CHANNEL NORMALISATION USING PHASE-CORRECTED RASTA

Johan de Veth *Louis Boves*

email: deveth@let.kun.nl,
A2RT, Department of Language and Speech, University of Nijmegen
P.O. Box 9103, 6500 HD Nijmegen, THE NETHERLANDS

ABSTRACT

Recently, we proposed an extension to the classical RASTA technique. The new method consists of classical RASTA filtering followed by a phase correction operation. In this manner, the influence of the communication channel is as effectively removed as with classical RASTA. However, our proposal does not introduce a left-context dependency like classical RASTA. Therefore the new method is better suited for automatic speech recognition based on context-independent modeling with Gaussian mixture hidden Markov models. In this paper we introduce an implementation of phase-corrected RASTA suited for real-time processing. In the context of connected digit recognition over the phone using context-independent phone-based models, we show that word error rate for this implementation is more than 20% lower compared to a real-time implementation of the cepstrum mean subtraction channel normalisation method.

1. INTRODUCTION

For automatic speech recognition (ASR) over the telephone it is well-known that the recognition performance may be seriously degraded due to the transfer characteristics of the handset microphone and the telephone channel [1]. In order to reduce the influence of the linear filtering effect of the communication channel, different channel normalisation (CN) techniques have been proposed (for example [2, 3, 4]). In two recent papers [5, 6] we presented a new, extended version of the classical RASTA filtering technique [3].

Classical RASTA filtering features two important properties: (1) attenuation at low modulation frequencies and (2) enhancement of the dynamic parts of the spectrogram [3]. The first property explains why classical RASTA filtering is such an effective method for CN: In the cepstral or log-energy domain, linear filtering by a quasi-stationary communication channel gives rise to an additive constant bias term [1]. The attenuation at low modulation frequencies effectively removes this DC-component. It has been suggested that the second property is also benificial for good recognition performance [3]. Recently, it was shown that the enhancement of the dynamic parts of the spectrogram obtained by classical RASTA represents a crude approximation of the effects of temporal forward masking in human auditory perception [7, 8]. Thus, classical RASTA may be viewed as a combination of CN and a crude model of human auditory time-masking.

The method we proposed in [5, 6] consists of classical RASTA filtering followed by a phase correction operation. The phase correction is chosen such that the frequency-dependent non-linear phase-shift of the classical RASTA filter is compensated, while at the same time preserving the original magnitude response of the classical RASTA filter. In this manner phase-corrected RASTA effectively removes the influence of the communication channel and at the same time does not enhance the dynamic parts of the spectrogram (i.e. does not model human auditory time-masking). In addition, phase-corrected RASTA removes the well-known left-context dependency introduced by classical RASTA. Therefore, one may expect that the new CN method is better suited for ASR based on context-independent (CI) modeling. In the context of connected digit recognition over the phone, we showed that indeed phase-corrected RASTA can outperform classical RASTA depending on the acoustic resolution of the models [6]. In addition, we showed that phase-corrected RASTA performs as well as cepstrum mean subtraction [5]. These results were obtained while processing the utterance as a whole. Clearly, such an implementation is not suited for a real-time application. In this paper, we discuss results for an implementation of the phase-corrected RASTA technique suited for real-time processing.

This paper is organised as follows. In section 2 we shortly describe the phase-corrected RASTA method and introduce an implementation suited for real-time processing. Next, in section 3, the signal processing for our experiments is described. The telephone database that we used for our experiments is discussed in section 4. After this, the topology of the hidden Markov models (HMMs), the way we performed training with cross-validation and the recognition syntax during testing are described in section 5. The results of our recognition experiments are discussed in section 6. As we will see, these experiments show that a real-time implementation of the new CN method can outperform a real-time implementation of the cepstrum mean subtraction method when using CI HMMs. Finally, in section 7 we sum up the main conclusions.

