# An extended experimental investigation of DNN uncertainty propagation for noise robust ASR

Karan Nathwani, Juan A Morales-Cordovilla, Sunit Sivasankaran, Irina Illina, Emmanuel Vincent

# AN EXTENDED EXPERIMENTAL INVESTIGATION OF DNN UNCERTAINTY PROPAGATION FOR NOISE ROBUST ASR

*Karan Nathwani*[1,2,3]*, Juan A. Morales-Cordovilla*[4]*, Sunit Sivasankaran*[1,2,3]*, Irina Illina*[1,2,3] *and Emmanuel Vincent* [1,2,3]

[1] Inria, Villers-lès-Nancy, F-54600, France
[2] Université de Lorraine, LORIA, UMR 7503, Vandœuvre-lès-Nancy, F-54506, France
[3] CNRS, LORIA, UMR 7503, Vandœuvre-lès-Nancy, F-54506, France
[4] Dept. of TSTC, Universidad de Granada, Spain

## ABSTRACT

Automatic speech recognition (ASR) in noisy environments remains a challenging goal. Recently, the idea of estimating the uncertainty about the features obtained after speech enhancement and propagating it to dynamically adapt deep neural network (DNN) based acoustic models has raised some interest. However, the results in the literature were reported on simulated noisy datasets for a limited variety of uncertainty estimators. We found that they vary significantly in different conditions. Hence, the main contribution of this work is to assess DNN uncertainty decoding performance for different data conditions and different uncertainty estimation/propagation techniques. In addition, we propose a neural network based uncertainty estimator and compare it with other uncertainty estimators. We report detailed ASR results on the CHiME-2 and CHiME-3 datasets. We find that, on average, uncertainty propagation provides similar relative improvement on real and simulated data and that the proposed uncertainty estimator performs significantly better than the one in [1]. We also find that the improvement is consistent, but it depends on the signal-to-noise ratio (SNR) and the noise environment.

***Index Terms***— Robust ASR, acoustic modeling, DNN, uncertainty estimation, uncertainty propagation.

## 1. INTRODUCTION

Robust ASR is a challenging issue in everyday environments. Classical approaches operate on the font-end or the back-end [2]. Front-end approaches estimate enhanced features from the distorted (noisy or reverberated) features, which are fed to the back-end and treated as if they were clean. However, the enhanced features are not clean: some distortions remain, which limit the ASR performance. Multi-condition training is a popular back-end approach that improves the ASR performance by training acoustic models on enhanced data. It compensates the distortions on average, but the ASR performance at a given time is still highly dependent on the distortions at that time.

Uncertainty decoding has emerged as a promising approach for dynamically tackling the speech distortions remaining after speech enhancement [3–5]. In this approach, instead of point estimates, the posterior distribution of the clean features given the observed features is estimated and dynamically applied to modify the acoustic model outputs while decoding. The distribution of speech distortions is typically approximated as a Gaussian from which the uncertainty or variance of speech distortions is derived. The uncertainty can be computed directly in the ASR feature domain [2, 6–10] or propagated from the spectral domain to the feature domain [1, 11–15]. It was observed that the latter approach yields better performance, because complex spectra provide additional spatial and pitch cues which help discriminating speech and reverberation or noise. In [16], a two-step nonparametric uncertainty estimation/propagation technique was proposed which is based on learning two separate non-linear mappings in the spectral and feature domains. In the case of Gaussian mixture model (GMM) based acoustic models, the expectation of senone likelihoods with respect to the feature uncertainty distribution can be computed in closed form [3–5].

Uncertainty propagation in deep neural network (DNN) acoustic models is more difficult due to the nonlinear activations. Piecewise exponential (PIE) approximation of the activation function, Monte Carlo (MC) sampling, and the unscented transform (UT) have been proposed for uncertainty propagation in this context [17–21]. In the following, we focus on the methods designed for noisy data and hybrid DNN based acoustic models [17–19]. The experimental results in these papers were obtained using either oracle speech and distortion estimates, which are not available in practice, or Kolossa's uncertainty (KU) estimator [1], which is known to be suboptimal compared to the nonparametric estimator in [16]. Additionally, these results were reported on simulated datasets only, namely Aurora-4 [22] and CHiME-2 [23], and the variation of performance between different uncertainty estimators at various SNRs and under different noise environments has not been explored.

