Recognition of Convolutive Speech Mixtures by Missing Feature Techniques for ICA

Dorothea Kolossa*†, Hiroshi Sawada†, Ramon Fernandez Astudillo*, Reinhold Orglmeister* and Shoji Makino†
*Electronics and Medical Signal Processing, TU Berlin
10587 Berlin, Germany
Email: d.kolossa@ee.tu-berlin.de
†NTT Communication Science Laboratories, NTT Corporation
2-4 Hikaridai, Seika-cho, Soraku-gun, Kyoto 619-0237, Japan
Email: {sawada,maki}@cslab.kecl.ntt.co.jp

***Abstract*— One challenging problem for robust speech recognition is the cocktail party effect, where multiple speaker signals are active simultaneously in an overlapping frequency range. In that case, independent component analysis (ICA) can separate the signals in reverberant environments, also. However, incurred feature distortions prove detrimental for speech recognition. To reduce consequential recognition errors, we describe the use of ICA for the additional estimation of uncertainty information. This information is subsequently used in missing feature speech recognition, which leads to far more correct and accurate recognition also in reverberant situations at RT_{60} = 300ms.**

I. Introduction

The application of speech recognition in real-world situations still suffers from insufficient robustness. Many scenarios which appear interesting, for example the use of portable mobile devices which can accept speech input with sufficiently large vocabulary, to take notes, offer translations or navigation information, require close talking microphones for acceptable reliability. Especially when there is interference in a frequency range overlapping that of speech, such as babble noise, or when reverberation times are long, speech recognition is severely degraded.

In many such cases, the use of blind source separation techniques can be very beneficial to improve recognition results. However, more recent techniques, which apply nonlinear masking functions to improve separation results further, can actually deteriorate recognition performance due to the consequential feature distortions [1]. In the following, we describe an integrated approach for systematically dealing with these problems.

Toward that goal, first, ICA with time frequency masking is applied and in addition to the speech estimate itself, we also estimate the error variance incurred by time-frequency masking. This error estimate is transformed from the domain of preprocessing, in our case the short-time spectrum, to the domain of speech recognition, in our case the mel cepstrum domain. However, the described approach is capable of almost arbitrary, nonlinear transformations, giving great flexibility in the choice of preprocessing domain and recognition features. Finally, the error estimate is used by a missing feature speech recognizer to compensate for the feature distortions incurred by masking as shown in Figure 1.

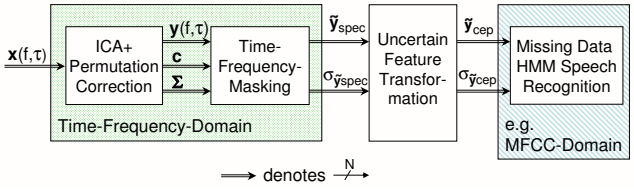

Fig. 1. Blockdiagram with Data Flow.

The following Sections II through V describe the algorithms used for obtaining the demixed signals, the time-frequency-mask and the estimated variance. Section VII continues by describing the transformation of the uncertain features to the recognition domain and the subsequent missing feature recognition. The test data and the experiments and results are detailed in Section VIII and conclusions are drawn in Section IX.

II. Independent Component Analysis

In order to deal with convolutive mixtures also, independent component analysis was carried out in the frequency domain. For this purpose, the STFT is computed of all microphone signals, so that the mixing system can be modeled by

$$\begin{aligned}\mathbf{X}(f,\tau) &\approx \mathbf{H}(f)\mathbf{S}(f,\tau)\\ &= \sum_{n=1}^{N} \mathbf{H}_{\cdot,n}(f) S_n(f,\tau) \end{aligned} \quad (1)$$

when the frames are chosen with sufficient length. $\mathbf{H}_{\cdot,n}(f)$ denotes the n'th column of the mixing matrix, and $S_n(f,\tau)$ is the value of source n in frequency f at time τ.

For the purpose of source separation, a complex valued FastICA [2] was computed and further improved by a natural gradient implementation of InfoMax [3], as described in more detail in [4], [5]. This results in an unmixing matrix $\mathbf{W}$, which is used to obtain estimated source signals via

$$\mathbf{Y}(f,\tau) = \mathbf{W}(f)\mathbf{X}(f,\tau). \quad (2)$$

The estimated sources can, due to the inherent ambiguities of ICA, be arbitrarily scaled and permutated versions of the true

source signals, which makes an additional permutation and scale correction necessary.

