9

Error Concealment

Reinhold Haeb-Umbach and Valentin Ion

Abstract. In distributed and network speech recognition the actual recognition task is not carried out on the user's terminal but rather on a remote server in the network. While there are good reasons for doing so, a disadvantage of this client-server architecture is clearly that the communication medium may introduce errors, which then impairs speech recognition accuracy. Even sophisticated channel coding cannot completely prevent the occurrence of residual bit errors in the case of temporarily adverse channel conditions, and in packet-oriented transmission packets of data may arrive too late for the given real-time constraints and have to be declared lost. The goal of error concealment is to reduce the detrimental effect that such errors may induce on the recipient of the transmitted speech signal by exploiting residual redundancy in the bit stream at the source coder output. In classical speech transmission a human is the recipient, and erroneous data are reconstructed so as to reduce the subjectively annoying effect of corrupted bits or lost packets. Here, however, a statistical classifier is at the receiving end, which can benefit from knowledge about the quality of the reconstruction. In this book chapter we show how the classical Bayesian decision rule needs to be modified to account for uncertain features, and illustrate how the required feature posterior density can be estimated in the case of distributed speech recognition. Some other techniques for error concealment can be related to this approach. Experimental results are given for both a small and a medium vocabulary recognition task and both for a channel exhibiting bit errors and a packet erasure channel.

9.1 Introduction

In a client-server speech recognition system the client, e.g. a cellular phone, captures the speech signal, codes it and sends it via a digital communication link to the remote recognition server. At the server side, the received signal is decoded and forwarded to the speech recognition engine, which outputs the decoded word string. Depending on the type of data transmitted, one distinguishes between *distributed* (DSR) and *network speech recognition* (NSR). In DSR speech recognition features, such as Mel-Frequency Cepstral Coefficients (MFCC), are computed, coded and transmitted

(Pearce 2000), while in NSR a typical speech codec, such as the adaptive multi-rate (AMR) codec is employed (Fingscheidt, Aalburg, Stan and Beaugeant 2002).

Compared to a realization of the recognizer on the client, the client-server architecture has many obvious advantages, such as ease of maintainability of the application data on the server and avoidance of resource-intensive tasks on the client. However, the price to pay is an additional processing delay due to transmission and the potential corruption of the digitized speech data due to channel-induced errors. Here we are concerned with the latter and show how error concealment techniques help mitigate the negative effects of transmission errors on the speech recognition accuracy.

Two channel models exhibiting different error types are considered in the following: a channel characterized by bit errors and a *packet erasure channel*. Channel degradations at the bit level are for example typical of cellular circuit-switched transmission, where noise, multi-path fading and interference from neighboring stations are frequent error causes. Packet loss is a typical phenomenon of packet-based transmission of data with real-time constraints over the internet. The combination of both error types is an approximate model for communications over a wireless packet network, or communications that involve both a wireless and a (packet-based) wireline link (Lahouti and Khandani 2007).

To mitigate transmission errors researchers have proposed several different approaches. One category is comprised of methods that perform error or packet-loss concealment techniques at the receiving end. Another class of techniques requires certain coordination with the transmitter side, e.g. forward error correction or diversity schemes based on multiple description coding. A third category requires a certain degree of support from the network, such as using packets with different priorities. The schemes rely on the network to drop the packets with low priority during congestion periods. Currently, this support, however, may only be available in proprietary networks and in the next generation of the Internet Protocol (IPv6) (RFC 2460, 1998).

In this contribution we restrict ourselves to purely receiver (server) based techniques which leave the transmitter (client) side untouched, since they have the striking advantage that they are fully compatible with the current European Telecommunications Standards Institute (ETSI) standards for distributed speech recognition (ETSI 2002; ETSI 2003a) and can be readily applied in current networks. Actually, the frame repetition scheme proposed in the ETSI standard is an example of such an error concealment method.

The term *error concealment* denotes techniques which aim to reduce or even eliminate the effect of uncorrected transmission errors on the quality as perceived by the consumer of the transmitted data. For data transmission with no latency constraints a virtually error-free transmission can be achieved by a combination of forward and backward error correction. This no longer holds for speech, audio or video transmissions, which typically have to adhere to real-time constraints.

The detrimental effect of transmission errors can be concealed by exploiting residual redundancy still present at the output of the (in the Shannon sense) imperfect source coder. One might argue that, since low-bit rate source coding has been an issue since the early days of digital speech transmission, it is unlikely to find enough

residual redundancy in the output bit stream of the speech coder to be exploited for error concealment. But even for the low-rate codes used in GSM successful error concealment based on exploiting the non-uniform bit pattern probabilities and the correlation between successive frames has been demonstrated (Fingscheidt and Vary 2001, Lahouti et al. 2007).

Error concealment has been studied extensively in the field of mobile communications and, more recently, for voice or other real-time data transmission over the internet protocol (IP). In cellular systems standards such as GSM error concealment algorithms are proposed as non-mandatory recommendations (GSM 1992), and very sophisticated techniques have been developed in recent years (Vary and Martin 2006). In Voice-over-IP packet loss is a frequent phenomenon, which is addressed, among others, by replacing the missing segments of speech with estimates constructed form previous or future available speech segments. For example, a waveform substitution algorithm based on pitch detection has been proposed for G.711 pulse code modulation speech coding standard (ITU-T Recommendation G.711 1999). Packet loss concealment methods for code excited linear prediction (CELP)-based coders often replace the missing parameters with the corresponding parameters of the previous frame (Cox, Kleijn and Kroon 1989) and use scaled-down gains. Methods that interpolate between previous and future frames can also be employed.

Similar techniques have been proposed for distributed speech recognition (DSR), where speech recognition related parameters, such as MFCCs (Davis and Mermelstein 1980) are computed in the user's terminal and then transmitted to the remote speech recognition engine (Tan, Dalsgaard and Lindberg 2005). Feature reconstruction techniques range from quite simple methods such as substitution (with silence, noise or source-data), repetition or interpolation (Boulis, Ostendorf, Riskin, and Otterson 2002; Milner and Semnani 2000) to more elaborated schemes, such as repetition on a subvector level (Tan, Dalsgaard and Lindberg 2004) and *minimum mean square error* (MMSE)-based reconstruction which models *inter-frame correlation* by a first-order Markov model (Peinado, Sanchez, Perez-Cordoba, and de la Torre 2003; James, Gomez and Milner 2004; Haeb-Umbach and Ion 2004). However, in DSR we can do even considerably more.

In a DSR scenario we would like to alleviate the effect that transmission errors have on the consumer of the data, the automatic speech recognition (ASR) decoder. Unlike a human recipient, the recognizer not only benefits from a good reconstruction of lost or corrupted data but also from knowledge about the quality of the reconstruction. The ASR decoder is then modified such that features deemed unreliable are deemphasized (Bernard and Alwan 2001; Bernard and Alwan 2002) or completely excluded from consideration in the recognizer (Weerackody, Reichl and Potamianos 2002; Endo, Kuroiwa and Nakamura 2003). However, it is not an easy task to identify corrupt features or even quantify the degree of corruption, at least on a channel exhibiting bit errors. While in (Haeb-Umbach et al. 2004) the availability of a soft-output channel decoder was assumed, in (Ion and Haeb-Umbach 2005) a technique was proposed which estimates bit error probabilities based on a priori knowledge of plausible bit patterns.

Actually, the close connection between feature reconstruction and modification of the decoding engine becomes apparent once the problem of speech recognition in

the presence of unreliable feature vectors is cast in a Bayesian framework. Here, results form noisy speech recognition can be borrowed, where so-called *Uncertainty Decoding* has been investigated already for a couple of years (Morris, Cooke and Green 1998; Morris, Barker and Bourlard 2001; Arrowood and Clements 2002; Droppo, Acero and Deng 2002; Kristjansson and Frey 2002). Let the speech be corrupted by additive noise or by transmission errors, in either case the original clean or uncorrupted speech feature vector is not observable, but rather a distorted version of it. Traditionally, the goal of speech feature enhancement is to obtain a point estimate of the clean speech feature, such as the MMSE estimate. This estimate is then "plugged into" the Bayes decision rule and used in the ASR decoder as if it were the true clean speech feature.

However, one can do better if one takes the reliability of the estimate into account. In one formulation of uncertainty decoding the probability density function of the corrupted speech feature vector, conditioned on the unobservable clean speech feature vector, is computed and averaged over the observation probability of the clean speech (Liao and Gales 2004). In another formulation the front-end delivers uncertain observations, expressed as a posteriori density of the clean speech feature vector, given the observed noisy vector. It is well known, that the mean of the posterior is exactly the MMSE estimate. Its variance is a measure of the uncertainty about this estimate. In the case of jointly Gaussian random variables, it is even equal to the variance of the estimation error. This frame-level uncertainty can be incorporated in the decoding process by using a modified Bayesian decision rule, where integration over the uncertainty in the feature space is carried out. Under certain assumptions this can be accomplished by a simple modification of the means and variances of the observation probabilities.

