TECHNIQUES FOR CAPTURING TEMPORAL VARIATIONS IN SPEECH SIGNALS WITH FIXED-RATE PROCESSING

S. Dharanipragada[1] *R.A. Gopinath*[1] *B.D. Rao*[2]

[1]IBM T.J. Watson Research Center
PO Box 218, Yorktown Heights, NY, 10598, USA
dsatya@watson.ibm.com rameshg@watson.ibm.com
[2]Department of Electrical and Computer Engineering
Univ. of California, La Jolla San Diego, CA 92093-0407
brao@ece.ucsd.edu

ABSTRACT

Fixed-rate feature extraction which is used in most current speech recognizers is equivalent to sampling the feature trajectories at a uniform rate. Often this sampling rate is well below the Nyquist rate and thus leads to distortions in the sampled feature stream due to aliasing. In this paper we explore various techniques, ranging from simple cepstral and spectral smoothing to filtering and data-driven dimensionality expansion using Linear Discriminant Analysis (LDA), to counter aliasing and the variable rate nature of information in speech signals. Smoothing in the spectral domain results in a reduction in the variance of the short term spectral estimates which directly translates to reduction in the variances of the Gaussians in the acoustic models. With these techniques we obtain modest improvements, both in word error rate and robustness to noise, on large vocabulary speech recognition tasks.

1. INTRODUCTION

It is well known that the rate of temporal change in acoustic realizations of phonetic units varies significantly according to the phonetic unit in question. For example, in plosives the temporal variation is much faster and shorter in duration than in vowels. This suggests that a variable frame rate should be used for feature extraction in speech recognition. However, most current state-of-the-art speech recognizers use a constant frame rate of 100 frames/second, which limits the extent of temporal variation in the cepstral features to 50Hz. Experiments indicate that variations in the cepstra sometimes occur at a much faster rate than this frame-rate. Therefore straightforward downsampling causes aliasing distortions in the cepstral features.

The standard method to alleviate aliasing effects is to lowpass filter the trajectories before downsampling. Lowpass filtering or smoothing can be achieved many ways. In this paper we explore various techniques, ranging from simple cepstral and spectral averaging to filtering using a lowpass filter designed using a constrained least-squares optimization method. This approach of filtering, motivated purely from signal processing considerations, is very different from RASTA processing, which is motivated from human auditory perception considerations, since the filtering is done within each frame and not across frames as is done in RASTA [1].

We also propose another approach to handle the variable rate nature of information in the cepstral trajectories with fixed-rate processing. This we achieve through filtering accompanied by dimensionality expansion. This can be achieved either by examining the spectral content in each of the cepstral trajectories and deciding to increase the number of filter-banks for that dimension if a significant amount of energy is outside the original lowpass filter or by using a purely data-driven method such as LDA.

This paper is organized as follows. In the next section we describe the effects of sampling feature trajectories and suggest various methods to reduce these effects. The following section presents results on large vocabulary tasks.

2. CEPSTRAL TRAJECTORIES

Let the short-time power spectrum of a speech signal, $s(n)$, be approximated by a windowed periodogram as follows:

$$S(\omega, m) = |\sum_n s(n)W(m-n)\exp^{-j\omega n}|^2 \qquad (1)$$

The kth cepstral trajectory sampled at the same rate as the speech is then given by:

$$c_k(m) = \frac{1}{2\pi}\int_{-\pi}^{\pi} \log S(\omega, m)\exp(-j\omega k)d\omega \qquad (2)$$

Current speech recognition systems compute cepstral features every centisecond which amounts to down-sampling each cepstral trajectory to a sampling rate of 100 samples/s. Straightforward down-sampling leads to aliasing when the frequency content in the cepstral trajectories exceeds the Nyquist frequency, which is 50Hz for a frame rate of 100. Figure 1 is a plot of the time trajectory of the the 11th cepstral coefficient and it spectral content. It clearly shows that for this cepstral dimension the Nyquist sampling rate is much higher than 100 samples/s.