III. Basis Vector Clustering and Permutation Correction

In order to detect permutations, the effect of the estimated mixing matrix was considered,

$$\begin{aligned}\mathbf{X}(f,\tau) &= \mathbf{W}(f)^{-1}\mathbf{Y}(f,\tau) \\ &= [\mathbf{a}_1(f),\mathbf{a}_2(f),\ldots,\mathbf{a}_N(f)]\mathbf{Y}(f,\tau), \end{aligned} \quad (3)$$

where the estimated mixing matrix is written in terms of its constituent column vectors $\mathbf{W}^{-1} = [\mathbf{a}_1,\mathbf{a}_2,\ldots,\mathbf{a}_N]$. Comparing (3) with (1) shows that the columns of $\mathbf{W}^{-1}$ correspond to the columns of $\mathbf{H}(f)$, the matrix containing the values of the room transfer function for each frequency. Thus, the basis vectors $\mathbf{a}_n(f)$ should correspond to $\mathbf{H}_{\cdot,n}(f)$ in those time-frequency bins, where only source n is active. As long as the mixing process can be approximated by an anechoic model, the basis vectors should form clusters, one for each of the sources, after appropriate normalization. For this purpose, the basis vectors are normalized concerning both phase and amplitude. For phase normalization, they are first normalized with respect to a reference sensor K and secondly frequency normalized, which gives

$$\overline{a}_{ni}(f) = |a_{ni}(f)| \exp\left(j \frac{\arg\left(\frac{a_{ni}(f)}{a_{Ki}(f)}\right)}{4fc^{-1}d_{max}} \right);\ n,i = 1\ldots N \quad (4)$$

as a normalized vector that only varies in phase between

$$-\frac{\pi}{2} \leq \arg(\overline{a}_{ni}(f)) \leq \frac{\pi}{2}, \quad (5)$$

which is important for computing a distance measure between vectors [5]. Here, c is the velocity of sound and d_{max} stands for the greatest inter-sensor distance between the reference sensor K and all other microphones $n = 1\ldots N$. Finally, amplitude normalization takes place according to

$$\overline{\mathbf{a}}_i(f) = \frac{[\overline{a}_{1i}(f),\overline{a}_{2i}(f),...,\overline{a}_{Ni}(f)]^T}{||\overline{\mathbf{a}}_i(f)||}. \quad (6)$$

$||\cdot||$ denotes the Euclidean norm of the basis vectors. After the normalized basis vectors $\mathbf{a}_n(f)$ are thus determined, clustering of basis vectors takes places in order to determine one cluster for each of the active sources. For this purpose, a k-means algorithm is employed, which results in estimated clusters $C_1\ldots C_N$, associated with each of the N sources. The centroid of each cluster n is determined via

$$\mathbf{c}_n \leftarrow \sum_{\overline{\mathbf{a}}\in C_n} \frac{\overline{\mathbf{a}}}{|C_n|}, \quad \mathbf{c}_n \leftarrow \frac{\mathbf{c}_n}{||C_n||} \quad (7)$$

with $|C_n|$ denoting the number of vectors in the n'th cluster [5].

At each frequency f, ICA produces a set of basis vectors $\overline{\mathbf{a}}_n(f)$ as components of the inverse unmixing matrix $\mathbf{W}^{-1}$. Permutation correction consists of re-ordering these column vectors in such a way, that the result corresponds best with the order of cluster centers C_n, where the degree of correspondence is measured by computing the sum of euclidean distances between re-ordered basis vectors and cluster centers.