The main contribution of this work is to investigate the impact of uncertainty propagation on the ASR performance in different data conditions (CHiME-2 and CHiME-3 [24]) and with different uncertainty estimation/propagation approaches. In addition, we propose a neural network uncertainty (NNU) estimator that is trained to estimate feature-domain uncertainties. We compare experimentally this NNU estimator, the KU estimator, and an oracle estimator (OU) with three propagation techniques, namely PIE, MC, and UT.

The remainder of the paper is organized as follows. Section 2 recalls the uncertainty propagation techniques for DNNs. Section 3 presents the data, the experimental setup, the proposed NNU estimator and other uncertainty estimators. The results are discussed in Section 4. Section 5 summarizes our conclusions and future work.

## 2. BACKGROUND

### 2.1. Uncertainty decoding

In GMM based ASR, decoding requires the log-likelihood of all states (senones) in each time frame. DNN based ASR operates on pseudo log-likelihoods instead: the logarithm of the softmax normal-

ization constant and the clean feature log-prior are removed during decoding as they are constant for all states. Thus for a DNN of $K$ hidden layers, the pseudo log-likelihood of a clean feature vector $\mathbf{y}$ given the $i$-th state $s_i$ is given by

$$\Upsilon^{\text{pseudo}} = \log p_{\text{pseudo}}(\mathbf{y}|s_i) = x_i^{K+1} - \log p(s_i) \quad (1)$$

with $x_i^{K+1}$ the $i$-th pre-activation of the DNN output layer and $p(s_i)$ the prior probability of $s_i$ [18].

In the context of noisy uncertain data, the posterior probability $p(\mathbf{y}|\widehat{\mathbf{y}}, \sigma^{\mathbf{2}}_{\widehat{\mathbf{y}}})$ of the clean features is observed instead of deterministic features. This distribution is assumed to be Gaussian with diagonal covariance. The mean $\widehat{\mathbf{y}}$ and the diagonal $\widehat{\boldsymbol{\sigma}}^{\mathbf{2}}_{\mathbf{y}}$ of the covariance represent the enhanced feature vector and the uncertainty, that is the variance of residual speech distortions after enhancement, respectively. In [18], two approaches were proposed to exploit the uncertainty in the decoding.

#### *2.1.1. Log-likelihood marginalization*

The log-likelihood marginalization approach consists of computing the conditional expectation of the pseudo log-likelihood in (1) given the enhanced features and the uncertainty

$$\Upsilon^{\text{LM}} = E[x_i^{K+1}|\widehat{\mathbf{y}}, \widehat{\boldsymbol{\sigma}}^{\mathbf{2}}_{\mathbf{y}}] - \log p(s_i). \quad (2)$$

To do so, the feature uncertainty distribution $p(\mathbf{y}|\widehat{\mathbf{y}}, \widehat{\boldsymbol{\sigma}}^{\mathbf{2}}_{\mathbf{y}})$ must be propagated up to the pre-activations of the output layer.

#### *2.1.2. Posterior marginalization*

The posterior marginalization approach computes the conditional expectation of the posterior $p(s_i|\mathbf{y})$ instead:

$$p(s_i|\widehat{\mathbf{y}}, \widehat{\boldsymbol{\sigma}}^{\mathbf{2}}_{\mathbf{y}}) = E[p(s_i|\mathbf{y})|\widehat{\mathbf{y}}, \widehat{\boldsymbol{\sigma}}^{\mathbf{2}}_{\mathbf{y}}] = E[h_i^{K+1}|\widehat{\mathbf{y}}, \widehat{\boldsymbol{\sigma}}^{\mathbf{2}}_{\mathbf{y}}] \quad (3)$$

with $h_i^{K+1}$ the output of the softmax layer. The pseudo log-likelihood is thus modified as

$$\Upsilon^{\text{PM}} = \log p(s_i|\widehat{\mathbf{y}}, \widehat{\boldsymbol{\sigma}}^{\mathbf{2}}_{\mathbf{y}}) - \log p(s_i). \quad (4)$$

The feature uncertainty distribution $p(\mathbf{y}|\widehat{\mathbf{y}}, \widehat{\boldsymbol{\sigma}}^{\mathbf{2}}_{\mathbf{y}})$ must then be propagated through the DNN output layer too.