2. PHASE-CORRECTED RASTA

Consider the signal shown in the upper panel of Figure 1 (we took a synthetic signal instead of a real MFCC coordinate time series for didactic purposes). The signal is a sequence of seven stationary segments ("speech states") preceded and followed by a rest state ("silence"). Notice that the speech states contain a constant overall DC-component (representing the effect of the communication channel). The RASTA filtered version of this signal is shown in the middle panel of Figure 1. Two important observations can be made. First, the DC-component has been effectively removed (at least for times larger than, say, 70 frames). Second, the shape of the signal has been altered.

With regards to the shape distortion the following can

In *Proc. ESCA-NATO Workshop, Apr 17-18, Pont-a-Mousson, France*, pp. 119-122, 1997

be noticed. First, the seven speech states of the signal that had a constant amplitude are now no longer stationary. Instead, the amplitude for each state shows a tendency to drift towards zero. Thus: RASTA filtering steadily decreases the value of cepstral coefficients in stationary parts of the speech signal, while the values immediately after an abrupt change are preserved. This explains the observation that the dynamic parts in the spectrogram of a speech signal are enhanced by RASTA filtering [3]. As a consequence of this drift, however, a description of the signal in terms of stationary states with well-located means and small variances becomes less accurate. Second, the mean amplitude of each state has become a function of the state itself as well as the amplitudes of states immediately preceding it. This is the well-known left-context dependency introduced by the RASTA filter [3]. Because the absolute ordering of signal amplitudes is lost, states can no longer be straightforwardly characterised by their mean amplitude (compare speech states two, five and six before and after RASTA filtering in the upper and middle panel of Figure 1). For this reason, RASTA is less well suited when using CI models (cf. the remarks in [3]). Finally, we mention a third aspect of the shape distortion for completeness (which we feel is less important though). Due to the small attenuation of high-frequency components, abrupt amplitude changes are smoothed.

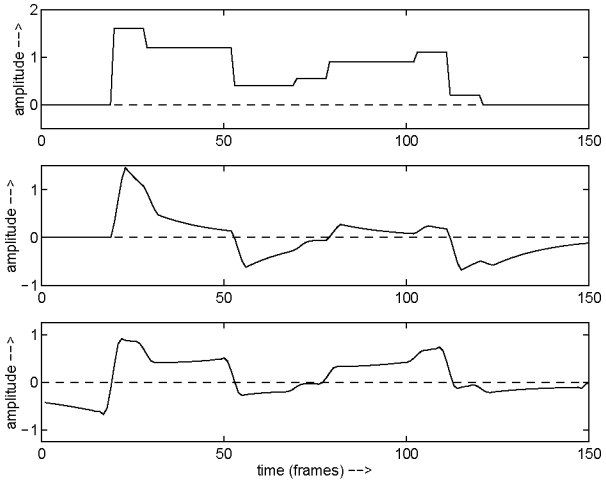

Figure 1: Synthetic signal representing one of the cepstral coefficients in the feature vector. Upper panel: Original signal containing a time-invariant DC-offset. Middle panel: RASTA filtered signal. Lower panel: Phase corrected RASTA filtered signal.

In two recent papers [5, 6] we showed that the nonlinear phase response of the classical RASTA filter is the main cause of the shape distortions observed in the middle panel of Figure 1. Furthermore, we showed how an all-pass phase correction filter may be calculated, such that the phase distortion of the original RASTA filter is compensated, while at the same time preserving the original magnitude response. The result of applying the phase correction filter is shown in the time-domain in the lowest panel of Figure 1. As can be seen, the shape of the phase-corrected RASTA filtered signal resembles the shape of the original signal much better compared to the RASTA filtered signal. The phase correction (1) removes the amplitude drift towards zero in stationary parts of the signal and (2) removes the left-context dependency. In other words, phase-corrected RASTA (1) does not feature enhanced spectral dynamics and (2) is better suited for CI modeling.

The phase correction filter was implemented as a pole-zero filter. Since this pole-zero filter was unstable, we applied the inverse of this pole-zero filter to the time-reversed signal after which a second time-reversal operation was performed [5, 6]. Clearly, this method cannot be used in a real-time application because the first time-reversal operation ideally requires that the whole signal is available.