In the context of distributed speech recognition the concept of uncertainty decoding has been proposed for the first time in (Haeb-Umbach et al. 2004). Here, inter-frame correlation has been identified as a major knowledge source which helps in reconstructing lost or corrupted features.

This book chapter is organized as follows. In the following section we present the probabilistic framework of speech recognition in the presence of corrupted observations. In section 9.3 this concept is applied to distributed speech recognition, where we consider channels characterized by either bit errors or packet loss. Experimental results, both for a small and a medium vocabulary recognition task, are given in section 9.4, followed by some conclusions drawn in section 9.5.

9.2 Speech Recognition in the Presence of Corrupted Features

9.2.1 Modified Observation Probability

The *Bayesian decision rule* is at the heart of statistical speech recognition. Given the sequence of T (uncorrupted) feature vectors $\mathbf{x}_1^T = (x_1, \ldots, x_T)$ extracted from an utterance, the goal is to find the sequence of words $\hat{\mathbf{W}}$ from of a given vocabulary, which maximizes the probability $P(\mathbf{W} \mid \mathbf{x}_1^T)$. Using the Bayesian theorem for condi-

tional probabilities this can be expressed more conveniently as maximizing the product between observation probability $p\left(\mathbf{x}_1^T \mid \mathbf{W}\right)$ and word sequence probability $P\left(\mathbf{W}\right)$:

$$\hat{\mathbf{W}} = \arg\max_{\mathbf{W}} \left\{ p\left(\mathbf{x}_1^T \mid \mathbf{W}\right) \cdot P\left(\mathbf{W}\right) \right\}. \tag{1}$$

Introducing the hidden state sequence $s_1^T = (s_1, s_2, ..., s_T)$ we obtain

$$p\left(\mathbf{x}_1^T \mid \mathbf{W}\right) = \sum_{s_1^T} p\left(\mathbf{x}_1^T, s_1^T \mid \mathbf{W}\right) = \sum_{s_1^T} p\left(\mathbf{x}_1^T \mid s_1^T\right) P\left(s_1^T\right), \tag{2}$$

where the sum is over all state sequences within $\mathbf{W}$. As there is exactly one word sequence corresponding to a state sequence, the condition on $\mathbf{W}$ can be left out.

A common assumption employed in speech recognition is the so-called *conditional independence* assumption , which states that $\mathbf{x}_t$ is conditionally independent of neighboring feature vectors, given the HMM state s_t:

$$p\left(\mathbf{x}_1^T \mid s_1^T\right) = \prod_{t=1}^{T} p\left(\mathbf{x}_t \mid \mathbf{x}_1^{t-1}, s_t\right) = \prod_{t=1}^{T} p\left(\mathbf{x}_t \mid s_t\right). \tag{3}$$

Using this in Eq. 2 we obtain

$$p\left(\mathbf{x}_1^T \mid \mathbf{W}\right) = \sum_{s_1^T} \prod_{t=1}^{T} p\left(\mathbf{x}_t \mid s_t\right) P\left(s_1^T\right), \tag{4}$$

Often we are unable to observe the uncorrupted feature vector sequence $\mathbf{x}_1^T$. We observe a corrupted sequence $\mathbf{y}_1^T = \left(\mathbf{y}_1, ...\mathbf{y}_T\right)$, which may differ from $\mathbf{x}_1^T$. In DSR, transmission errors are the reason for this difference. The speech recognition problem thus amounts to solving

$$\hat{\mathbf{W}} = \arg\max_{\mathbf{W}} \left\{ p\left(\mathbf{y}_1^T \mid \mathbf{W}\right) \cdot P\left(\mathbf{W}\right) \right\}. \tag{5}$$

In solving this we need to find an efficient way to compute $p\left(\mathbf{y}_1^T \mid s_1^T\right)$. To this end we introduce the (hidden) uncorrupted feature sequence:

$$p\left(\mathbf{y}_1^T \mid s_1^T\right) = \int_{\mathbf{x}_1^T} p\left(\mathbf{y}_1^T \mid \mathbf{x}_1^T\right) p\left(\mathbf{x}_1^T \mid s_1^T\right) d\mathbf{x}_1^T \tag{6}$$

Using Eq. 3 and noting that

$$p\left(\mathbf{y}_1^T \mid \mathbf{x}_1^T\right) = \prod_{t=1}^{T} p\left(\mathbf{y}_t \mid \mathbf{x}_t\right) \tag{7}$$

we obtain

$$p\left(\mathbf{y}_1^T \mid s_1^T\right) = \int_{\mathbf{x}_1^T} \prod_{t=1}^{T} p\left(\mathbf{y}_t \mid \mathbf{x}_t\right) p\left(\mathbf{x}_t \mid s_t\right) d\mathbf{x}_1^T = \prod_{t=1}^{T} \int_{\mathbf{x}_t} p\left(\mathbf{y}_t \mid \mathbf{x}_t\right) p\left(\mathbf{x}_t \mid s_t\right) d\mathbf{x}_t \tag{8}$$

i.e. it is possible to interchange the product and the integral since the terms inside the integral only depend on *t*.

Often it is more convenient to express $p\left(\mathbf{y}_t \mid \mathbf{x}_t\right)$ via a posterior probability

$$p\left(\mathbf{y}_t \mid \mathbf{x}_t\right) = \frac{p\left(\mathbf{x}_t \mid \mathbf{y}_t\right) p\left(\mathbf{y}_t\right)}{p\left(\mathbf{x}_t\right)} \tag{9}$$

If inter-frame correlation among the feature vector sequence is to be taken into account, $p\left(\mathbf{x}_t \mid \mathbf{y}_t\right)$ has to be replaced by $p\left(\mathbf{x}_t \mid \mathbf{y}_1^T\right)$, i.e. the *a posteriori density* of the clean feature sequence, given all observed corrupted features. This posterior is, from an estimation theory point of view, the complete solution to the problem of estimating the clean feature vector, given all observations. In section 9.3 we will show how this posterior can be efficiently estimated in a distributed speech recognition scenario.

Since we are eventually only interested in the word (state) sequence which maximizes Eq. 8, the probability of the noisy features $p\left(\mathbf{y}_t\right)$ can be disregarded. Further, replacing $p\left(\mathbf{x}_t \mid \mathbf{y}_t\right)$ by $p\left(\mathbf{x}_t \mid \mathbf{y}_1^T\right)$ in Eq. 9 and using it in Eq. 8 we arrive at

$$p\left(\mathbf{y}_1^T \mid s_1^T\right) = \prod_{t=1}^{T} \int_{\mathbf{x}_t} \frac{p\left(\mathbf{x}_t \mid \mathbf{y}_1^T\right)}{p\left(\mathbf{x}_t\right)} p\left(\mathbf{x}_t \mid s_t\right) d\mathbf{x}_t \ . \tag{10}$$

Replacing $p\left(\mathbf{x}_1^T \mid s_1^T\right)$ by $p\left(\mathbf{y}_1^T \mid s_1^T\right)$ in Eq. 2 and using Eq. 10 we finally arrive at

$$p\left(\mathbf{y}_1^T \mid \mathbf{W}\right) = \sum_{s_1^T} \prod_{t=1}^{T} \int_{\mathbf{x}_t} \frac{p\left(\mathbf{x}_t \mid \mathbf{y}_1^T\right)}{p\left(\mathbf{x}_t\right)} p\left(\mathbf{x}_t \mid s_t\right) d\mathbf{x}_t \cdot P\left(s_1^T\right) . \tag{11}$$

The only difference to the standard ASR decoder is that the observation probability $p\left(\mathbf{x}_t \mid s_t\right)$ has to be replaced by the modified observation probability:

$$p\left(\mathbf{x}_t \mid s_t\right) \rightarrow \int_{\mathbf{x}_t} \frac{p\left(\mathbf{x}_t \mid \mathbf{y}_1^T\right)}{p\left(\mathbf{x}_t\right)} p\left(\mathbf{x}_t \mid s_t\right) d\mathbf{x}_t \ . \tag{12}$$

It is instructive to consider the extreme cases of an error-free transmission and a completely unreliable transmission. In case of an error-free transmission there is $\mathbf{y}_t = \mathbf{x}_t$, and the a posteriori density $p\left(\mathbf{x}_t \mid \mathbf{y}_1^T\right)$ reduces to a Dirac delta-impulse. As a result, the modified observation probability, Eq. 12, reduces to the standard observation probability (the denominator is then a constant and can be neglected as it does not influence the maximization in Eq. 5).