### 2.2. Propagation through each layer

The PIE approximation propagates the first and second order statistics of the pre-activations layer-by-layer by approximating nonlinear sigmoid activations by the sum of two exponential functions as detailed in [17]. The computation of the cumulative density function for multivariate Gaussians of very large dimension is not trivial, especially when the covariance matrix is not diagonal. To address this issue, the off-diagonal elements of the covariance matrix of the hidden layer pre-activations are neglected, which causes estimation errors that propagate through the DNN layers.

### 2.3. Propagation through the entire DNN

#### *2.3.1. Monte Carlo (MC) sampling*

MC sampling achieves posterior marginalization by randomly drawing a number of samples from the feature uncertainty distribution and passing each sample through the DNN. The average of the resulting outputs approximates the posterior expectation (3) [18, 21].

#### *2.3.2. Unscented transform (UT)*

In the UT approach, the samples are not drawn randomly but according to a deterministic criterion. For $L$-dimensional feature vectors, $2L$+1 sample vectors with corresponding weights are computed [18, 25]. Each sample is then passed through the DNN and a weighted average of the resulting outputs is computed.

## 3. EXPERIMENTAL SETUP

In order to evaluate the performance of different uncertainty estimators and propagation techniques under different noise conditions, the CHiME-2 [23] and CHiME-3 [24] datasets are selected for this work. In this section, we present the datasets, the experimental setup, and the various uncertainty estimators considered.

### 3.1. Datasets

The CHiME-2 Track 2 dataset was created by convolving clean Wall Street Journal (WSJ0) utterances with binaural room impulse responses (BRIRs) and adding real background noise at six different SNR ranges centered around -6, -3, 0, 3, 6, and 9 dB. The BRIRs and the background noises were recorded in a domestic living room. The training set contains 7138 simulated noisy utterances spoken by 83 speakers. The development and test sets contain 2460 and 1980 simulated noisy utterances spoken by 10 and 8 speakers, respectively.

The CHiME-3 dataset provides both real and simulated recordings of WSJ0 utterances acquired by a tablet equipped with 6 microphones in four noise environments: bus, café, pedestrian area, and street. A total of 1600 real and 7138 simulated utterances from 87 speakers are used for training. In a similar fashion, the development set contains 1640 real and 1640 simulated utterances from 4 speakers, and the test set contains 1320 real and 1320 simulated utterances from 4 speakers.

For both datasets, the speakers and the noise signals in the training, development, and test sets are disjoint. The noise backgrounds in CHiME-2 are louder and more nonstationary than in CHiME-3.

### 3.2. Speech enhancement and ASR baseline

We enhanced the noisy development and test data using multichannel nonnegative matrix factorization (NMF) [26] as implemented in the FASST toolbox [27]. The speech model was trained on the training set (distinct models were trained for CHiME-2 and CHiME-3), while the noise model was trained on the noise preceding each utterance. We used the same settings as in [16, 18].

For each dataset, we trained DNN based acoustic models as follows. First, a GMM based acoustic model was trained on the clean training set using 40-dimensional feature space maximum likelihood linear regression (fMLLR) features. The trained GMM models were used to obtain senone level alignments for the clean training data, which were carried over to the simulated noisy training data. Indeed, the simulation process does not affect the alignment. For the real training data, alignments were obtained using enhanced data instead. Given these alignments, a DNN based acoustic models was trained on the noisy training set using 40-dimensional logmel features with 11-frame splicing (5 frames before and after the current frame). For CHiME-3, the full noisy training set (real and simulated) was used. The DNNs are composed of a 440-dimensional input layer followed by seven 2048-dimensional hidden layers. The output layer consists of 2000 and 1978 states for CHiME-2 and CHiME-3, respectively. The DNN parameters were initialized by restricted Boltzmann machine (RBM) pre-training and fine-tuned using stochastic gradient

descent (SGD). Decoding was performed on enhanced development and test sets using a trigram language model with 5k vocabulary size. No rescoring (using, e.g., neural network language models or sequence-level minimum Bayes risk) was performed.