2.1. Overcoming aliasing

The effects of aliasing can be overcome in the following ways:

1. by sampling each dimension of the cepstral stream at

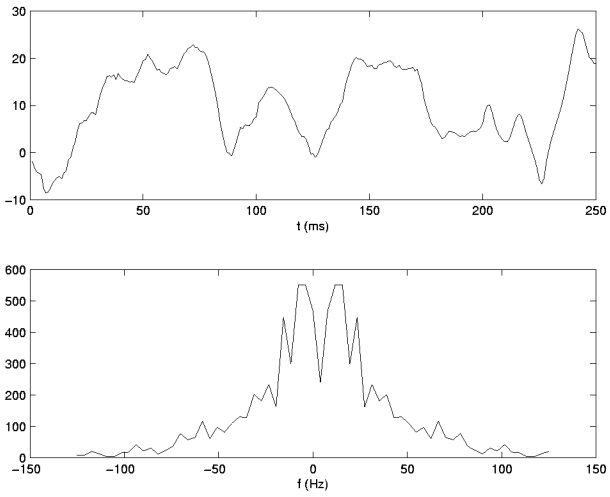

Figure 1: Spectral content in 11th cepstral trajectory

its own Nyquist rate leading to variable rate processing on the dimensions or

2. by sampling the cepstral stream at a very high rate and lowpass filtering each dimension of the cepstral stream before downsampling to 100 frames/s leading to standard fixed frame-rate processing,

3. by processing all cepstral dimensions at the maximal Nyquist rate across dimensions which leads to fixed-rate processing at a higher rate.

The first approach is the preferable one, since it implies no redundancy and no loss of in "information". It is worth noting that this notion of variable rate processing along dimensions is different from the standard notion of variable rate processing where the local frame rate is selected based on the local phonetic unit in question. Since acoustic models in speech recognizers today assume, from modeling and practical considerations, fixed-rate signal processing, we only explore the second approach in this paper. The last approach is presently unattractive due to increased computational burden at high frame rates.

Lowpass filtering before down-sampling is a standard technique to remove distortions due to aliasing. Averaging or mean filtering is one simple example of lowpass filtering where a smoothed cepstral stream is generated by taking an L-point average of the surrounding cepstral vectors in a high rate cepstral stream i.e.,

$$\hat{c}_k(m) = \frac{1}{L}\sum_{l=m}^{m+L} c_k(l) \tag{3}$$

The final feature stream is obtained by downsampling $\hat{c}_k(m)$ do the desired frame-rate. A simple extension would be to use a more general low pass filter with cutoffs suitable for the frame-rate in question instead of an averaging filter. That is,

$$\hat{c}_k(m) = \sum_{l=m}^{m+L} c_k(l)h(l) \tag{4}$$

Instead of averaging (or filtering) the cepstral trajectories one could average the spectral trajectories and then compute the cepstrum from it i.e.,

$$\hat{S}(\omega, m) = \frac{1}{L}\sum_{l=m}^{m+L} S(\omega, l) \tag{5}$$

$$\tilde{c}_k(m) = \frac{1}{2\pi}\int_{-\pi}^{\pi} \log \hat{S}(\omega, m)\exp(-j\omega k)d\omega \tag{6}$$

Spectral averaging is well known as the Welch's method[3] in the spectrum estimation literature for reducing the variance power spectral estimate. Reduction in spectral variance translates directly into a reduction in the variances of the Gaussian mixtures used in the acoustic models leading to tighter models. Reduction in variance can also lead to better class separability which is highly desirable in a speech recognition system. Smoothing the spectrum or cepstrum can also result in the reduction of the ill-effects of noise, especially when the noise is white or in the high frequency bands. Our work was originally motivated by these considerations.

It is well-known that the cepstral features (especially C1) are sensitive to shifts in the speech signal. Cepstral/Spectral smoothing alleviates this problem to some extent.

2.2. Data-driven filtering (LDA)

The variable rate nature of information in the cepstral trajectories can also be captured by a combination of filtering and increasing the dimensionality of the feature space. This can be done in the following two ways:

1. by examining the spectral content in each of the cepstral trajectories and increasing the number of filterbanks for that dimension if a significant amount of energy is outside the original lowpass filter's passband

2. by sampling the cepstral trajectories at a high rate and computing an LDA on blocks of this higher rate stream. The LDA is computed every centisecond to generate a fixed low rate stream.

The LDA based method has the advantage that it is purely-data driven. The hope is that some of the LDA dimensions capture fast variations in the cepstra while some others the slow variations in the cepstra. This approach is similar in principle to DISCO filtering proposed by Avendano et. al. [2]. However, here the analysis window for LDA is of a shorter duration and furthermore it is done on a higher frame rate feature stream.