The same problem arises in case CN is based on cepstrum mean subtraction (CMS). In that case a short window of fixed length (say) L frames preceding the current frame is commonly used in real-time applications (see for example [9]). In this manner a running mean is used instead of the mean over the whole utterance. In practice this can be implemented as a FIR filter with coefficients $b_0 = 1, b_j = \frac{-1}{L}$ for $j = 1, \ldots, L$. The time domain effects of CMS(L=whole utterance) and CMS(L=45 frames) are shown together in the upper panel of Figure 2 (we again used the synthetic signal of Figure 1). Notice that the fixed length implementation CMS(45) introduces deviations from the ideal CMS(whole) signal shape. Even for moments later than L (i.e. the length of the FIR filter) such deviations are present. Thus, the deviations are not transients of the filtering operation.

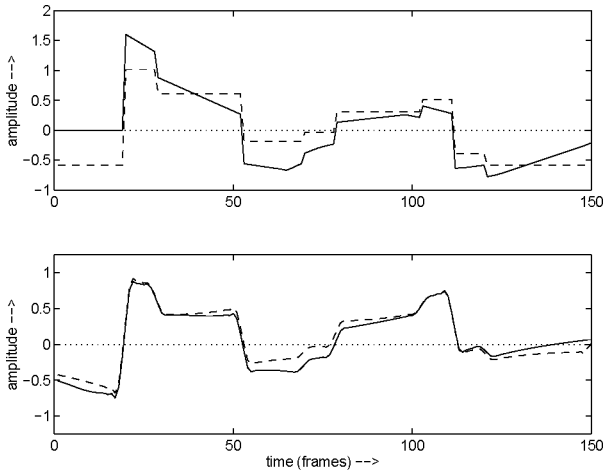

Figure 2: Upper panel: Time domain result of CMS applied over the whole utterance (dashed line) and using a finite length mean calculated over 45 frames (solid line). Lower panel: Time domain result of phase-corrected RASTA applied over the whole utterance (dashed line) and using a finite length estimate calculated over 45 frames (solid line).

In case of our phase-corrected RASTA method (shorter: pcR), we followed an approach similar to the one discussed above for CMS. The implementation of pcR suited for real-time application also uses a fixed length window to do the time-reversal operation. The time-domain effects of pcR(whole) and pcR(45) are shown in the lower panel of Figure 2. As can be seen, these two curves are in good agreement with each other, although small differences are visible. In a number of recognition experiments using CI HMMs we tested the performance of pcR(whole) and pcR(45). For comparison, we tested CMS(whole) and CMS(45) as well. The details of these experiments are described in the next sections.

Table 1. Phonemic transcriptions (column 2) and the number of realisations (columns 3 till 6) of each digit.

digit	transcription	trn960	trn480	cv240	tst671
nul	n Y l	590	294	136	412
een	e n	590	286	165	397
twee	t w e	591	296	181	416
drie	d r i	597	299	155	419
vier	v i r	569	284	135	388
vijf	v Ei f	573	273	124	402
zes	z E s	578	301	136	400
zeven	z e v Q n	582	270	130	380
acht	a x t	554	297	151	374
negen	n e x Q n	534	281	121	435

3. SIGNAL PROCESSING

Speech signals were digitized at 8 kHz and stored in A-law format. After conversion to a linear scale, preemphasis with factor 0.98 was applied. A 25 ms Hamming analysis window that was shifted with 10 ms steps was used to calculate 24 filterband energy values for each frame. The 24 triangular shaped filters were uniformly distributed on a mel-frequency scale. Finally, 12 mel-frequency cepstral coefficients (MFCC's) were derived. In addition to the twelve MFCC's we also used their first time-derivatives (delta-MFCC's), log-energy (logE) and its first time-derivative (delta-logE). In this manner we obtained 26-dimensional feature vectors. Feature extraction was done using HTK v1.4 [10].

We applied the CN techniques to the twelve MFCC coordinates of the feature vector in this paper. We used integration factor -0.94 [3] in case of our phase-corrected RASTA method. We always kept the original values of delta-MFCC's, logE and delta-logE.