IV. Time-Frequency Masking

After permutation correction, a time-frequency mask $\mathcal{M}$ is applied according to

$$\tilde{y}_n(f,\tau) = \mathcal{M}_n(f,\tau) y_n(f,\tau) \quad (8)$$

to improve separation results further. The mask is a soft mask, whose value is based on the angle $\theta_n(t,\tau)$ between the observed vector $\mathbf{X}(f,\tau)$ and the basis vector $\mathbf{a}_n(f)$ associated with source n. This angle is computed in a whitened space, where $\mathbf{X}(f,\tau)$ and $\mathbf{a}(f)$ are premultiplied by a whitening matrix $\mathbf{V}$, which is the inverse square root of the sensor autocorrelation matrix, $\mathbf{V}(f) = \mathbf{R}_{xx}^{-1/2}$. From $\theta_n(t,\tau)$, the mask is determined by the logistic function

$$\mathcal{M}_n(\theta_n) = \frac{1}{1+\mathrm{e}^{g(\theta_n-\theta_T)}}. \quad (9)$$

Here, g describes the steepness of the mask and θ_T is the transition point, where the mask takes on the value $\frac{1}{2}$. More details on the mask computation can be found in [5].

V. Estimation of Variance Information

In order to estimate the feature uncertainty, two cases were distinguished. If the speaker under consideration, e.g. speaker n, is active in the given time-frequency bin, the observed vector $\mathbf{X}(f,\tau)$ should correspond well with the cluster of basis vectors $\mathbf{c}_n(f)$ of this speaker, after it has been normalized according to (4) and (6) to yield $\overline{\mathbf{X}}(f,\tau)$. On the other hand, if speaker n is inactive at (f,τ), the normalized $\overline{\mathbf{X}}(f,\tau)$ will likely correspond to another cluster center $\mathbf{c}_j(f)$. Thus, an hypothesis $\mathcal{H}_n$ was defined as "speaker n is dominant at (f,τ)", and a decision regarding its value was made for each speaker according to

$$H_i = p(s_i \text{ is dominant}) > p(s_j \text{ is dominant}) \quad \forall j \neq i. \quad (10)$$

The probability of each speaker's dominance was assessed by means of a Gaussian model for the basis vector clusters

$$p(\overline{\mathbf{X}}(\omega,t)|C_i) = \mathcal{N}(\overline{\mathbf{X}}(\omega,t),\mathbf{c}_i,\mathbf{\Sigma}_i), \quad (11)$$

with $\mathcal{N}$ as a Gaussian distribution of $\overline{\mathbf{X}}$ whose mean and covariance parameters are $\mathbf{c}_i$ and $\mathbf{\Sigma}_i$, the center and variance of cluster C_i. This model was used to calculate

$$\begin{aligned} &p(s_i \text{ is dominant}) \approx \\ &\quad p(C_i|\overline{\mathbf{X}}(\omega,t)) = \frac{p(\overline{\mathbf{X}}(\omega,t)|C_i)P(C_i)}{\sum_{j=1}^{N} p(\overline{\mathbf{X}}(\omega,t)|C_j)P(C_j)}. \end{aligned} \quad (12)$$

Based on the value of the hypothesis $\mathcal{H}_n$, the error of the estimate $\tilde{y}_n(f,\tau)$ was computed, under the assumptions that

- in periods of speaker activity, errors are underestimation errors due to excessive masking, whereas
- in periods, where the speaker is dominated by another signal, the error is overestimation due to insufficient masking.

These errors were estimated separately by

$$\sigma_{\tilde{y}_n,o} = y_n(f,\tau) \tag{13}$$

for overestimation and

$$\sigma_{\tilde{y}_n,u} = y_n(f,\tau) - \tilde{y}_n(f,\tau) \tag{14}$$

for underestimation errors and combined to

$$\sigma_{\tilde{y}_n} = \sigma_{\tilde{y}_n,o}\overline{\mathcal{H}}_n + \sigma_{\tilde{y}_n,u}\mathcal{H}_n. \tag{15}$$

VI. Propagating Variance Information to Speech Recognition Feature domain

The output of source separation consists of the estimated speech features $\tilde{y}_{spec}(f,\tau)$ and an associated variance estimate $\sigma_{\tilde{y}_{spec}}(f,\tau)$ for each source. This section describes how this data is processed to obtain the features in the mel cepstrum domain, $\tilde{y}_{cep}(t,\tau)$, together with associated variance estimates $\sigma_{\tilde{y}_{cep}}(t,\tau)$. However, the methods used here are not limited to this specific set of recognition features but can be used for a wide set of linear as well as nonlinear feature transforms, so that speech processing and speech recognition can be carried out in domains which are related to each other by almost arbitrary transforms[1], while still passing variance values from the preprocessing to the recognition stage. Since the transformation between the complex speech spectrum and the mel cepstrum, which is used here, consists of linear as well as nonlinear transformation stages, and since analytic calculations are simple for the linear and computationally intensive for the nonlinear transforms, the feature transformation was carried out stage by stage.

