q,l} \left(\boldsymbol{\mu}_l^x - \boldsymbol{\mu}_q \right),$$

$$\widehat{\mathbf{R}}_{\mathbf{xx},q,l} = \hat{\mathbf{x}}_{q,l} \hat{\mathbf{x}}_{q,l}^T + (\mathbf{I} - \mathbf{W}_{q,l}) \boldsymbol{\Sigma}_q.$$

M-step: update GMM parameters

$$\omega_q = \frac{1}{L} \sum_l \gamma_{q,l},$$

$$\boldsymbol{\mu}_q = \frac{1}{\sum_l \gamma_{q,l}} \sum_l \gamma_{q,l} \hat{\mathbf{x}}_{q,l},$$

$$\boldsymbol{\Sigma}_q = \mathrm{diag}\left(\frac{1}{\sum_l \gamma_{q,l}} \sum_l \gamma_{q,l} \widehat{\mathbf{R}}_{\mathbf{xx},q,l} - \boldsymbol{\mu}_q \boldsymbol{\mu}_q{}^T \right).$$

By analogy with UD, we call this algorithm uncertainty training.

Toolbox available from http://bass-db.gforge.inria.fr/amulet/

Speaker recognition benchmark

Same data, enhancement algorithm and baseline classifier as above.

Empirical STFT uncertainty estimation (better than MSE here) + log-normal UP.

Training: 20 clean utterances from each of the 34 speakers, each mixed at 6 different SNRs

Results averaged into 4 training conditions:

- clean,
- matched (same SNR),
- unmatched (different SNR),
- multicondition (all SNRs, hence more noisy data)

Results

Speaker recognition accuracy (average over all SNRs)

Enhanced signal	Training approach	Decoding approach	Training condition			
			Clean	Matched	Unmatched	Multi
No	Conventional	Conventional	65.17	71.81	69.34	84.09
Yes	Conventional	Conventional	55.22	82.11	80.91	90.12
Yes	Conventional	Uncertainty	**75.51**	78.60	77.58	85.02
Yes	Uncertainty	Uncertainty	**75.51**	**82.87**	**81.52**	**91.13**

The naive approach decreases performance in this setting.

Uncertainty training overcomes this issue and improves by up to 1% absolute in all training conditions. Improvements up to 4% may be observed with other uncertainty estimators.

Conclusion: Uncertainty training makes it possible to exploit all available training data, whatever their noise level. The more data, the better.

EM training algorithm for HMMs (1)

Notations (M states, Q components):

- HMM parameters $\mathcal{M} = \{\boldsymbol{\mu}_{mq}, \boldsymbol{\Sigma}_{mq}, \omega_{mq}, \pi_m, a_{mn}\}$,
- uncertain training data $p(x_l|\mathbf{Y}) \sim \mathcal{N}(\boldsymbol{\mu}_l^x, \boldsymbol{\Sigma}_l^x)$,
- observation probabilities $b_m(\boldsymbol{\mu}_l^x, \boldsymbol{\Sigma}_l^x) = p(\boldsymbol{\mu}_l^x, \boldsymbol{\Sigma}_l^x|s_l = m, \mathcal{M})$,
- forward probabilities $\alpha_{m,l} = p(\boldsymbol{\mu}_{1:l}, \boldsymbol{\Sigma}_{1:l}, s_l = m|\mathcal{M})$,
- backward probabilities $\beta_{m,l} = p(\boldsymbol{\mu}_{l+1:L}, \boldsymbol{\Sigma}_{l+1:L}|s_l = m, \mathcal{M})$,
- $\gamma_{mq,l} = p(s_l = m, q_l = q|\boldsymbol{\mu}, \boldsymbol{\Sigma}, \mathcal{M})$,
- $\xi_{mn,l} = p(s_l = m, s_{l+1} = n|\boldsymbol{\mu}, \boldsymbol{\Sigma}, \mathcal{M})$

EM training algorithm for HMMs (2)

E-step:

$$b_m(\boldsymbol{\mu}_l^x, \boldsymbol{\Sigma}_l^x) = \sum_q \omega_{mq}\, \mathcal{N}(\boldsymbol{\mu}_l^x | \boldsymbol{\mu}_{mq}, \boldsymbol{\Sigma}_{mq} + \boldsymbol{\Sigma}_l^x),$$