4. DATABASE

The speech material for this experiment was taken from the Dutch POLYPHONE corpus [11]. Speakers were recorded over the public switched telephone network in the Netherlands. Handset and channel characteristics are not known; especially handset characteristics are known to vary widely. The speakers were selected in such a way that all major dialect backgrounds in the Netherlands are represented. None of the utterances used for training or test had a high background noise level.

Among other things, the speakers were asked to read a connected digit string containing six digits. We divided this set of digit strings in three parts. For training we reserved a set of 960 strings, i.e. 80 speakers (40 females and 40 males) from each of the 12 provinces in the Netherlands (denoted trn960 in short). An independent set of 240 utterances (cv240; 120 females, 120 males) was set apart for cross-validation tests and was used during our training procedure. As a third independent set we took 671 utterances for final testing of the models (tst671; 341 females, 330 males). (In principle we wanted to have 30 female and 30 male speakers from each of the 12 provinces, but the very sparsely populated province of Flevoland provided only 11 female and 0 male test speakers for tst671). For proper initialisation of the models, we manually corrected automatically generated begin- and endpoints of each utterance in the trn960 data set.

For the experiments in this paper we did not use all available training material. We restricted ourselves to using only half the amount of training data (i.e. 480 utterances, trn480; 240 females, 240 males). We listed the number of available realisations of each digit for all of our data sets in columns 3 till 6 of Table 1.

5. MODELS

5.1. Model topology

The digit set of the Dutch language was described using 18 CI phone models. In addition, we used four models to describe silence, very soft background noise, other background noise and out-of-vocabulary speech, respectively. Each CI model consisted of three states. The total number of different states describing the digit HMMs was 56. All HMMs were left-to-right, where only self-loops and transitions to the next state are allowed. The emission probability density functions were described as a continuous mixture of 26-dimensional Gaussian probability density functions (diagonal covariance matrices). In order to be able to study the recognition performance as a function of acoustic resolution, we used mixtures containing 1, 2, 4 and 8 Gaussians for the emission probability density function of each state.

5.2. Training and recognition

The models were initialised starting from a linear segmentation within the boundaries taken from the hand-validated segmentations. After this initialisation, an embedded Baum-Welch re-estimation was used to further train the models. Starting with a single Gaussian emission probability density function for each state, 20 Baum-Welch iterations were conducted; the models resulting from each iteration cycle were stored. Next, the optimal number of iterations was determined using the cv240 data set. For the set of models with the best recognition rate, the number of Gaussians was doubled and again 20 embedded Baum-Welch re-estimation iterations were performed. This process of training with cross-validation was repeated until models with 8 Gaussians per state were obtained.

During cross-validation as well as during recognition with data set tst671, the recognition syntax allowed for zero or more occurrences of either silence or very soft background noise or other background noise or out-of-vocabulary speech in between each pair of digits. At the beginning and at the end of the digit string one or more occurrences of either silence or very soft background noise or other background noise or out-of-vocabulary speech were allowed.

6. EXPERIMENTS

We trained CI HMMs using train set trn480 in the following four experimental conditions: CMS(whole), CMS(45), pcR(whole) and pcR(45). In both pcR cases we used integration factor -0.94. We used test set tst671 to determine the recognition performance of each CN method as a function of the acoustic resolution. The recognition word error rate WER was defined as

$$WER = 1 - \frac{N - S - I - D}{N}, \qquad (1)$$

where N is the total number of digits tested and S, I, D are the number of substitutions, insertions and deletions, respectively.

We found no significant differences between CMS(whole), pcR(whole) and pcR(45), except when 1 Gaussian per state was used. In that case, we found $WER(pcr(45)) = 9.5\%$, $WER(CMS(whole)) = 8.0\%$ and $WER(pcR(whole)) = 7.3\%$. Figure 3 shows the results for pcR(45) and CMS(45). As can be seen, optimal performance is reached at an acoustic resolution corresponding to 4 - 8 Gaussians per state. In this region

pcR(45), pcR(whole) and CMS(whole) are equivalent and all perform significantly better than CMS(45). We found that WER for pcR(45) is more than 20% lower compared to CMS(45) when 8 Gaussians per state are used. In practical applications CMS(whole) and pcR(whole) cannot be used. Therefore, these experiments suggest that pcR(45) is the CN method to be preferred in actual applications.