In linear stages of transformation, the analytic solution is easily computed. For a vector valued Gaussian random variable $\mathbf{m}$ which is transformed linearly to $\mathbf{n} = \mathbf{T}\mathbf{m}$, the mean and the covariance of the transformed variable $\mathbf{n}$ are

$$\mu_{\mathbf{n}} = \mathbf{T}\mu_{\mathbf{m}} \tag{16}$$

and

$$\Sigma_n = \mathbf{T}\Sigma_m\mathbf{T}^T. \tag{17}$$

This result also holds for mixtures of Gaussian densities, when means and covariances are computed separately for each component. Thus, means and variances can be easily computed for the linear transformation stages, and only the nonlinear stages need to be considered in more detail.

A. Analytical Solution

When a random variable $\mathbf{v}_1$ is transformed nonlinearly to $\mathbf{v}_2 = T(\mathbf{v}_1)$, the resulting probability distribution can be found by first computing the cumulative distribution of $\mathbf{v}_2$ via

$$P(\mathbf{v}_2 < V) = \int_{\mathbf{v}_1:T(\mathbf{v}_1)<V} p_{v1}(\mathbf{v}_1)d\mathbf{v}_1. \tag{18}$$

[1]Recognizer features $\mathbf{v}_2$ can be arbitrary, nonlinear functions of preprocessing features $\mathbf{v}_1$ at time t. They can also be functions of $\mathbf{v}_1(t), \mathbf{v}_1(t-1), \ldots, \mathbf{v}_1(t-k)$, as long as it is possible to give a finite state space representation of the transformation. In contrast, transformations with hysteresis do not fit into the framework.

The derivative of this cumulative distribution is the desired probability density $p_{v2}(\mathbf{v}_2)$. However, we are interested only in the first two moments of the output distribution. Thus, computing the entire pdf is not necessary, and rather, the output statistics can estimated directly via

$$\mu_{v2} = \int_{-\infty}^{\infty} T(\mathbf{v}_1)p_{v1}(\mathbf{v}_1)d\mathbf{v}_1 \tag{19}$$

and

$$\sigma_{v2} = \sqrt{\int_{-\infty}^{\infty} (T(\mathbf{v}_1) - \mu_{v2})^2 p_{v1}(\mathbf{v}_1)d\mathbf{v}_1}. \tag{20}$$

B. Unscented Transform

As a flexible approximation to the above analytical integration, which can be used with an almost arbitrary recognizer parameterization the *Unscented Transform* has been employed to compute the effect of the nonlinear transformations on the uncertain features. It consists of the following steps:

- Given the d-dimensional distribution of processing features $\mathbf{v}_1$, (for example the distribution of $\tilde{\mathbf{y}}$ in the spectral domain with $NFFT$ dimensions) a set of D so-called *sigma points* $\mathcal{P} = \{p_1, \ldots, p_D\}$ is calculated, which capture the statistics of the features up to the desired order.
- The sigma points are propagated through the nonlinearity to form a set of transformed points $\mathcal{Q} = g(\mathcal{P}) = \{q_1, \ldots, q_D\}$.
- The statistics of $\mathbf{v}_2 = g(\mathbf{v}_1)$ are then approximated up to the required order by the appropriate statistics of the transformed set $\mathcal{Q}$, e.g. by the mean $\overline{\mathcal{Q}}$ and covariance $\Sigma_{\mathcal{Q}\mathcal{Q}}$ as the first and second order statistics of $\mathbf{v}_2$, which here corresponds to the cepstral features $\tilde{\mathbf{y}}_{cep}$.

This approach is also illustrated in Figure 2.

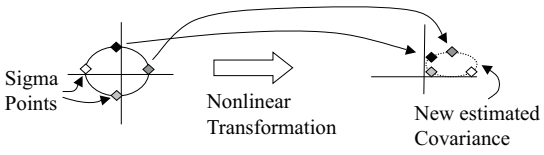

Fig. 2. The sigma points of the signal probability distribution are transformed to obtain an estimate of the statistics after the transformation.