$\alpha_{m,l}$ computed via the forward algorithm using $b_m(\boldsymbol{\mu}_l^x, \boldsymbol{\Sigma}_l^x)$,

$\beta_{m,l}$ computed via the backward algorithm using $b_m(\boldsymbol{\mu}_l^x, \boldsymbol{\Sigma}_l^x)$,

$$\gamma_{mq,l} \propto \alpha_{m,l}\, \beta_{m,l}\, \omega_{mq}\, \mathcal{N}(\boldsymbol{\mu}_l^x | \boldsymbol{\mu}_{mq}, \boldsymbol{\Sigma}_{mq} + \boldsymbol{\Sigma}_l^x),$$

$$\xi_{mn,l} \propto \alpha_{m,l}\, a_{mn}\, b_n(\boldsymbol{\mu}_{l+1}^x, \boldsymbol{\Sigma}_{l+1}^x)\, \beta_{n,l+1},$$

$$\mathbf{W}_{mq,l} = \boldsymbol{\Sigma}_{mq}(\boldsymbol{\Sigma}_{mq} + \boldsymbol{\Sigma}_l^x)^{-1},$$

$$\widehat{\mathbf{x}}_{mq,l} = \boldsymbol{\mu}_{mq} + \mathbf{W}_{mq,l}(\boldsymbol{\mu}_l^x - \boldsymbol{\mu}_{mq}),$$

$$\widehat{\mathbf{R}}_{\mathbf{xx},mq,l} = \widehat{\mathbf{x}}_{mq,l}\widehat{\mathbf{x}}_{mq,l}^T + (\mathbf{I} - \mathbf{W}_{mq,l})\boldsymbol{\Sigma}_{mq}.$$

EM training algorithm for HMMs (3)

M-step:

$$\pi_m = \sum_q \gamma_{mq,1},$$

$$a_{mn} = \frac{1}{\sum_l \sum_q \gamma_{mq,l}} \sum_l \xi_{mn,l},$$

$$\omega_{mq} = \frac{1}{\sum_l \sum_{q'} \gamma_{mq',l}} \sum_l \gamma_{mq,l},$$

$$\boldsymbol{\mu}_{mq} = \frac{1}{\sum_l \gamma_{mq,l}} \sum_l \gamma_{mq,l} \widehat{\mathbf{x}}_{mq,l},$$

$$\boldsymbol{\Sigma}_{mq} = \mathrm{diag}\left(\frac{1}{\sum_l \gamma_{mq,l}} \sum_l \gamma_{mq,l} \widehat{\mathbf{R}}_{\mathbf{xx},mq,l} - \mathbf{x}_{mq} \mathbf{x}_{mq}^T \right).$$

Other variants

This algorithm is also applicable with minor modification to

- maximum a posteriori (MAP) acoustic model adaptation,
- maximum likelihood linear regression (MLLR) acoustic model adaptation,
- other data than audio.

For MAP/MLLR, only the M-step should be modified, while the E-step remains unchanged.

Connections with NAT-VTS [Kalinli 2009] and JUD [Liao 2007] designed for more stationary noise.

Wrap-up and Perspectives

Emmanuel Vincent
INRIA Rennes – Bretagne Atlantique, France
emmanuel.vincent@inria.fr

One-slide wrap-up

Noise-robust ASR techniques can be classified into 6 classes, among which hybrid approaches combining feature and model compensation perform best.

The uncertainty $p(\mathbf{x}|\mathbf{Y})$ estimated via a parametric model of speech distortion can be exploited for dynamic model compensation using the modified acoustic likelihood

$$\int_{\mathbb{R}^{I.L}} \frac{p(\mathbf{x}|\mathbf{W})}{p(\mathbf{x})} p(\mathbf{x}|\mathbf{Y}) d\mathbf{x} \approx \int_{\mathbb{R}^{I.L}} p(\mathbf{x}|\mathbf{W}) p(\mathbf{x}|\mathbf{Y}) d\mathbf{x} \quad \text{instead of} \quad p(\mathbf{Y}|\mathbf{W})$$

This modified acoustic likelihood can be used both for decoding (UD) or training. In the case of HMM/GMMs, the variances of the acoustic model and the uncertainty simply add up.