3. EXPERIMENTS

All experiments were conducted on the IBM rank-based LVCSR system. The IBM LVCSR system uses context-dependent sub-phone classes which are identified by growing a decision tree using the training data and specifying the terminal nodes of the tree as the relevant instances of these classes [4, 5, 6]. The training feature vectors are poured down this tree and the vectors that collect at each leaf are modeled by a mixture of Gaussian pdf's, with diagonal covariance matrices. Each leaf of the decision tree is modeled by a 1-state Hidden Markov Model with a self loop and a forward transition. Output distributions on the state transitions are expressed in terms of the rank of the leaf instead of in terms of the feature vector and the mixture of Gaussian pdf's modeling the training data at the leaf. The rank of a leaf is obtained by computing the log-likelihood of the acoustic vector using the model at each leaf, and then ranking the leaves on the basis of their log-likelihoods.

Results were obtained on two different systems. The first system was trained on 100 hours of the HUB4 Broadcast News data. Overall, the decision tree had 5700 leaves and the system had approximately 90,000 Gaussians. A portion (balanced across all conditions) of the 1997 test set was used as the test set. The baseline results are obtained using feature vectors consisting of 60 components derived from 24 mel-frequency cepstral components using LDA and a global transform. The analysis window was 25ms long and frame-shift was 10ms leading to a rate of 100 frames/s. Next, 24 dimensional cepstral features were generated at a high rate of 500 frames/s (frame-shift of 2ms). An averaged cepstral stream was generated by an 5-point average followed by downsampling by a factor of 5 (Avg Ceps). Spectral averaged (Welch) and filtered (LP) features were generated as described in the previous section. The lowpass filter was a 9-tap filter designed using a constrained least squares approach. Frequency responses of the mean filter and the lowpass filter are shown in Figure 2. In all cases the same LDA and transform matrices that were used in the baseline system were used to generate 60 dimensional features. For each feature stream the means and the variances of the Gaussians and the transition probabilities of the HMM's were re-estimated using a Baum-Welch procedure.

A histogram of the ratios of the variances of the Gaussians in the baseline system and the system using the averaged cepstra, shown in Figure 3, clearly shows a reduction in the variances as expected. Results with the different feature streams are summarized in the Table 1

The averaged cepstra system gave performance improvements in all conditions with an relative improvement of 7% in the F0 (clean, prepared speech). It is notable that lowpass filtering the cepstra with a strict lowpass filter does not give better results than averaging the cepstra. This seems to suggest that aliasing in a reduced form (because of the large sidelobes in the mean filter) seems better than throwing away high frequency information altogether. A similar experiment was conducted on a voice-mail transcription task. Details of this experiment can be found in [7].