The Kaldi speech recognition toolkit [28] was used. The above steps match the CHiME-2 and CHiME-3 recipes in Kaldi, except that we computed alignments from clean data instead of noisy data and we decoded enhanced development and test data. This resulted in a moderate improvement compared to the baseline results reported in [16, 24] when aligning and decoding noisy data.

### 3.3. Uncertainty estimation

#### *3.3.1. Oracle uncertainty (OU) estimator*

The OU estimator is the best possible uncertainty estimator when the information of clean data is known [16]. It is given by

$$\widehat{\boldsymbol{\sigma}}_{\mathbf{y}}^2 = (\widehat{\mathbf{y}} - \mathbf{y})^2 \tag{5}$$

where $\widehat{\mathbf{y}}$ and $\mathbf{y}$ are the enhanced and clean feature vectors, respectively, and the squared difference is taken elementwise. This estimator cannot be computed in practice when the clean data is unknown, but it provides a lower bound on the word error rate (WER) achievable via uncertainty decoding. For real CHiME-3 data, we computed the OU using "pseudo-clean" features obtained by least-squares subband filtering of the noisy signals using the signal recorded by a close-talk microphone as a reference, as described in [29].

#### *3.3.2. Kolossa's uncertainty (KU) estimator*

The KU estimator is obtained by assuming the uncertainty to be proportional to the squared difference between the enhanced features and the noisy features $\mathbf{z}$ [1, 18, 19]:

$$\widehat{\boldsymbol{\sigma}}_{\mathbf{y}}^2 = \alpha(\widehat{\mathbf{y}} - \mathbf{z})^2. \tag{6}$$

The constant $\alpha$ is found by minimizing the WER on development data. In the following, we use the value $\alpha = 0.4$ obtained in [18] for the CHiME-2 dataset enhanced by FASST.

#### *3.3.3. Neural network based uncertainty (NNU) estimator*

We propose to train a neural network (NN) to estimate the uncertainty in the feature domain. This NNU estimator is inspired from [16, 30] but it has greater learning ability due to its less shallow architecture. Also, by contrast with the binary mask regression tree estimator in [30], it operates on soft masks. We found these two changes to improve performance, however we do not provide a detailed evaluation here due to space limitations.

The inputs of the NN are 40-dimensional noisy logmel features concatenated with 40-dimensional soft mask features forming an 80-dimensional input vector. The soft mask features $\mathbf{w}$ are obtained as $\mathbf{w} = \exp(\widehat{\mathbf{y}} - \mathbf{z})$. The outputs are 40-dimensional uncertainty vectors $\widehat{\boldsymbol{\sigma}}_{\mathbf{y}}^2$. Training is performed on the training set (distinct NNs are trained for CHiME-2 and CHiME-3), using oracle uncertainty vectors as training targets. The NN contains two hidden layers with sigmoid activation units. The hidden layers have a dimension of 500 and are initialized by RBM pre-training. The output layer is also passed through a sigmoid to limit the output range between 0 and 1. The oracle uncertainties which are used to train the DNN are scaled across each dimension by the corresponding maximum value over the training set to limit the range between 0 and 1. The same scale factor is used to scale the estimated uncertainty during the testing phase. The mean square error (MSE) is used as an objective function and the weights of the network are fine-tuned using SGD.