Compared to Monte Carlo simulation, this algorithm has the advantage of efficiency: Whereas a large set of points needs to be simulated to obtain low errors in Monte Carlo simulation, using the unscented transform, only $2d+1$ points are simulated for each feature vector, where d is the feature vector size [6].

VII. Missing Feature Recognition

The proposed method is intended for speech recognition systems based on statistical speech models, which, in principle, may use features in any appropriate domain. In the case presented here, speech recognition is carried out via Hidden Markov Models using Mel-Frequency Cepstral Coefficients

(MFCCs). The recognizer is a Matlab implementation of a phoneme-based HMM recognizer, in which the output distributions are modeled by Mixture of Gaussian models.

A. Modified Imputation for Data with Variances

In HMM speech recognition, the probability of an observation vector $\mathbf{o}$ of speech features, e.g. $\tilde{\mathbf{y}}_{cep}$ in our case, is evaluated at each frame for all HMM-states. For this purpose, the model states q are equipped with output probability distributions denoted by b_q, where $b_j(\mathbf{o})$ gives the probability that observation vector $\mathbf{o}$ will occur at time t when the Markov Model is known to be in state j at that time, so:

$$b_j(\mathbf{o}) = p(o(t) = \mathbf{o}|q(t) = j). \tag{21}$$

For the recognition of given, fixed observation vectors $\mathbf{o}$, the probability distribution b_q can be evaluated for the available vector $\mathbf{o}$. This is the customary computation of output probabilities, denoted by $p_{\mathbf{o}|q}(\mathbf{o}|q)$. With additional information from the preprocessing stage, however, rather than only the observation $\mathbf{o}$, its entire pdf $p(\mathbf{o}|x)$ is approximately given. Thus, a new approach, termed *modified imputation* has been developed for calculating observation likelihoods, which makes use of all available information: output probability distributions of the states $p(\mathbf{o}|q)$ as well as the observation probability distributions obtained from the preprocessing stage, $p(\mathbf{o}|x)$.

To evaluate the likelihood of an HMM state, the likelihood of the current observation $p(\mathbf{o}|q)$ given q is combined with the likelihood of the observation given the preprocessing model $p(\mathbf{o}|x)$ via

$$\begin{aligned} p(\mathbf{o}|x,q) &= \frac{p(\mathbf{o},x,q)}{p(x,q)} \\ &= \frac{p(\mathbf{o},x|q)}{p(x|q)} \\ &= \frac{p(\mathbf{o}|q)p(x|\mathbf{o},q)}{\int_{\mathbf{o}} p(x|\mathbf{o},q)p(\mathbf{o}|q)d\mathbf{o}}. \end{aligned} \tag{22}$$

All statistical dependencies between the microphone signals x and the HMM state q are assumed to be captured in the feature vector $\mathbf{o}$. Therefore $p(x|\mathbf{o},q) = p(x|\mathbf{o})$ and

$$\begin{aligned} p(\mathbf{o}|x,q) &= \frac{p(\mathbf{o}|q)p(x|\mathbf{o})}{\int_{\mathbf{o}} p(x|\mathbf{o})p(\mathbf{o}|q)d\mathbf{o}} \\ &= \frac{p(\mathbf{o}|q)p(\mathbf{o}|x)p(x)}{p(\mathbf{o})\int_{\mathbf{o}} p(x|\mathbf{o})p(\mathbf{o}|q)d\mathbf{o}}. \end{aligned} \tag{23}$$

In (23), the integral in the denominator as well as $p(x)$ are independent of the feature vector $\mathbf{o}$. But since the equation will only be needed for the optimization problem stated in 25, they can be considered invariant scale factors. Defining a likelihood function p' via

$$p'(\mathbf{o}|x,q) = \frac{p(\mathbf{o}|q)p(\mathbf{o}|x)}{p(\mathbf{o})} \propto p(\mathbf{o}|x,q) \tag{24}$$

allows to estimate $\mathbf{o}$ by

$$\hat{\mathbf{o}} = \arg\max_{\mathbf{o}'} p(\mathbf{o}'|x,q) = \arg\max_{\mathbf{o}'} p'(\mathbf{o}'|x,q). \tag{25}$$