In the other extreme case the channel does not transmit any information, which can be expressed by $p\left(\mathbf{x}_t \mid \mathbf{y}_1^T\right) = p\left(\mathbf{x}_t\right)$ for all $t = 1,...,T$. In this case the modified

observation probability evaluates to one and Eq. 5 reduces to $\hat{\mathbf{W}} = \arg\max_{\mathbf{W}} \{p(\mathbf{W})\}$. As the observed features are uninformative, the recognizer has to rely solely on the prior word probabilities.

The key element of the novel decoding rule is the posterior density $p(\mathbf{x}_t \mid \mathbf{y}_1^T)$. The processing of the corrupted features in front of the recognizer has to produce a posterior density instead of a point estimate. It is well-known from estimation theory, that the posterior density comprises all information about the parameter to be estimated, here $\mathbf{x}_t$, that is available from the observations, here $\mathbf{y}_1^T$. Optimal point estimates, such as MMSE or *maximum a posteriori* (MAP) can be obtained as the mean or mode of this density. Further, the (co)variance of the posterior is a measure of reliability of the point estimate. For this reason the posterior has sometimes been called *soft feature* (Haeb-Umbach et al. 2004).

Related decoding rules can be found e.g. in (Morris et al. 1998; Morris et al. 2001; Arrowood et al. 2002; Droppo et al. 2002; Kristjansson et al. 2002; Liao et al. 2004). However, in most cases past and future observed feature vectors are not taken into account for the estimation of the posterior density of the current uncorrupted feature vector, i.e. $p(\mathbf{x}_t \mid \mathbf{y}_1^T)$ is replaced by $p(\mathbf{x}_t \mid \mathbf{y}_t)$. In doing so inter-frame correlation is neglected for the posterior estimation. In section 9.3, however, we will show that inter-frame correlation is a powerful knowledge source to be utilized for transmission error-robust speech recognition.

9.2.2 Gaussian Approximation

Still, the modified observation probability given in Eq. 12 looks intimidating. The computation of the observation probabilities is the single most time consuming processing step in speech recognition. Replacing the evaluation of a mixture density by the numerical evaluation of an integral may increase the computational burden beyond the limits of practical interest. Fortunately, the integral can be solved analytically, if we make the following assumptions:

1. The observation probability is a Gaussian mixture density:

$$p(\mathbf{x}_t \mid s_t) = \sum_{m=1}^{M} c_{s,m} N(\mathbf{x}_t; \boldsymbol{\mu}_{s,m}, \boldsymbol{\Sigma}_{s,m}) \tag{13}$$

2. The *a priori density* of the uncorrupted feature vector can be modeled by a Gaussian density:

$$p(\mathbf{x}_t) = N(\mathbf{x}_t; \boldsymbol{\mu}_{\mathbf{x}}, \boldsymbol{\Sigma}_{\mathbf{x}}) \tag{14}$$

3. The a posteriori density of the uncorrupted feature vector, given the sequence of received feature vectors, can be approximated by a Gaussian density:

$$p(\mathbf{x}_t \mid \mathbf{y}_1^T) \approx p_N(\mathbf{x}_t \mid \mathbf{y}_1^T) = N(\mathbf{x}_t; \boldsymbol{\mu}_{\mathbf{x}_t|\mathbf{y}}, \boldsymbol{\Sigma}_{\mathbf{x}_t|\mathbf{y}}) \tag{15}$$

Further we assume that all Gaussians, Eqs. 13 - 15, have diagonal covariance matrices. Since the individual elements of a diagonal-covariance Gaussian are independent, the densities can then be factorized over the feature vector elements. Let $\mu_{s,m}$, μ_x and $\mu_{x_t|\mathbf{y}}$ denote the means and $\sigma^2_{x,m}$, σ^2_x and $\sigma^2_{x_t|\mathbf{y}}$ the corresponding variances of the Gaussians of an individual vector component of the observation, prior and posterior density, respectively. Then the integral present in Eq. 12 can be solved analytically (Droppo et al. 2002; Ion and Haeb-Umbach 2006c), where for each dimension we obtain the following:

$$\int \sum_{m=1}^{M} c_{s,m} N\left(x_t; \mu_{s,m}, \sigma^2_{s,m}\right) \frac{N\left(x_t; \mu_{x_t|\mathbf{y}}, \sigma^2_{x_t|\mathbf{y}}\right)}{N\left(x_t; \mu_x, \sigma^2_x\right)} dx_t = \sum_{m=1}^{M} c_{s,m} A N\left(\mu_e; \mu_{s,m}, \sigma^2_{s,m} + \sigma^2_e\right) \tag{16}$$

where

$$\begin{aligned} \frac{\mu_e}{\sigma^2_e} &= \frac{\mu_{x_t|y}}{\sigma^2_{x_t|\mathbf{y}}} - \frac{\mu_x}{\sigma^2_x} \\ \frac{1}{\sigma^2_e} &= \frac{1}{\sigma^2_{x_t|\mathbf{y}}} - \frac{1}{\sigma^2_x} \\ A &= \frac{N\left(0; \mu_{x_t|\mathbf{y}}, \sigma^2_{x_t|\mathbf{y}}\right)}{N\left(0; \mu_x, \sigma^2_x\right) N\left(0; \mu_e, \sigma^2_e\right)} \end{aligned} \tag{17}$$

if $\sigma^2_x > \sigma^2_{x_t|\mathbf{y}}$. Eq. 16 states that the variance of the original observation probability of the uncorrupted features is to be increased by σ^2_e and that it is to be evaluated at μ_e and weighted by A.

The assumption of Eq. 13 is the standard model for observation probabilities. Further, the prior density of the feature vector $p(\mathbf{x}_t)$ can be reasonably well approximated by a Gaussian density. The most critical assumption seems to be Eq. 15. We often observed a multi-modal shape of the posterior density. However, the Gaussian approximation was adopted due to computational complexity reasons.

9.3 Feature Posterior Estimation in a DSR Framework

The decoding rule derived in the last section requires knowledge of $p\left(\mathbf{x}_t \mid \mathbf{y}_1^T\right)$, the a posteriori density of the transmitted feature vector, given all received feature vectors. In this section we show how this term can be estimated in the case of distributed speech recognition, where coded MFCCs are transmitted over an error-prone channel. We first describe the ETSI DSR standard to the extent necessary for understanding the subsequent derivation. Subsection 9.3.2 quantifies the redundancy present in the output bit stream of the source coder. The two channel models under consideration are explained in subsection 9.3.3, and 9.3.4 shows how the feature posterior density can be computed from a priori and "transmission probabilities".

This section is concluded by relating other approaches for error concealment to the one presented here.

9.3.1 ETSI DSR Standards

The ETSI distributed speech recognition standards define two feature extraction schemes, standard front end and advanced front end processing, together with the source coding, packet construction, and the backend source decoding scheme (ETSI 2002; ETSI 2003a). For the purpose of error concealment we need to consider the source coder in more detail.

A *source coder* is a mapping of the *N*-dimensional Euclidian space into a finite index set *J* of 2^M elements. It consists of two components: the *quantizer* and the *index generator*. The quantizer maps the *N*-dimensional parameter vector $\mathbf{x}$ to a *N*-dimensional codeword (*centroid*) $\mathbf{c}$ in the finite codebook *C*. This codeword represents all vectors falling in this quantization cell. The index generator then maps this codeword $\mathbf{c}$ to an index (bit pattern) $\mathbf{b}$ in an index set *J*.

The source coder of the ETSI DSR standard employs a *split vector quantizer* (VQ) for the quantization of the static MFCC parameters. The input to the quantizer is the $N = 14$ dimensional parameter vector, consisting of the thirteen-dimensional MFCC feature vector and as a fourteenth component the logarithmic frame energy ($\log E$). The parameter vector is split into seven *subvectors*, each of dimension two, which are quantized with bit-rates (6,6,6,6,6,5,8) bits, respectively. Including one bit for voice-activity information this sums to 44 bits per frame. Before transmission two quantized frames are grouped together creating a frame pair. A 4-bit cyclic redundancy check (CRC) is calculated for each frame pair, resulting in a total of 92 bits per frame pair.

In our notation we will not distinguish between individual subvectors in the following, since the same operations are performed for all subvectors. We even do not make a distinction between the complete vector and any of the subvectors in our notation. Which interpretation is used should become clear from the context.