For all considered estimators, the DNN uncertainty propagation toolbox in [18][1] is used to perform PIE, MC, and UT. The 40 dimensional uncertainty vectors are spliced in the same way as the features themselves (using 11 frames).

| Uncertainty | | CHiME-2 | CHiME-3 | |
|---|---|---|---|---|
| **Est.** | **Prop.** | **Test** | **Test Real** | **Test Simu** |
| No uncertainty | | **24.53** | **29.59** | **16.82** |
| OU | PIE | 23.20 | 27.74 | 16.85 |
| | MC | 19.80 | 24.71 | 13.50 |
| | UT | **18.60** | **24.22** | **13.44** |
| KU | PIE | 26.89 | 31.10 | 18.66 |
| | MC | 23.31 | 27.26 | 15.46 |
| | UT | 22.22 | 26.98 | 15.50 |
| NNU | PIE | 25.73 | 29.71 | 18.35 |
| | MC | 22.33 | 25.41 | **14.79** |
| | UT | **21.46** | **25.26** | 14.89 |

**Table 1**. Average WER (%) on the CHiME-2/3 test sets.

## 4. RESULTS AND DISCUSSION

In the following, we analyze the WER achieved using the above uncertainty estimation and propagation techniques. In each table, we highlight three sets of results: the WER achieved without uncertainty (baseline), the best WER achieved using OU (lower bound), and the best WER achieved using KU or NNU (actual result).

### 4.1. Overall results

Table 1 summarizes the results on average over all data conditions. The absolute WERs on CHiME-2 and CHiME-3 Real/Simu cannot be compared because the test data and the speech enhancement performance differ. We focus on the relative WER improvement brought by uncertainty estimation and propagation techniques compared to the baseline. We make the following observations.

First, as previously noted in [18], PIE performs poorly for all datasets and all uncertainty estimators. This is due both the approximation involved at each layer and to the fact that it marginalizes the log-likelihood instead of the posterior. UT performs better than MC on CHiME-2 and similarly on CHiME-3.

Second, aside from PIE, uncertainty decoding provides a consistent improvement (the WER always improves) that is roughly comparable across datasets for real vs. simulated data.

Third, when used with the best propagation technique, the OU estimator improves performance by 18 to 20% relative compared to the baseline. This shows that significant improvements can potentially be achieved by uncertainty decoding for a variety of data.

Fourth, the proposed NNU estimator outperforms KU for all datasets and propagation techniques tested. When used with the best propagation technique, KU improves the WER by 5 to 9% relative, compared to 10 to 15% relative for NNU. The latter is roughly double the improvement achieved by KU and 50 to 80% of the improvement achieved by OU, which shows that NNU is able to account for a significant proportion of the actual uncertainty in the data.

The three latter findings are novel in the context of uncertainty propagation for DNN acoustic models.

[1] https://github.com/makladios/Kaldi_Matlab_DNN_UP

| Uncertainty | | Test Set | | | | | | Development Set | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Est.** | **Prop.** | **-6dB** | **-3dB** | **0dB** | **3dB** | **6dB** | **9dB** | **-6dB** | **-3dB** | **0dB** | **3dB** | **6dB** | **9dB** |
| No uncertainty | | **40.11** | **31.01** | **24.88** | **20.42** | **15.99** | **14.78** | **47.22** | **37.20** | **29.57** | **25.33** | **21.91** | **18.88** |
| OU | PIE | 39.95 | 30.42 | 23.13 | 19.18 | 14.02 | 12.55 | 47.10 | 36.65 | 28.03 | 24.09 | 21.12 | 17.64 |
| | MC | 35.44 | 26.19 | 19.04 | 15.88 | 11.39 | 10.87 | 42.55 | 32.05 | 24.00 | 20.17 | 17.10 | 14.49 |
| | UT | **33.21** | **24.99** | **18.00** | **15.19** | **10.77** | **9.48** | **41.13** | **30.20** | **23.31** | **19.49** | **16.40** | **13.18** |
| KU | PIE | 44.67 | 34.59 | 26.08 | 23.10 | 17.43 | 15.49 | 48.03 | 38.79 | 30.33 | 26.45 | 23.82 | 19.54 |
| | MC | 38.05 | 30.43 | 23.00 | 19.04 | 15.03 | 14.31 | 44.52 | 35.84 | 27.02 | 24.11 | 19.95 | 17.85 |
| | UT | 37.43 | **28.04** | 21.88 | 18.53 | 14.19 | 13.28 | 44.02 | 34.99 | 27.01 | 23.35 | 19.51 | 17.73 |
| NNU | PIE | 42.52 | 33.69 | 25.93 | 21.03 | 15.41 | 15.84 | 47.10 | 38.44 | 29.91 | 26.52 | 22.73 | 19.78 |
| | MC | 38.01 | 30.22 | 20.23 | 18.48 | 14.37 | 12.70 | 44.69 | 35.91 | **26.19** | 22.59 | 19.02 | **16.30** |
| | UT | **36.76** | 28.74 | **20.11** | **17.53** | **13.48** | **12.19** | **42.91** | **33.94** | 26.21 | **22.04** | **18.96** | 16.35 |