Compared to feature-domain uncertainty estimation techniques, STFT-domain uncertainty estimation followed by uncertainty propagation (UP) enables the exploitation of additional enhancement cues (f0, spatial position, etc).

UP recipes exist for a variety of features (MFCC, RASTA-PLP, MLP, ETSI-AFE).

Is it used in practice?

Back to the 2011 CHiME Speech Separation and Recognition Challenge [Barker, to appear].
http://spandh.dcs.shef.ac.uk/projects/chime/PCC/results.html

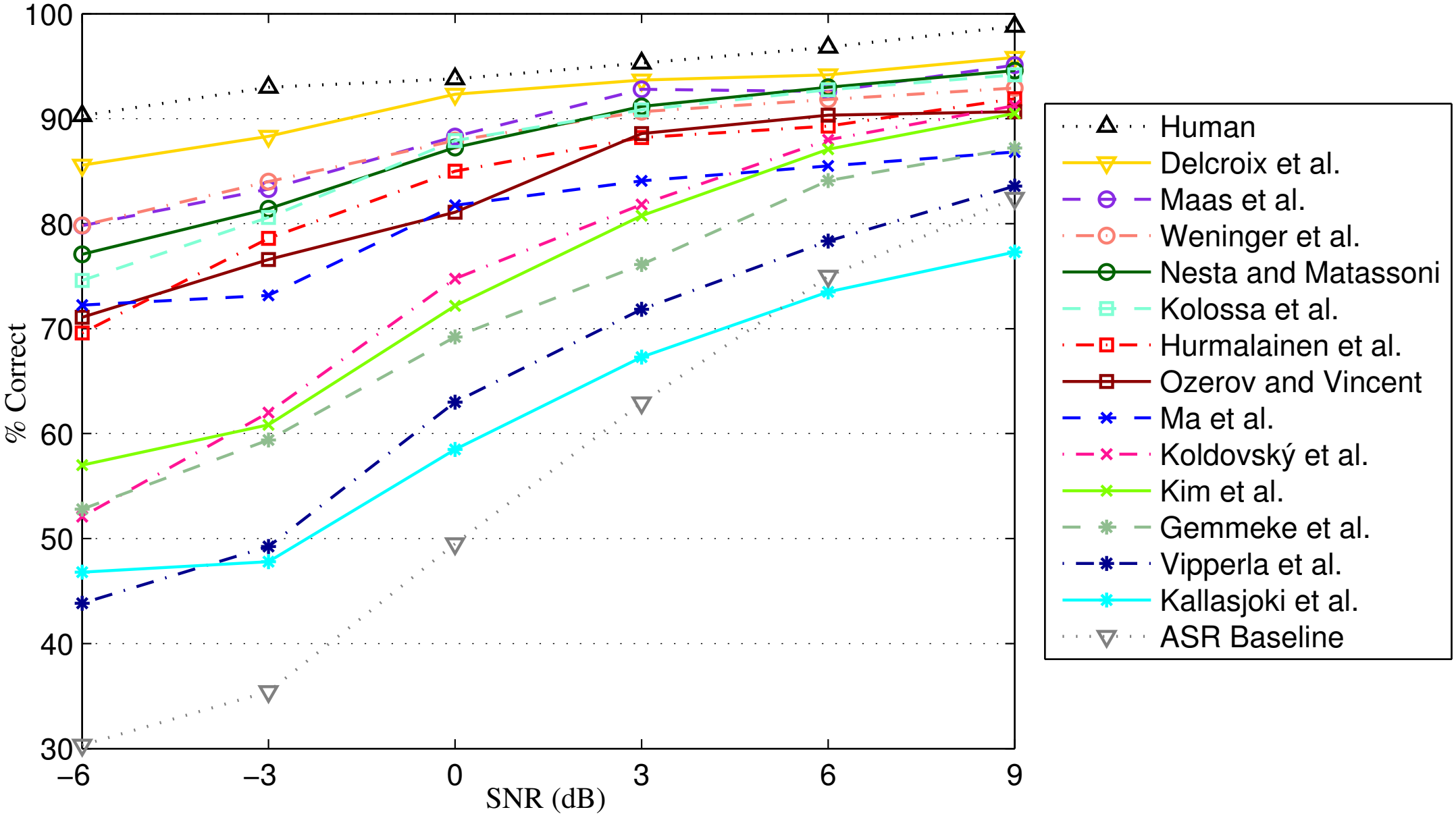
	Feature compensation	Robust features	Model compensation
Delcroix et al.	X		X
Maas et al.	X		X
Weninger et al.	X	X	X
Nesta & Matassoni	X	X	X
Kolossa et al.	X	X	X
Hurmalainen et al.		X	X
Ozerov & Vincent	X		X
Ma et al.	X	X	X
Koldovský et al.	X		
Kim et al.	X	X	X
Gemmeke et al.	X		X
Vipperla et al.	X		X
Kallasjoki et al.	X	X	X