9.3.2 Source Coder Redundancy

The key to error concealment is the exploit the residual redundancy present in the source coder output bit stream. Let $\mathbf{x}_t$ denote any of the real-valued MFCC subvectors at time frame *t* produced by the front-end. The source coder quantizes the subvector to a codeword $\mathbf{c}_t$ and maps the codeword to a bit pattern $\mathbf{b}_t = (b_t(0),...b_t(M-1)$ of *M* bits, which is transmitted over an equivalent discrete-time channel.

Table 9.1 gives the entropies $\mathrm{H}(\mathbf{b}_t)$ and mutual information $I(\mathbf{b}_t;\mathbf{b}_{t-1})$ of the individual subvectors. The values have been obtained on the training set of the Aurora 2 database (Hirsch and Pearce 2000) using the ETSI advanced feature extraction front-end. Here, subvector 1 denotes the bit pattern corresponding to the first and second mel-frequency cepstral coefficient, subvector 2 the third and fourth, and so

on. Subvector 7 comprises the zero-th cepstral coefficient and $\log E$. M is the number of bits used to code a subvector, i.e. the length of the bit pattern $\mathbf{b}_t$. Comparing M with the entropy $\mathrm{H}(\mathbf{b}_t)$ of the bit pattern, one can observe that for all subvectors the two values are fairly close to each other. This indicates that the bit pattern has almost a uniform distribution. Not much redundancy is left within a subvector which could be utilized for error concealment.

Table 9.1. Entropies and mutual information among the subvectors produced by the ETSI advanced DSR front-end (measured on Aurora 2 training database).

Subvector	1	2	3	4	5	6	7
M	6	6	6	6	6	5	8
$\mathrm{H}(\mathbf{b}_t)$	5.8	5.8	5.8	5.8	5.8	4.8	7.7
$I(\mathbf{b}_t;\mathbf{b}_{t-1})$	2.6	2.1	1.6	1.4	1.2	1.0	3.4
$I(\mathbf{b}_t;(\mathbf{b}_{t-1},\Delta\mathbf{b}_{t-1},\Delta^2\mathbf{b}_{t-1}))$	3.0	2.4	1.9	1.7	1.5	1.3	4.5

The mutual information $I(\mathbf{b}_t;\mathbf{b}_{t-1})$ indicates how much information about the current bit pattern $\mathbf{b}_t$ is already present in the previous $\mathbf{b}_{t-1}$. The larger the mutual information the better a bit pattern following in time can be predicted from the one of the previous frame. Obviously, strong inter-frame correlation exists.

The last line of the table gives the mutual information between the current bit pattern and the bit pattern $(\mathbf{b}_{t-1},\Delta\mathbf{b}_{t-1},\Delta^2\mathbf{b}_{t-1})$ of the previous frame, which consists of the coded static MFCC components $\mathbf{b}_{t-1}$ and the coded dynamic features. For this experiment a $D_1 = 3$ bit vector quantizer was used for the delta (velocity) and just a $D_2 = 1$ bit quantizer for the delta-delta (acceleration) parameter. Obviously, the dynamic parameters of the previous frame provide additional knowledge about the static parameters of the current frame, since the measured mutual information is larger than the one observed between $\mathbf{b}_t$ and $\mathbf{b}_{t-1}$. This comes to no surprise, as the dynamic features capture the trend present in the feature trajectory.

Obviously, the key to successful error concealment is the exploitation of the strong inter-frame correlation of MFCC feature vectors. In specifying the inter-frame correlation models of different complexity may be chosen. A good compromise between modeling accuracy and complexity is to assume that the source vector sequence $\mathbf{b}_t$, $t = 1,2,...$ is a homogeneous first-order Markov process, whose "transition probabilities" $P(\mathbf{b}_t^{(i)} \mid \mathbf{b}_t^{(j)})$, $i, j = 1,...,2^M$ are independent of time. With this model, however, long-term dependencies e.g. on the phone level cannot be captured.

9.3.3 Channel Models

Let us now consider the transmission model of Figure 9.1. At the channel output a bit pattern $\mathbf{y}_t = (y_t(0),...y_t(M-1)$ is observed. Due to transmission errors $\mathbf{y}_t$ and $\mathbf{b}_t$

are not identical. Please note that $\mathbf{y}_t$ is a discrete random variable here, while we assumed $\mathbf{y}_t$ to be a continuous random variable in section 9.2. We prefer this abuse of notation to more easily link the DSR case considered in this section to the more general theory presented in section 9.2.

In the following we use a superscript if we want to denote a specific bit pattern, i.e. $\mathbf{b}_t^{(i)}$ indicates the bit pattern corresponding to the *i*-th codebook centroid $\mathbf{c}_t^{(i)}$, $i \in \{1,...,2^M\}$.

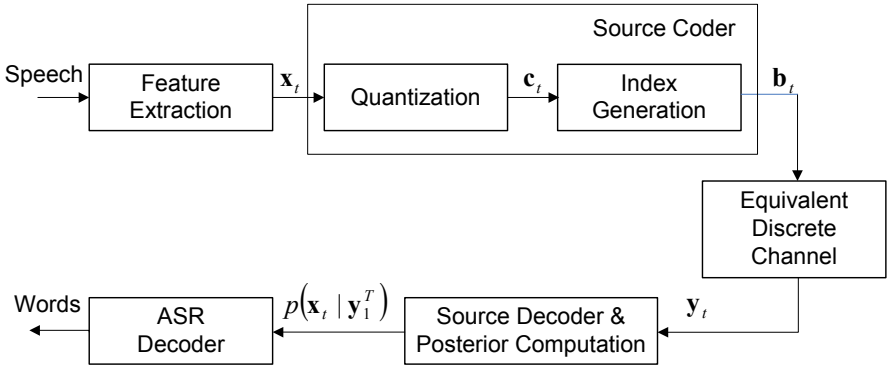

Fig. 9.1. Block diagram of distributed speech recognition system

We consider two channel models:

a) Time-variant binary symmetric channel (TV-BSC)

The TV-BSC is an equivalent discrete channel which models the effects of additive white Gaussian noise on the transmitted bit sequence. While one usually assumes constant bit error probability in a BSC, we want to allow here the bit error probability p_t to be time-variant. This model can be used to characterize wireless circuit-switched transmission, where the bit error rate varies, e.g. due to time-variant multi-path fading.

As the channel is assumed to be memoryless, the probability of a received bit pattern given the sent can be expressed as

$$P\left(\mathbf{y}_t \mid \mathbf{b}_t^{(i)}\right) = \prod_{m=0}^{M-1} P\left(y_t(m) \mid b_t^{(i)}(m)\right) \tag{18}$$

where

$$P\left(y_t(m) \mid b_t^{(i)}(m)\right) = \begin{cases} 1 - p_t(m) & \text{if} \quad y_t(m) = b_t^{(i)}(m) \\ p_t(m) & \quad\quad y_t(m) \neq b_t^{(i)}(m) \end{cases} \tag{19}$$

Here, $p_t(m)$ is the (instantaneous) bit error probability of the *m*-th bit of the *t*-th bit pattern. This probability can either be obtained from a soft-output channel de-

coder or can be estimated from consistency checks applied to the received bits (Ion and Haeb-Umbach 2006a).

b) Packet erasure channel

In this channel model, a data packet is either completely lost or received without any bit error. It models the random loss of data packets, e.g. due to network congestion. Most real communication channels exhibit packet losses occurring in bursts. Such channels can be modeled by a 2-state Markov chain, known as *Gilbert model*, see Fig. 9.2. In the figure p is the probability that the next packet is lost, provided the previous one has arrived; q is the probability that the next packet is not lost, given that the previous one was lost. The parameter q can be seen as controlling the burstiness of packet losses. This channel model is often described in terms of the *mean loss probability* $mlp = p/(p+q)$, the average probability of loosing a packet, and *conditional loss probability* $clp = 1-q$, i.e. the probability of loosing a packet, conditioned on the event that the previous packet was lost.

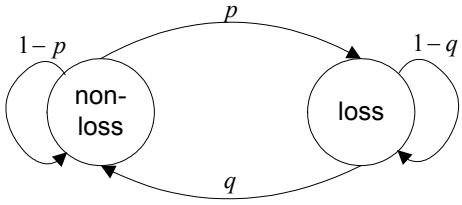

Fig. 9.2. Gilbert model

It is important to model the bursty nature of packet losses. It was shown that the word error rate of a DSR system depends strongly on the burstiness of the channel: Frame losses of up to 50% hardly have an effect on the word error rate, provided the average burst length is one packet (i.e. one frame pair), while the word error rate dramatically increases for longer average burst lengths (Gómez, Peinado, Sánchez and Rubio, 2007).