**Table 2**. Detailed WER (%) per SNR condition on the CHiME-2 test and development sets.

| Uncertainty | | Test Set | | | | | | | | Development Set | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Real | | | | Simulated | | | | Real | | | | Simulated | | | |
| **Est.** | **Prop.** | **BUS** | **CAF** | **PED** | **STR** | **BUS** | **CAF** | **PED** | **STR** | **BUS** | **CAF** | **PED** | **STR** | **BUS** | **CAF** | **PED** | **STR** |
| No uncert. | | **42.89** | **29.81** | **24.38** | **21.31** | **15.88** | **18.17** | **17.44** | **15.81** | **20.72** | **17.00** | **14.23** | **14.12** | **10.77** | **11.22** | **12.99** | **13.15** |
| OU | PIE | 38.29 | 30.02 | 23.19 | 19.46 | 15.91 | 17.99 | 17.30 | 16.23 | 19.44 | 14.59 | 12.95 | 14.48 | 11.58 | 14.72 | 14.87 | 15.12 |
| | MC | **34.08** | 26.34 | **19.62** | 18.80 | 13.22 | **15.00** | 14.79 | **11.00** | 16.31 | 13.78 | 12.04 | **12.77** | 9.00 | 10.72 | **10.75** | **12.00** |
| | UT | 34.19 | **25.29** | 19.72 | **17.69** | **12.63** | 15.42 | **14.42** | 11.29 | **15.72** | **13.67** | **12.02** | 12.84 | **8.84** | **10.63** | **10.75** | 12.05 |
| KU | PIE | 41.44 | 34.29 | 26.10 | 22.55 | 17.95 | 18.94 | 18.94 | 18.79 | 19.94 | 17.33 | 14.91 | 16.00 | 12.51 | 15.88 | 16.30 | 17.45 |
| | MC | 38.12 | 28.59 | 22.65 | 19.68 | 14.71 | 18.00 | 15.94 | 13.21 | 17.22 | 16.66 | 12.31 | 13.40 | 10.55 | **11.20** | 12.11 | 12.81 |
| | UT | 37.91 | 28.01 | 22.90 | 19.11 | **14.50** | 17.72 | 16.00 | 13.79 | 17.89 | 16.08 | **12.00** | 13.81 | 10.22 | 11.52 | 12.50 | 12.41 |
| NNU | PIE | 39.77 | 31.51 | 25.80 | 21.26 | 17.64 | 18.50 | 18.21 | 19.05 | 20.21 | 16.66 | 15.00 | 15.77 | 12.00 | 15.89 | 16.58 | 17.00 |
| | MC | **35.19** | **26.70** | 20.34 | 19.43 | 14.83 | **15.94** | 15.67 | **12.75** | 16.55 | **14.34** | 12.30 | **13.20** | 9.66 | 11.51 | **11.16** | 12.29 |
| | UT | 35.63 | 26.73 | **20.08** | **18.61** | 14.61 | 16.02 | **14.95** | 14.00 | **16.19** | 15.57 | 12.10 | **13.20** | **9.60** | 11.51 | 11.88 | **12.13** |

**Table 3**. Detailed WER (%) per noise environment on the CHiME-3 test and development sets.

### 4.2. Impact of the SNR

Table 2 reports the WER as a function of the SNR in CHiME-2. Our previous findings still hold: for all SNRs, NNU outperforms KU except in one case and PIE performs poorly, while UT performs better or similarly to MC. Interestingly, the achieved WER improvement remains comparable or increases with the SNR. On the test set, the relative WER improvement achieved by NNU with UT increases from 8% at $-6$ dB to 18% at 9 dB SNR. On the development set, it is more stable across SNRs, from 9 to 13% relative. This behavior differs from GMM based acoustic models, for which the relative improvement achieved by uncertainty decoding decreases with the SNR. This can be explained by the fact that, as the SNR decreases, the uncertainty increases and therefore the approximations involved in MC and UT become coarser, while the closed-form uncertainty decoding rule for GMMs remains valid.