11 hybrid approaches among 13.

5 exploited boolean or Gaussian uncertainty:

- fragment decoding [Ma 2011] or channel-attentive decoding [Kim 2011],
- UD [Kallasjoki 2011], MI [Ma 2011] or dynamic variance adaptation [Delcroix 2011].

How well does it work? (1)



How well does it work? (2)

The most effective strategies are the simplest ones:

- mixed training,
- careful handling of model size and speaker adaptation,
- enhancement based on spatial cues.

Uncertainty handling improves accuracy by up to 10% absolute alone, but by 1% absolute only when used in combination with the best front ends and back ends.

Perspective 1: greater enhancement

A research field of its own.

Tighter communication needed with the Audio and Acoustic Signal Processing (AASP) and Machine Learning for Signal Processing (MLSP) communities.

See the annual Signal Separation Evaluation Campaign (SiSEC) for recent achievements
http://sisec.wiki.irisa.fr/

Localization and separation of moving sources or sources within a strongly reverberant environment remain very challenging.

Perspective 2: more accurate uncertainty estimation

The potential of uncertainty handling is much greater than what has been achieved so far, according to experiments using oracle uncertainty [Deng 2005, Ozerov 2012].

STFT-domain uncertainty estimation still involves many heuristics and approximations.

Better theoretical understanding of successful heuristics is needed.

This should help deriving improved approximations.

Perspective 3: exploitation of underexploited uncertainties

Uncertainties used so far:

- target and background spectra,
- some hyper-parameters: steering vectors, NMF basis spectra and scaling coefficients, late reverberation decay parameters.

The uncertainty about other hyper-parameters remains to be better exploited:

- target spatial direction,
- uncertainty in other modalities, e.g., video [Vorwerk 2011].

Perspective 4: understanding the big picture

Why is it that uncertainty handling sometimes degrades performance?

System components often studied in isolation:

- feature enhancement,
- uncertainty estimation,
- decoding rule.

But strong interplay exists: certain uncertainty estimators better estimate uncertainty for certain forms of speech distortion than others.

Again, better understanding of interplay should help designing improved noise-robust ASR systems.

The 2nd CHiME Speech Separation and Recognition Challenge

Goal: recognizing distant-microphone speech mixed in two-channel nonstationary noise recorded over a period of several weeks in a real family house.

Two tracks:

- medium vocabulary: WSJ 5k sentences uttered by a static speaker (similar to Aurora 4)
- small vocabulary: simpler commands but small head movements.

Deadline: January 15, 2013
ICASSP satellite workshop: June 1, 2013, Vancouver, Canada

Any approach is welcome, whether emerging or established!

http://spandh.dcs.shef.ac.uk/chime_challenge/

Let's join efforts

We (Ramón and Emmanuel) are proposing to create an ISCA SIG on robust speech processing.

What about?

- robustness to noise, reverberation, inter- and intra-speaker variability, speaking style…
- application to speech enhancement, ASR, speaker recognition…

What for?

- share problems, resources and good practices,
- promote our area within the communities involved (SL, AASP, MSLP) and the industry,
- support initiatives (special sessions, workshops, challenges, etc.),
- ultimately increase funding and foster new research directions.

If you are interested, please come and meet us.