For a packet erasure channel model the probability of the received bit pattern, given the sent, is as follows:

$$P\left(\mathbf{y}_t \mid \mathbf{b}_t^{(i)}\right) = \begin{cases} \delta\left(\mathbf{y}_t - \mathbf{b}_t^{(i)}\right) & \text{packet received} \\ \dfrac{1}{2^M} & \text{packet lost} \end{cases} \quad if \tag{20}$$

Here $\delta(\cdot)$ denotes the Kronecker delta impulse.

Note that in practice often a combination of both error types is present. Communications that involve both a wireless and a packet-based wireline link may exhibit both packet losses and bit errors. Packets with bit errors are discarded by the User Datagram Protocol (UDP). While this is reasonable for many payloads, for DSR or speech transmission it would make more sense to deliver packets with bit errors, as it allows for more effective error concealment. UDP-Lite (RFC 3828, 2004) is a transport protocol that allows the application to receive partially corrupted packets.

9.3.4 Estimation of Feature Posterior

At the receiving end we are given the sequence $\mathbf{y}_1^T = \mathbf{y}_1,...\mathbf{y}_T$, and our goal is to carry out speech recognition by employing the modified observation probability given by Eq. 12.

To this end we need to compute the a posteriori probability density $p\left(\mathbf{x}_t \mid \mathbf{y}_1^T\right)$. Figure 9.3 illustrates the different processing steps. Note, that the input to the ASR decoder is no longer a feature vector, but a probability density function.

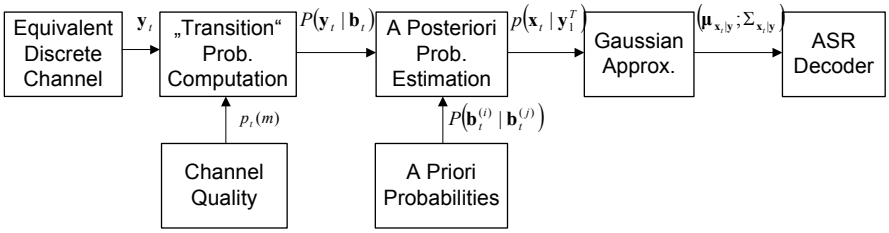

Fig. 9.3. Block diagram of posterior estimation and uncertainty decoding

Introducing the hidden (unobservable) sent bit pattern, we can express the posterior density as follows:

$$p\left(\mathbf{x}_t \mid \mathbf{y}_1^T\right) = \sum_{i=1}^{2^M} p\left(\mathbf{x}_t \mid \mathbf{b}_t^{(i)}\right) P\left(\mathbf{b}_t^{(i)} \mid \mathbf{y}_1^T\right) \tag{21}$$

The computation of the posterior probability $P\left(\mathbf{b}_t^{(i)} \mid \mathbf{y}_1^T\right)$ can be accomplished using the *Forward-Backward (FB) algorithm* (Bahl, Cocke, Jelinek and Raviv 1974; Peinado et al. 2003):

$$P\left(\mathbf{b}_t^{(i)} \mid \mathbf{y}_1^T\right) = \frac{\alpha_t^{(i)} \beta_t^{(i)}}{\sum_{j=0}^{2^M-1} \alpha_t^{(j)} \beta_t^{(j)}} \tag{22}$$

where

$$\begin{aligned} \alpha_t^{(i)} &= P\left(\mathbf{b}_t^{(i)}, \mathbf{y}_1, \mathbf{y}_2, ...\mathbf{y}_t\right) \\ \beta_t^{(i)} &= P\left(\mathbf{y}_{t+1}, \mathbf{y}_{t+2}, ...\mathbf{y}_T \mid \mathbf{b}_t^{(i)}\right) \end{aligned} \tag{23}$$

Both $\alpha_t^{(i)}$ and $\beta_t^{(i)}$ are computed recursively.

Using the FB algorithm the a posteriori density can be computed for either of the two channel models outlined in section 9.3.3 and either of the two source models considered in section 9.3.2. In the case of a packet erasure channel a very efficient realization of the recursions can be found exploiting the property of Eq. 20 (Ion and Haeb-Umbach 2006b).

Although the dynamic vector components are not transmitted, error concealment can benefit from the superior prediction quality of a source model including static and dynamic vector components. For the source model which models the sequence of bit patterns corresponding to the static MFCC vectors only as a first-order Markov model, there are 2^M bit patterns $\mathbf{b}^{(i)}$, $i \in \{1,...,2^M\}$ per subvector, and the inter-frame correlation is captured by a $2^M \times 2^M$ matrix, whose $(i,j)^{\text{th}}$ element is $P(\mathbf{b}_t^{(i)} \mid \mathbf{b}_t^{(j)})$. On the other hand, for the source model which considers a feature vector including dynamic components, inter-frame correlation is captured by a $2^{M+D_1+D_2} \times 2^{M+D_1+D_2}$ matrix, where M, D_1, and D_2 are the number of bits used to code the subvector of static, first-order and second-order differential coefficients. The matrices are estimated beforehand on clean training data. Since only the bits corresponding to the static MFCC vector are actually transmitted, the "transition probability" is independent of the bits corresponding to the dynamic part of the feature vector: $P(\mathbf{y}_t \mid \mathbf{b}_t, \Delta\mathbf{b}_t, \Delta^2\mathbf{b}_t) = P(\mathbf{y}_t \mid \mathbf{b}_t)$. In section 9.4 we compare the two source models w.r.t. speech recognition accuracy obtained on an error-prone channel. To simplify notation we will assume the source model of static components only in the remainder of this section.

Note that the FB algorithm needs to be performed only inside isolated erroneous regions (error bursts), i.e. when $P(\mathbf{y}_t \mid \mathbf{b}_t^{(i)})$ is not a Delta impulse. Then the FB recursions are initialized using the last uncorrupted feature vector before and the first uncorrupted feature vector after the error burst. Detecting the presence of an uncorrupted feature vector is trivial in the case of a packet erase channel, but it is not that trivial in the case of a time-variant BSC. In the latter case erroneous bit patterns can be detected based on consistency checks among subsequent bit patterns and on cyclic redundancy check failure (Ion et al. 2006a).

The other term needed in Eq. 21, $p(\mathbf{x}_t \mid \mathbf{b}_t^{(i)})$, is the probability density function (pdf) of the feature vector, given the *i*-th centroid. This VQ cell-conditioned pdf is modeled as a Gaussian $p(\mathbf{x}_t \mid \mathbf{b}_t^{(i)}) = N(\mathbf{x}_t; \mathbf{c}_t^{(i)}, \Sigma_t^{(i)})$, where $\mathbf{c}_t^{(i)}$ is the VQ centroid corresponding to $\mathbf{b}_t^{(i)}$. The within-cell covariance matrix $\Sigma_t^{(i)}$ can be estimated on the training data.

In order to simplify subsequent processing, the feature posterior, Eq. 21, is approximated a Gaussian density $p_N(\mathbf{x}_t \mid \mathbf{y}_1^T) = N(\mathbf{x}_t; \boldsymbol{\mu}_{\mathbf{x}_t|\mathbf{y}}, \Sigma_{\mathbf{x}_t|\mathbf{y}})$, see Eq. 15. The parameters $\boldsymbol{\mu}_{\mathbf{x}_t|\mathbf{y}}, \Sigma_{\mathbf{x}_t|\mathbf{y}}$ of this Gaussian can be obtained by finding that Gaussian which has the smallest *Kullback-Leibler divergence* to the original non-Gaussian posterior $p(\mathbf{x}_t \mid \mathbf{y}_1^T)$. This results in the following estimates:

$$\mu_{\mathbf{x}_t|\mathbf{y}} = \sum_{i=1}^{2^M} P(\mathbf{b}_t^{(i)} \mid \mathbf{y}_1^T) \mathbf{c}_t^{(i)} \tag{24}$$

$$\Sigma_{\mathbf{x}_t|\mathbf{y}} = \sum_{i=1}^{2^M} P\left(\mathbf{b}_t^{(i)} \mid \mathbf{y}_1^T\right)\left(\left(\mathbf{c}_t^{(i)} - \boldsymbol{\mu}_{\mathbf{x}_t|\mathbf{y}}\right)\left(\mathbf{c}_t^{(i)} - \boldsymbol{\mu}_{\mathbf{x}_t|\mathbf{y}}\right)^T + \Sigma_t^{(i)}\right) \tag{25}$$

This result makes intuitively sense: The mean $\boldsymbol{\mu}_{\mathbf{x}_t|\mathbf{y}}$ of the Gaussian is equal to the mean of the original posterior, and the covariance is the sum of the between-VQ-cell covariance and the within-VQ-cell covariance. For high resolution, i.e. sufficiently large *M*, as is e.g. the case for the vector quantizer used in the ETSI DSR standard, the within-cell variance is negligibly small, such that Eq. 25 simplifies to

$$\Sigma_{\mathbf{x}_t|\mathbf{y}} \approx \sum_{i=0}^{2^M-1} P\left(\mathbf{b}_t^{(i)} \mid \mathbf{y}_1^T\right)\left(\mathbf{c}_t^{(i)} - \boldsymbol{\mu}_{\mathbf{x}_t|\mathbf{y}}\right)\left(\mathbf{c}_t^{(i)} - \boldsymbol{\mu}_{\mathbf{x}_t|\mathbf{y}}\right)^T . \tag{26}$$

The posterior probability is the complete solution to the problem of estimating the uncorrupted features from the corrupted ones. The mean of the posterior given in (24) is the MMSE estimate of the feature vector $\mathbf{x}_t$. If one were only interested in the reconstruction of the uncorrupted feature vector, one could, for example, use this estimate. The maximum of the posterior is the maximum a posteriori estimate of the feature vector, another estimate commonly used in various estimation problems. The covariance matrix of the posterior, Eq. 25, is a measure of reliability of the reconstructed features. If the parameter to be estimated and the observation are jointly Gaussian, it equals the covariance matrix of the MMSE estimation error.