### 4.3. Impact of the noise environment

Table 3 reports the WER as a function of the noise environment in CHiME-3. These results are more difficult to analyze because different environments have not only different noise properties, but also different SNRs which cannot easily be told apart. Nevertheless, our previous findings still hold: NNU outperforms KU in 13 out of 16 cases and PIE performs poorly, while MC and UT perform similarly. For the real test set and for both the real and simulated development sets, the WER improvement achieved by NNU with MC or UT is larger for BUS and PED than for CAF or STR.

## 5. CONCLUSION AND FUTURE WORK

In this work, we investigated the performance of various DNN uncertainty estimation and propagation approaches in several experimental conditions. Furthermore, we proposed an NN based uncertainty estimator. In addition to confirming earlier findings in [18] regarding the performance of PIE, MC, and UP, we found that uncertainty decoding consistently improves the WER for both real and simulated data and for all noise conditions, and that the proposed NNU estimator generally outperforms the KU estimator. The achieved improvement depends on the noise condition and the SNR.

In future work, we will conduct a deeper analysis of the performance of uncertainty propagation as a function of the noise characteristics (e.g., nonstationarity), independently of the SNR. We will also seek to understand the reasons behind the lower WER improvement on the CHiME-2 development set w.r.t. the test set, and on the CHiME-3 simulated development set w.r.t. the real development set and the test set. Finally, we will extend these results to other DNN input features (e.g., fMLLR) and training conditions (e.g., training on clean or enhanced data). We would also investigate the effect of NNU estimator on CHiME-3 corpus when trained on CHiME-2 corpus and vice-versa.

## 6. ACKNOWLEDGMENTS

We acknowledge the support of Bpifrance (FUI voiceHome). Experiments presented in this paper were carried out using the Grid'5000 testbed, supported by a scientific interest group hosted by Inria and including CNRS, RENATER and several Universities as well as other organizations (see `https://www.grid5000.fr`).

## 7. REFERENCES


[1] D. Kolossa, R. F. Astudillo, E. Hoffmann, and R. Orglmeister, "Independent component analysis and time-frequency masking for speech recognition in multitalker conditions," *EURASIP Journal on Audio, Speech, and Music Processing*, vol. 2010, no. 1, pp. 1–13, 2010.

[2] L. Deng, "Front-end, back-end, and hybrid techniques for noise-robust speech recognition," in *Robust Speech Recognition of Uncertain or Missing Data*. Springer, 2011, pp. 67–99.

[3] J. A. Arrowood and M. A. Clements, "Using observation uncertainty in HMM decoding," in *Proc. Interspeech*, 2002, pp. 1561–1564.

[4] N. Becerra Yoma and M. Villar, "Speaker verification in noise using a stochastic version of the weighted Viterbi algorithm," *IEEE Transactions on Speech and Audio Processing*, vol. 10, no. 3, pp. 158–166, 2002.

[5] 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 Transactions on Speech and Audio Processing*, vol. 13, no. 3, pp. 412–421, 2005.

[6] H. Liao and M. Gales, "Joint uncertainty decoding for noise robust speech recognition," in *Proc. Interspeech*, 2005, pp. 3129–3132.

[7] V. Stouten and P. Wambacq, "Model-based feature enhancement with uncertainty decoding for noise robust ASR," *Speech Communication*, vol. 48, no. 11, pp. 1502–1514, 2006.

[8] M. Delcroix, T. Nakatani, and S. Watanabe, "Static and dynamic variance compensation for recognition of reverberant speech with dereverberation preprocessing," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 17, no. 2, pp. 324–334, 2009.