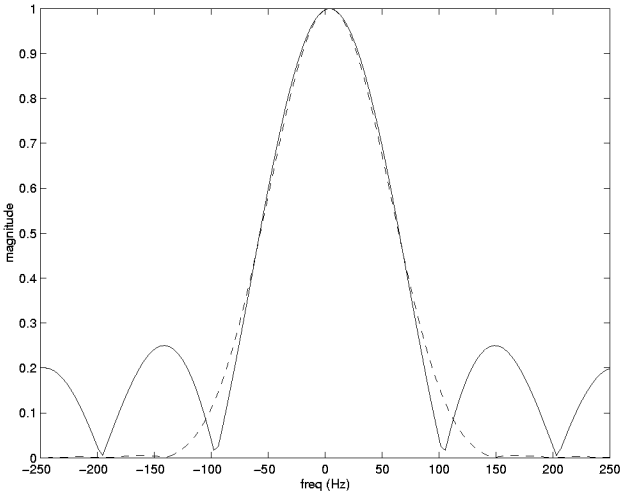

Figure 2: Mean (solid line) and Constrained LP (dashed) filter responses

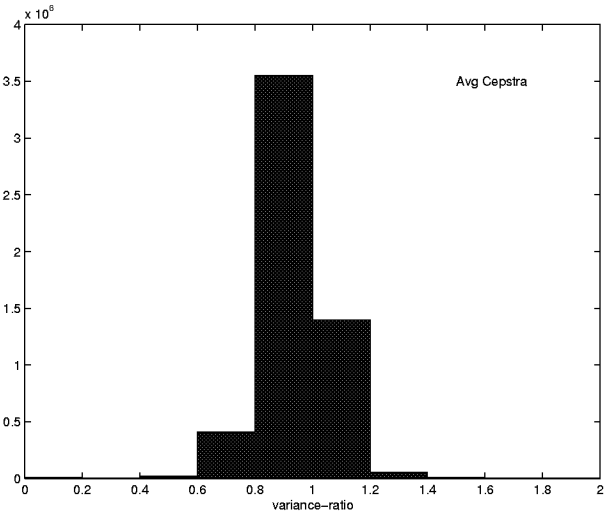

Figure 3: Histogram of $\frac{\sigma_{avg}}{\sigma_{base}}$

The effect of cepstral smoothing and filtering on accuracy and noise robustness was also studied on a internal IBM task. It is a speaker independent task using read speech data recorded in clean environments with the same microphone. The training data consists of 1670 speakers with a total of 36272 sentences. The decision tree for this systems has 2755 leaves with an average of 12 Gaussians modeling each leaf. The test set consists of 10 speakers each uttering 61 sentences, giving a total of around 11000 words in the test set. Table 2 summarizes the results on this test set under clean and noisy conditions. Additive Gaussian noise was used only during testing.

System	Overall	F0	F1	F2	F3	F4	F5	FX
Baseline	26.8	13.2	23.6	32.1	28.5	28.4	24.0	44.6
Avg Ceps	26.3	12.3	23.5	32.0	28.2	27.8	23.8	43.4
Welch	26.6	12.3	23.3	33.0	29.0	27.5	26.0	43.1
LP filter	26.7	13.0	23.0	32.4	29.4	28.3	26.2	43.4

Table 1: Word Error Rate on Hub4 Task: F0 - prepared, F1 - spontaneous, F2 - telephone, F3 - music in background F4 - noise in background, F5 - non-native and FX - other data

System	0dB	15dB
Baseline	12.54	44.57
Avg Ceps	12.45	37.59

Table 2: Comparison of baseline system with cepstral averaging in additive white Gaussian noise

These results indicate only a slight improvement in recognition accuracy with the averaged cepstra. Both systems degrade heavily with noise.

4. CONCLUSIONS AND FUTURE WORK

This paper addresses the temporal variability in speech signals that typically suggests the use of variable-frame-rate processing of speech. We propose the use of spectral/cepstral averaging and faster-rate LDA to account for the temporal variability with fixed-rate processing. Smoothing in the spectral domain results in a reduction in the variance of the short term spectral estimates which directly translates to reduction in the variances of the Gaussians in the acoustic models. These techniques give modest improvements in both word error rate and robustness to noise on some large vocabulary speech recognition tasks.

5. REFERENCES

1. H. Hermansky and N Morgan, "RASTA Processing of Speech," IEEE Transactions on Speech and Audio Processing, Vol. 2, No. 4, pp. 587-589, October 1994.
2. C. Avendano et. al. (1996). *Data Based Filter Design for RASTA-like Channel Normalization in ASR*, Proc. ICSLP'96, Philadelphia, pp. 2087–2090.
3. P.D. Welch, "The use of Fast Fourier Transform for the Estimation of Power Spectra: A Method Based on Time Averaging over Short Modified Periodograms," IEE Trans. Audio Electroacoust., vol. AU-15, pp. 70-73, June 1967.
4. L.R. Bahl and P.V. deSouza and P.S. Gopalakrishnan and D. Nahamoo and M.A. Picheny, "Robust methods for context-dependent features and models in a continuous speech recognizer," in *Proc., Intl Conf. on Acoust., Speech, and Sig. Proc.*, 1994.
5. P.S. Gopalakrishnan and L.R. Bahl and R. Mercer, "A tree search strategy for large vocabulary continuous speech recognition," in *Proc., Intl Conf. on Acoust., Speech, and Sig. Proc.*, 1995.
6. L. R. Bahl et al ., "Performance of the IBM large vocabulary continuous speech recognition system on the ARPA wall street journal task," in *Proc., Intl Conf. on Acoust., Speech, and Sig. Proc.*, pp. 41–44, 1995.
7. M. Padmanabhan, B. Ramabhadran, S. Basu, "Speech recognition performance on a new Voicemail transcription task," Proc. ICSLP'98, Sydney.