9.3.5 Related Work

Several server based error mitigation schemes proposed for distributed speech recognition can be related to the framework presented in this article.

Peinado et al. 2003 employ the MMSE estimate, Eq. 24, to reconstruct corrupted feature vectors on a channel exhibiting bit errors. A crucial issue, however, is the determination of the instantaneous bit error probability $p_t(m)$ needed in Eq. 19. It may either be obtained from the soft-output of the channel and SNR estimation (Peinado et al. 2003) or a soft-output channel decoder (Haeb-Umbach et al. 2004). If the soft-output is not available the bit error probability can be estimated from consistency checks applied to the received bit patterns (Ion and Haeb-Umbach 2006a).

Marginalization reformulates the classification to perform recognition based on the reliable features alone (Endo et al. 2003). On a packet erasure channel there is a straightforward association of packet loss with unreliable data. However on a channel characterized by bit errors it is difficult to decide whether a feature is reliable or not, even if the instantaneous bit error probability of all bits making up the representation of the feature is available. In (Endo et al. 2003) a threshold was experimentally determined. If the bit error probability was larger than the threshold the corresponding feature was marginalized.

Marginalization can be obtained in the presented framework, if the (simpler) feature posterior $p(\mathbf{x}_t \mid \mathbf{y}_t)$, which is only conditioned on the received data corresponding to the current frame, is used instead of $p(\mathbf{x}_t \mid \mathbf{y}_1^T)$ in the modified observation probability of Eq. 12. If a feature is declared lost, then $p(\mathbf{x}_t \mid \mathbf{y}_t) = p(\mathbf{x}_t)$. Using this in Eq. 12, the integral evaluates to one, i.e. the corresponding frame is marginalized.

The binary reliability measure used in marginalization can be replaced by a continuous confidence measure γ, taking values between zero and one. *Weighted Viterbi* (WV) decoding takes into account the confidence about a feature vector by raising the observation probability to the power of γ (Bernard and Alwan 2001). Obviously, for the correctly received feature vectors there is $\gamma = 1$, and no changes to the observation probability occur. For a lost feature vector the maximum uncertainty is expressed by $\gamma = 0$, resulting in an observation probability evaluating to one and being independent of the state. Thus with binary weighting WV is equivalent to marginalization. However, raising the observation probability to some power γ anywhere between zero and one lacks a probabilistic interpretation. Moreover, determining an optimal value for γ is not an easy task. The methods proposed to determine the confidence measure γ are rather empirical, and the optimal value depends on the recognition task (Cardenal-López, Garcia-Mateo and Docío-Fernández 2006).

The effect of raising the observation probability to some power between zero and one is to deemphasize the contribution of this frame to the ASR decision. The same effect is achieved with the observation probability of Eq. 12 proposed in this paper, if the feature posterior is not a Dirac delta impulse.

9.4 Performance Evaluations

In this section we present experimental results for distributed speech recognition employing the proposed error concealment techniques. We first describe the experimental setup and then give speech recognition results for the two channel models outlined in section 9.3.3 and for two recognition tasks, a small vocabulary and a medium vocabulary task.

9.4.1 Experimental Setup

We consider a setup which is compatible to the ETSI standards for DSR. The whole front-end processing, consisting of feature extraction, source coding and packetization is carried out according to the ETSI advanced front-end (ETSI 2002) standard.

As an example for a channel exhibiting bit errors the GSM data channel was considered. A realistic simulation of the GSM physical layer processing was carried out including channel coding/decoding, interleaving/deinterleaving, modulation/demodulation. The channel coding was TCH/F4.8 described in (ETSI 2003b) which uses convolutional coding at a rate $r = 1/3$. The channel decoding employed

the FB algorithm (Bahl et al. 1974) which is able to provide the instantaneous bit error probability $p_t(m)$. We preferred this full channel simulation, since if we had used merely GSM error patterns, the instantaneous bit error rate would not have been available.

We have chosen a channel model approximating a "typical urban" profile specified by COST 207 (COST 1989). The model is characterized by 12 propagation paths, delay spread of 1.03 μs and Rayleigh fading. The terminal was assumed to be moving at 50 km/h. Various Carrier-to-Interference (C/I) power ratios were simulated, ranging from 10 dB to 2.5 dB. Note that C/I=2.5 dB is a very poor channel, where the bit error rate is as high as 3.6%.

For the packet erasure channel we adopted the Gilbert model to model that packet losses occur in bursts. In the literature often four channel conditions are evaluated, with C1 corresponding to mildly bad and C4 to very poor channel conditions. Table 9.2 gives the conditional and mean loss probabilities of the four conditions (Boulis et al. 2002). In our simulations we transmitted one frame pair per packet.

Table 9.2. Packet erasure channel test conditions

Condition	C1	C2	C3	C4
clp	0.147	0.33	0.50	0.60
mlp	0.006	0.09	0.286	0.385

Different error concealment techniques were applied at the receiving (server) side and compared in terms of achieved word error rate obtained on two databases.

The small vocabulary task is the clean test set of the Aurora 2 database, which consists of 4004 utterances from 52 male and 52 female speakers distributed over four subsets. The sampling rate is 8 kHz. The acoustic models used in the recognizer were those described in (Hirsch et al. 2000): 16 states per word, 3 Gaussians per state.

The medium vocabulary task is the Wall Street Journal WSJ0 5k Nov. '92 evaluation test set (Paul and Baker 1992) comprising 330 utterances of 4 male and 4 female speakers, summing up to 40 min of speech. Here, the sampling rate is 16 kHz. Recognition experiments were carried out on this test set using a closed vocabulary bigram language model. The acoustic model consisted of 3437 tied states. The parameters of the 10-component mixture densities were trained on the SI-84 set of the WSJ corpus using the HTK toolkit (Young et al. 2004).

9.4.2 Results on GSM Data Channel

Figure 9.4 gives an illustrative example of the reconstruction achieved by employing the a posteriori density. The figure shows how the feature $\log E$ is reconstructed in the presence of bit errors during transmission. The continuous solid line labeled x_t is the sent ("true") value of the parameter over the frame index *t*. $\mu_{x_t|\mathbf{y}}$ is the MMSE estimate, and $\mu_{x_t|\mathbf{y}} \pm \sigma_{x_t|\mathbf{y}}$ the MMSE estimate plus/minus one standard deviation of the a posteriori density. The interval given in this way can be interpreted as confi-

dence interval for the MMSE estimate. The curve NFR shows the reconstruction by *nearest frame repetition*, which is the error concealment strategy proposed in the ETSI standard. The grey areas show intervals in which transmission errors occurred. We used two grey scales to distinguish between regions where transmission errors occurred in the bit pattern carrying the $\log E$ component (dark grey) and regions where the bit patterns corresponding to other subvectors of the same frame were affected by errors (light grey). It can be seen that the $\log E$ component is not affected by transmission errors in other subvectors. This can be attributed to the fact that the a posteriori computation operates on a per-subvector basis. Uncorrupted parts are forwarded to the recognizer without modification. A subvector-based error concealment, such as this or the one proposed by (Tan, Dalsgaard and Lindberg 2004) is superior to a frame-based scheme, such as NFR, where a complete frame is modified, even if only one subvector is degraded by transmission errors. But even if the illustrated $\log E$ component is affected by transmission errors, much better feature reconstruction is achieved with the proposed method compared to NFR.