Assuming a uniform prior for $\mathbf{o}$, the term to be maximized is

$$\hat{\mathbf{o}} = \arg\max_{\mathbf{o}'} p(\mathbf{o}'|q)p(\mathbf{o}'|x). \tag{26}$$

For a Gaussian model, the optimization problem becomes

$$\hat{\mathbf{o}} = \arg\max_{\mathbf{o}'} e^{-\frac{1}{2}((\mathbf{o}'-\mu_x)^T\Sigma_x^{-1}(\mathbf{o}'-\mu_x)+(\mathbf{o}'-\mu_q)^T\Sigma_q^{-1}(\mathbf{o}'-\mu_q))}$$

and the maximum likelihood estimate $\hat{\mathbf{o}}$ can be obtained from

$$\begin{aligned} \Sigma_x^{-1}(\hat{\mathbf{o}}-\mu_x) &\stackrel{!}{=} -\Sigma_q^{-1}(\hat{\mathbf{o}}-\mu_q) \\ \Leftrightarrow \hat{\mathbf{o}} &= (\Sigma_x^{-1}+\Sigma_q^{-1})^{-1}(\mu_q\Sigma_q^{-1}+\mu_x\Sigma_x^{-1}). \end{aligned}$$

This resulting estimate $\hat{\mathbf{o}}$ can be used for recognition in the same way as $\mathbf{o}$ is used in (21) when the features are considered given and fixed. For Gaussian mixture models, the same computations need to be carried out for all mixtures separately.

B. Uncertainty Decoding

As an alternative to modified imputation, the use of *uncertainty decoding* was also investigated. In this method, described in [7], the original aim is for an improvement of noisy speech recognition. When speech is additively corrupted

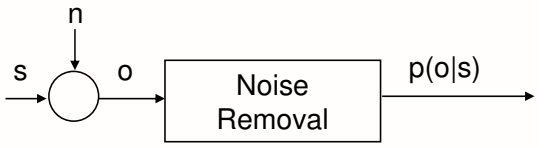

Fig. 3. Signal model for uncertainty decoding.

by zero-mean Gaussian noise, with the signal model shown in Figure 3, the probability of observing a feature vector $\mathbf{o}$ given the clean speech signal s is

$$\mathrm{p}(\mathbf{o}|s) = \mathcal{N}(\mathbf{o}, s, \sigma_n^2). \tag{27}$$

For the purpose of recognition, the distribution $\mathrm{p}(\mathbf{o}|q)$ is needed. This can be obtained from

$$\mathrm{p}(\mathbf{o}|q) = \int_{-\infty}^{\infty} \mathrm{p}(\mathbf{o},s|q)\mathrm{d}s \tag{28}$$

$$= \int_{-\infty}^{\infty} \mathrm{p}(\mathbf{o}|q,s)\mathrm{p}(s|q)\mathrm{d}s \tag{29}$$

$$= \int_{-\infty}^{\infty} \mathrm{p}(\mathbf{o}|s)\mathrm{p}(s|q)\mathrm{d}s. \tag{30}$$

When a single mixture i in the mixture model of state q is considered, this leads to

$$\mathrm{p}(\mathbf{o}|q) = \int_{-\infty}^{\infty} \mathcal{N}(\mathbf{o}, s, \sigma_n^2)\mathcal{N}(s, \mu_{qi}\sigma_{qi})\mathrm{d}s \tag{31}$$

$$= \mathcal{N}(\mathbf{o}, \mu_{qi}, \sigma_{qi}^2 + \sigma_n^2)\mathrm{d}s \tag{32}$$

which is evaluated for each mixture.

VIII. Experiments and Results

A. Speech Data

Recordings were made in an office room with dimensions of about 10m $\times$ 15m $\times$ 3.5m. The distance between each of the two loudspeakers and the array center was set to one meter. At this distance, the reverberation time was measured to be 300ms. Simultaneous speech signals of two male speakers of equal mean amplitude from the TIDigits database [8] were played back and recorded in two different setups of loudspeakers, with the angles of incidence, relative to broadside, once set to $(-45°, 25°)$ (Configuration A) and the other time to $(-10°, 25°)$ (Configuration B). Two microphones, Behringer ECM 8000, placed 2.1cm apart, were used to record the signals.