[9] M. Delcroix, S. Watanabe, T. Nakatani, and A. Nakamura, "Cluster-based dynamic variance adaptation for interconnecting speech enhancement pre-processor and speech recognizer," *Computer Speech & Language*, vol. 27, no. 1, pp. 350–368, 2013.

[10] L. Lu, K. Chin, A. Ghoshal, and S. Renals, "Joint uncertainty decoding for noise robust subspace Gaussian mixture models," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 21, no. 9, pp. 1791–1804, 2013.

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

[12] R. F. Astudillo and R. Orglmeister, "Computing MMSE estimates and residual uncertainty directly in the feature domain of ASR using STFT domain speech distortion models," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 21, no. 5, pp. 1023–1034, 2013.

[13] A. Ozerov, M. Lagrange, and E. Vincent, "Uncertainty-based learning of acoustic models from noisy data," *Computer Speech & Language*, vol. 27, no. 3, pp. 874–894, 2013.

[14] A. H. Abdelaziz, S. Zeiler, D. Kolossa, V. Leutnant, and R. Haeb-Umbach, "GMM-based significance decoding," in *Proc. ICASSP*, 2013, pp. 6827–6831.

[15] D. T. Tran, E. Vincent, and D. Jouvet, "Fusion of multiple uncertainty estimators and propagators for noise robust ASR," in *Proc. ICASSP*, 2014, pp. 5512–5516.

[16] ——, "Nonparametric uncertainty estimation and propagation for noise robust ASR," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 23, no. 11, pp. 1835–1846, 2015.

[17] R. F. Astudillo and J. P. da Silva Neto, "Propagation of uncertainty through multilayer perceptrons for robust automatic speech recognition," in *Proc. Interspeech*, 2011, pp. 461–464.

[18] A. H. Abdelaziz, S. Watanabe, J. R. Hershey, E. Vincent, and D. Kolossa, "Uncertainty propagation through deep neural networks," in *Proc. Interspeech*, 2015, pp. 3561–3565.

[19] Y. Tachioka and S. Watanabe, "Uncertainty training and decoding methods of deep neural networks based on stochastic representation of enhanced features," in *Proc. Interspeech*, 2015, pp. 3541–3545.

[20] R. F. Astudillo, A. Abad, and I. Trancoso, "Accounting for the residual uncertainty of multi-layer perceptron based features," in *Proc. ICASSP*, 2014, pp. 6859–6863.

[21] C. Huemmer, R. Maas, A. Schwarz, R. F. Astudillo, and W. Kellermann, "Uncertainty decoding for DNN-HMM hybrid systems based on numerical sampling." in *Proc. Interspeech*, 2015, pp. 3556–3560.

[22] N. Parihar, J. Picone, D. Pearce, and H. G. Hirsch, "Performance analysis of the Aurora large vocabulary baseline system," in *Proc. EUSIPCO*, 2004, pp. 553–556.

[23] E. Vincent, J. Barker, S. Watanabe, J. Le Roux, F. Nesta, and M. Matassoni, "The second 'CHiME' speech separation and recognition challenge: An overview of challenge systems and outcomes," in *Proc. ASRU*, 2013, pp. 162–167.

[24] J. Barker, R. Marxer, E. Vincent, and S. Watanabe, "The third 'CHiME' speech separation and recognition challenge: Analysis and outcomes," *Computer Speech and Language*, to appear.

[25] S. J. Julier and J. K. Uhlmann, "New extension of the Kalman filter to nonlinear systems," in *Proc. AeroSense*, 1997, pp. 182–193.

[26] A. Ozerov, E. Vincent, and F. Bimbot, "A general flexible framework for the handling of prior information in audio source separation," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, no. 4, pp. 1118–1133, 2012.

[27] Y. Salaün, E. Vincent, N. Bertin, N. Souviraà-Labastie, X. Jaureguiberry *et al.*, "The flexible audio source separation toolbox version 2.0," in *ICASSP Show & Tell*, 2014.

[28] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek *et al.*, "The Kaldi speech recognition toolkit," in *Proc. ASRU*, 2011, pp. 1–4.

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

[30] S. Srinivasan and D. Wang, "Transforming binary uncertainties for robust speech recognition," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 7, pp. 2130–2140, 2007.