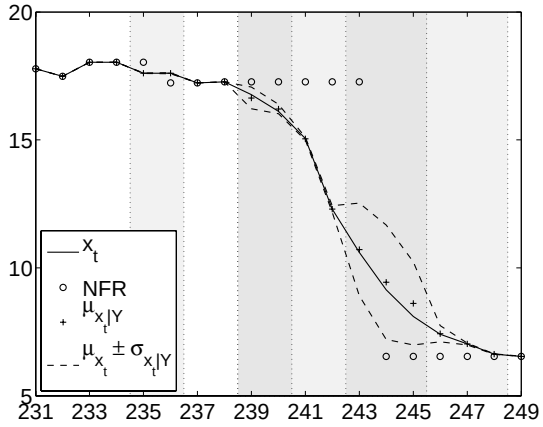

Fig. 9.4. Example of feature reconstruction. The figure shows the trajectory of the $\log E$ feature over time (labeled x_t) and its reconstructions, either by nearest frame repetition (NFR) or by the proposed scheme. The shaded areas indicate regions where bit errors occurred during transmission, either in the $\log E$ component (dark grey) or another component of the feature vector (light grey).

Figure 9.5 and Fig. 9.6 present word error rates for different Carrier-to-Interference (C/I) power ratios for the Aurora 2 and WSJ0 database, respectively. In these figures, the performance of the proposed scheme, termed *uncertainty decoding* (UD), is compared with *marginalization* (M), *nearest frame repetition* (NFR) and *Weighted Viterbi* decoding (WV). For WV, the confidence γ was computed as in (Potamianos and Weerackody 2001), however using the instantaneous bit error probability from the channel decoder. For UD, we employed the source model based on the correlation of static features only. It can be seen, that UD outperforms all other

schemes. Speech recognition accuracy is hardly affected for C/I-values as low as 2.5 dB. Figure 9.5 also shows the bit error rate (BER) at the output of the channel decoder. It is interesting to note that BER increases by almost three orders of magnitude when C/I is reduced from 10 dB to 2.5 dB, while the word error rate achieved by UD is only mildly affected. This underscores that uncertainty decoding makes the ASR decoder very robust towards degraded channel conditions.

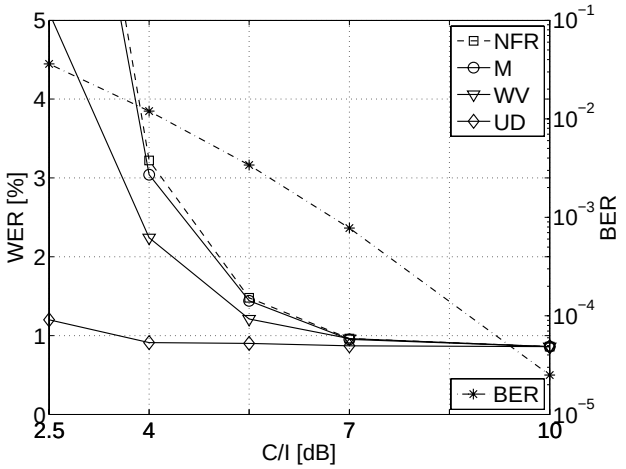

Fig. 9.5. Word error rates for transmission over GSM TCH/F4.8 channel using different error concealment schemes; Aurora 2 task. The dash-dotted line indicates the bit error rate (BER) at the output of the channel decoder.

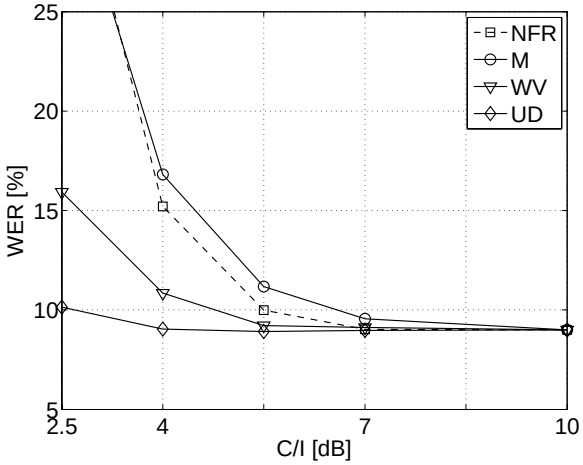

Fig. 9.6. Word error rates for transmission over GSM TCH/F4.8 channel using different error concealment schemes; WSJ0 task.

As the two are closely related, the frame error rate increases similarly to BER, from 0.08% at C/I = 10 dB to 60% at C/I = 2.5 dB. As a consequence marginalization and nearest frame repetition, which operate on a vector rather than a subvector basis, perform poorly.

9.4.3 Results on Packet Erasure Channel

For the experiments on the packet erasure channel we used the channel conditions C1 to C4, specified in Table 9.2. Figure 9.7 and Fig. 9.8 display the word error rates of different error concealment techniques for the Aurora 2 and WSJ0 task, respectively. In the figures we included a condition C0 as a reference, which corresponds to an error-free transmission. Results are presented for two variants of the proposed scheme: *uncertainty decoding* employing an a priori model of the source which captures correlation among the static MFCCs alone (UD) and the one utilizing the correlation among the full (static and dynamic) feature vector (UD-dyn). It can be seen, that UD outperforms *marginalization* (M) and *nearest frame repetition* (NFR). The performance of *Weighted Viterbi* (WV) decoding comes close to UD. For the WV curve the lost features were reconstructed by NFR, and their confidence $\gamma(t)$ was chosen dependant on the relative position (τ) within an error burst. It equals one at the start and end of the burst and decreases exponentially according to $\gamma(t_{start}+\tau)=\gamma(t_{end}+\tau)=\alpha^{\tau}$ towards the middle (Cardenal-López et al. 2006). Here, t_{start} and t_{end} denote the starting and ending time of the error burst. The optimal value of α was experimentally found to be $\alpha=0.7$ for this task.

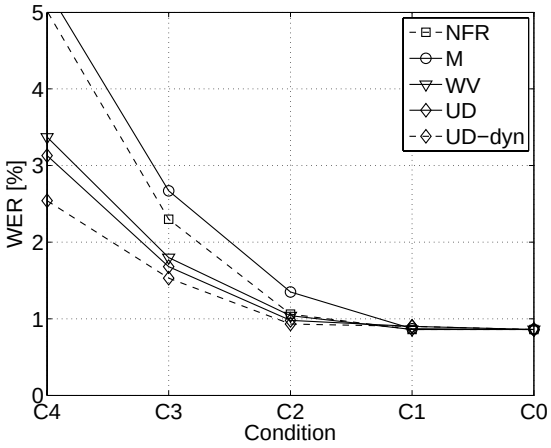

Fig. 9.7. Word error rates for packet erasure channel using different error concealment schemes; Aurora 2 task

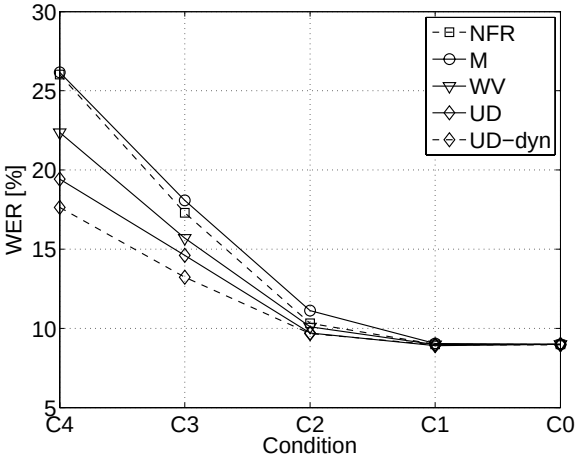

Fig. 9.8. Word error rates for packet erasure channel using different error concealment schemes; WSJ0 task

9.5 Conclusions

Error concealment is concerned with mitigating the detrimental effect that transmission errors may have on the recipient of the signal by exploiting residual redundancy in the bit stream of the source coder output. In distributed speech recognition (DSR) the recipient is the ASR decoder, which, unlike a human listener, can take advantage of both the optimally reconstructed transmitted data and information about the reliability of the reconstruction. The Bayes decision rule therefore has to be reformulated to account for a corrupted or unreliable feature vector sequence. This results, under certain assumptions, in just a modification of the observation probability computation, while the structure of the decoder, which is based on the Viterbi search, remains unchanged. Crucial to the performance of this modified decoding rule is the accuracy of the a posteriori probability density estimate of the uncorrupted feature vector, given all the received corrupted ones. For DSR we were able to find an efficient estimation method, both for channels characterized by bit errors and channels exhibiting packet losses. The key was to exploit the high inter-frame correlation of MFCC feature vectors. Using these techniques high recognition accuracy can be maintained over a wide range of channel conditions.