B. Recognition Performance Measurement

To measure recognition performance, the number of reference labels (R), substitutions (S), insertions (I) and deletions (D) is counted. Then, two criteria can be calculated:
The *correctness* is the percentage of correctly hypothesized words

$$PC = \frac{R - D - S}{R}. \tag{33}$$

Correctness has one disadvantage for judging ICA performance though. Since it ignores insertion errors, it will not penalize clear audibility of the interfering speaker during periods, when the desired speaker is silent. Therefore, a second important criterion is recognition accuracy, defined as

$$PA = \frac{R - D - S - I}{R}. \tag{34}$$

C. Reference Results

In order to evaluate the effect of ICA and of subsequent time-frequency-masking (TF-masking) on speech recognizer performance, the recognition correctness and accuracy were measured for both configurations. The results were obtained with a speaker independent recognizer trained on the clean TIDigits database [8], without adaptation to the room. For both scenarios, the recognition rate for clean, but reverberant, data was 97.1% PC and 94.1% PA. Results for the recordings and the ICA outputs are shown in Table I.

TABLE I
Average Correctness (PC) and Accuracy (PA) of overlapping and separated speech, without and with TF-masking.

	Config. A	Config. B
mixtures		
PC	67.5%	74.2%
PA	40.0%	41.5%
ICA only		
PC	84.2%	81.4%
PA	60.5%	57.5%
ICA + TF-Mask		
PC	93.6%	71.4%
PA	79.8%	59.7%

As can be seen, by time-frequency masking alone, the average correctness of the ICA results is not improved, and though accuracy gains are notable in some cases, they are not observed reliably. This is likely due to the nonlinear distortions caused by time-frequency masking, which are great enough to outweigh the separation gain of time-frequency masking.

D. Results with Variance Information

When variance information is used to aid in decoding, the accuracy as well as the correctness improves greatly for all configurations, as seen in Table II.

TABLE II
Recognition Rates of Time-Frequency-Masked Speech by using Feature Uncertainties

	Config. A	Config. B
Modified Imputation		
PC	94.1%	96.4%
PA	91.2%	85.0%
Uncertainty Decoding		
PC	94.7%	93.4%
PA	86.8%	84.0%

IX. Conclusions

A new framework has been presented for using uncertain features, derived from ICA with a probabilistic source activity model, to aid in recognition of overlapping speech in reverberant environments. Through use of the unscented transform, it becomes possible to use uncertainty information from the time-frequency domain to derive the values of uncertain features in a suitable feature domain for speech recognition. The use of this technique for combining time-frequency domain ICA with an MFCC speech recognizer has been demonstrated and resulted both in significantly increased correctness as well as accuracy.

References

[1] D. Kolossa and R. Orglmeister, "Separation and recognition of noisy, convolutive speech mixtures using time-frequency masking and missing data techniques," in *Proc. WASPAA 2005*, pp. 832–839.

[2] E. Bingham and A. Hyvärinen, "A fast fixed-point algorithm for independent component analysis of complex valued signals," *Int. J. Neural Systems*, vol. 10, no. 1, pp. 1–8, 2000.

[3] A. J. Bell and T. J. Sejnowski, "An information-maximisation approach to blind separation and blind deconvolution," *Neural Computation*, vol. 7, no. 6, pp. 1004–1034, 1995.

[4] H. Sawada, R. Mukai, S. Araki, and S. Makino, "Polar coordinate based nonlinear function for frequency domain blind source separation," *IEICE Trans. Fundam.*, vol. E86-A, no. 3, pp. 590–596, Mar 2003.

[5] H. Sawada, S. Araki, R. Mukai, and S. Makino, "Blind extraction of dominant target sources using ica and time-frequency masking," *IEEE Trans. Audio, Speech and Language Processing*, vol. 14, no. 6, pp. 2165–2173, 2006.

[6] S. Julier and J. Uhlmann, "A general method for approximating nonlinear transformations of probability distributions," *Technical Report, University of Oxford, UK*, 1996.

[7] J. Droppo, A. Acero, and L. Deng, "Uncertainty decoding with splice for noise robust speech recognition," in *Proc. ICASSP 2002*, vol. 1, 2002, pp. 57–60.

[8] LDC, "Tidigits speech database: Studio quality speaker-independent connected-digit corpus," 1993.