It should be noted that server-based error concealment techniques, as the ones described in this contribution, are fully compatible with the ETSI standards for distributed speech recognition.

9.6 Acknowledgment

This work was supported by Deutsche Forschungsgemeinschaft under contract numbers HA 3455/2-1 and HA 3455/2-3.

References

Arrowood, J.A. and Clements, M.A. (2002) Using observation uncertainty in HMM decoding. In *Proc. ICSLP*, Denver, Colorado.

Bahl, L., Cocke, J. Jelinek, F. and Raviv, J. (1974) Optimal decoding of linear codes for minimizing symbol error rate, *IEEE Trans. Inf. Theory*, vol. 10, pp. 284-287.

Bernard, A. and Alwan, A. (2001) Joint channel decoding – Viterbi recognition for wireless applications. In *Proc. EUROSPEECH*, Aalborg, Denmark.

Bernard, A. and Alwan, A. (2002) Low-bitrate distributed speech recognition for packet-based and wireless communication. *IEEE Trans. Speech and Audio Process*., vol. 10, no. 8, Nov., 2002.

Boulis, C., Ostendorf, M., Riskin, E.A. and Otterson, S. (2002) Graceful degradation of speech recognition performance over packet-erasure networks. *IEEE Trans. on Speech and Audio Processing*, vol. 10, no. 8, Nov. pp. 580-590.

Cardenal-López, A., García-Mateo , C. and Docío-Fernández , L. (2006) Weighted Viterbi decoding strategies for distributed speech recognition over IP networks, Speech Communication, vol. 48, no. 11, Nov., pp. 1422-1434.

COST 207 (1989) *Digital land mobile radio communication – final report*. Office for official publications of the European Communities, Luxembourg.

Cox, R.V., Kleijn, W.B. and Kroon, P. (1989) Robust CELP coders for noisy backgrounds and noisy channels. In *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process*., 1989, pp. 739-742.

Davis, S.B. and Mermelstein P. (1980), Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. *IEEE Trans. on Acoust., Speech and Signal Process.,* vol. 28, pp. 357-366.

Droppo, J., Acero, A., and Deng, L. (2002) Uncertainty decoding with Splice for noise robust speech recognition. In *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process*., Orlando, Florida.

Endo, T., Kuroiwa, S. and Nakamura, S. (2003) Missing feature theory applied to robust speech recognition over IP networks. In *Proc. EUROSPEECH*, Geneva, Switzerland.

ETSI Standard ES 202 050 (2002) *Speech Processing, Transmission and Quality Aspects (STQ);Distributed speech recognition; Advanced front-end feature extraction algorithm; Compression algorithms.* v1.1.1, Oct.

ETSI Standard ES 201 108 (2003a) *Speech Processing, Transmission and Quality Aspects (STQ);Distributed speech recognition; Front-end feature extraction algorithm; Compression algorithms.* v1.1.3, Sep.

ETSI Standard TS 100 909 v8.7.1 (2003b) *Digital cellular telecommunications system (phase 2+); channel coding.* (3GPP TS 05.03 version 8.7.0; Release 1999).

Fingscheidt, T., Aalburg, S., Stan, T. and Beaugeant,C. (2002) Network-based vs. distributed speech recognition in adaptive multi-rate wireless systems. In *Proc. Int. Conf. on Spoken Language Proc.*, Denver.

Fingscheidt, T. and Vary, P. (2001) Softbit speech decoding: A new approach to error concealment. *IEEE Trans. Speech and Audio Proc.*, vol. 9, no. 3, March, pp. 1-11.

Gómez, A.M., Peinado, A.M., Sánchez, V. and Rubio, J. (2007) On the Ramsey class of interleavers for robust speech recognition in burst-like packet loss, *IEEE Trans. Audio, Speech and Lang. Process*., vol. 15, no. 4, May, pp. 1496-1499.

GSM 06.11 Recommendation (1992) *Substitution and Muting of Lost Frames for Full Rate Speech Traffic Channels*. ETSI TC-SMG.

Haeb-Umbach, R. and Ion, V. (2004) Soft features for improved distributed speech recognition over wireless networks. In *Proc. ICSLP*, Jeju, Korea.

Hirsch, H.G. and Pearce, D. (2000) The Aurora experimental framework for the performance evaluation ofspeech recognition systems undernoisy conditions. In *Proc. ISCA ITRW Workshop ASR2000*, Paris, France, pp. 181-188.

Ion, V. and Haeb-Umbach, R (2005) A unified probabilistic approach to error concealment for distributed speech recognition. In *Proc. Interspeech*, Lisbon.

Ion, V. and Haeb-Umbach, R. (2006a) Uncertainty decoding for distributed speech recognition over error-prone networks, *Speech Communication* 48, pp. 1435-1446.

Ion, V. and Haeb-Umbach, R. (2006b) An inexpensive packet loss compensation scheme for distributed speech recognition based on soft-features, in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Toulouse, France.

Ion, V. and Haeb-Umbach, R. (2006c) Improved source modeling and predictive classification for channel robust speech recognition, in *Proc. Interspeech*, Pittsburgh.

ITU-T Recommendation G.711 Appendix I (1999) *A high quality low-complexity algorithm for packet loss concealment with G.711*.

James, A.B., Gomez, A. and Milner, B.P. (2004) A comparison of packet loss compensation methods and interleaving for speech recognition in burst-like packet loss. In *Proc. ICSLP*, Jeju, Korea.

Kristjansson, T.T. and Frey, B.J. (2002) Accounting for uncertainty in observations: A new paradigm for robust speech recognition. In *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Orlando, Florida.

Lahouti, F. and Khandani, A.K. (2007) Soft reconstruction of speech in the presence of noise and packet loss. *IEEE Trans. Audio, Speech and Language Proc*., vol. 15, no. 1, Jan., pp. 44-56.

Liao, H. and Gales, M.J.F. (2004) *Uncertainty decoding for noise robust automatic speech recognition*. Technical Report TR.499, Cambridge University Engineering Department.

Milner, B. and Semnani, S. (2000) Robust speech recognition over IP networks. In *Proc. Int. Conf. Acoust., Speech, Signal Process.*, Istanbul, Turkey.

Morris, A., Cooke, M. and Green, P. (1998) Some solutions to the missing feature problem in data classification, with application to noise-robust ASR. In *Proc. Int. Conf. Acoust., Speech, Signal Process.*, Seattle.

Morris, A., Barker, J. and Bourlard, H. (2001) From missing data to maybe useful data: Soft data modeling for noise robust ASR. In *Proc. WISP*, vol. 6.

Paul, D. and Baker, J. (1992) *The design for the Wall Street Journal based CSR corpus*. DARPA techn. Report.

Pearce, D. (2000) Enabling new speech driven services for mobile devices: An overview of the ETSI standards activities for distributed speech recognition front-ends, in Proc. *Voice Input/Output Soc. Speech Applications Conference*, May.

Peinado, A.M., Sanchez, V., Perez-Cordoba, J.L. and de la Torre, A. (2003) HMM-based channel error mitigation and its application to distributed speech recognition. *Speech Communication*, 41, pp. 549-561.

Potamianos, A. and Weerackody, V. (2001) Soft-feature decoding for speech recognition over wireless channels, in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Salt Lake City, Utah.

RFC 2460 (1998) *Internet Protocol, Version 6 (IPv6) Specification*, http://www.ietf.org/rfc/rfc2460.txt, Internet Engineering Task Force, Dec.

RFC 3828 (2004) *The Lightweight User Datagram Protocol (UDP-Lite)*, http://www.ietf.org/rfc/rfc3828.txt, Internet Engineering Task Force, July.

Tan, Z.-H., Dalsgaard, P., and Lindberg, B. (2004) A subvector-based error concealment algorithm for speech recognition over mobile networks. In *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Montreal, Quebec, Canada.

Tan, Z.H., Dalsgaard, P. and Lindberg, B. (2005) Automatic speech recognition over error-prone wireless networks, *Speech Communication*, vol. 47, no. 1-2, Sep.-Oct., pp 220-242.

Vary, P. and Martin, R. (2006) *Digital Speech Transmission – Enhancement, Coding and Error Concealment*. John Wiley, New York.

Weerackody, V., Reichl, W. and Potamianos, A. (2002) An error-protected speech recognition system for wireless communications. *IEEE Trans. on Wireless Communications*, vol. 1, no. 2, April, pp. 282-291.

Young, S.J. et al. (2004) *HTK: Hidden Markov Model Toolkit V3.2.1 Reference Manual*. Cambridge University Speech Group, Cambridge, U.K.