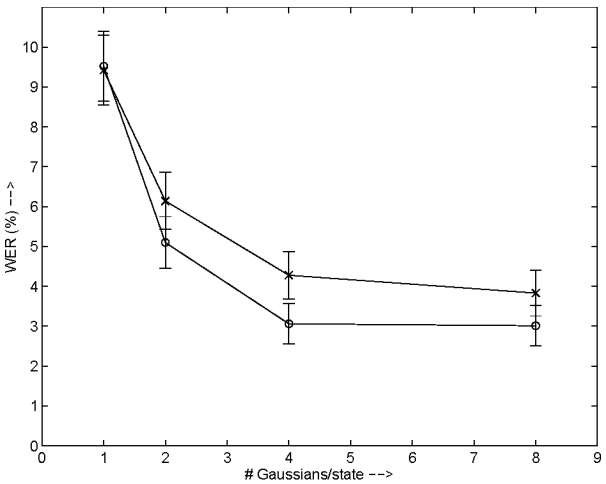

Figure 3: Recognition WER for CMS(45) (X) and phase-corrected RASTA(45) (O) using CI HMMs.

7. CONCLUSIONS

We have proposed a new extension to the classical RASTA CN technique. In our proposal the classical RASTA filter is followed by an all-pass phase correction filter. In this manner the left-context dependency introduced by the classical RASTA filter is removed, while at the same time DC-components are still as effectively removed. Experiments using CI HMMs for connected digit string recognition over the phone, suggest that an implementation of phase-corrected RASTA suited for real-time applications (i.e. pcR(45)) is to be preferred over cepstrum mean normalisation based on a finite fixed window length (i.e. CMS(45)). In the region of 4 - 8 Gaussians per state, our experiments show that pcR(45) significantly outperforms CMS(45). We found that WER for pcR(45) is more than 20% lower compared to CMS(45) when 8 Gaussians per state are used. In addition, we found that pcR(45) is as effective as the ideal CN method CMS(whole).

ACKNOWLEDGEMENT

This work was funded by the Netherlands Organisation for Scientific Research (NWO) as part of the NWO Priority programme Language and Speech Technology.

REFERENCES

[1] H. Hermansky, N. Morgan, A. Bayya & P. Kohn, 'Compensation for the effect of the communication channel in auditory-like analysis of speech', in Proc. Eurospeech-91, 1991.

[2] S. Furui, 'Cepstral analysis technique for automatic speaker verification', IEEE Trans. Acoust. Speech Signal Process., ASSP-29, pp. 254-272, 1981.

[3] H. Hermansky & N. Morgan, 'RASTA processing of speech', IEEE Trans. Speech Audio, 2(4), pp. 578-589, 1994.

[4] J-C. Junqua, D. Fohr, J-F. Mari, T.H. Applebaum & B.A. Hanson, ' Time derivatives, cepstral normalisation and spectral parameter filtering for continuously spelled names over the telephone' in Proc. Eurospeech-95, pp. 1385-1388, 1995.

[5] J. de Veth & L. Boves, 'Comparison of channel normalisation techniques for automatic speech recognition over the phone', in Proc. ICSLP-96, pp. 2332-2335, 1996.

[6] J. de Veth & L. Boves, 'Phase-corrected RASTA for automatic speech recognition over the phone', to appear in Proc. ICASSP-97, Apr. 21-24, Muenchen, Germany, 1997.

[7] H. Hermansky & M. Pavel, 'Psychophysics of speech engineering systems', in Proc. ICPhS-95, pp. 3.42-3.49, 1995.

[8] H. Hermansky, 'Auditory modeling in automatic recognition of speech', ESCA Workshop on the Auditory basis of speech perception, Keele University (UK), 15-19 July, 1996.

[9] R. Haeb-Umbach, P. Beyerlein & D. Geller, 'Speech recognition algorithms for voice control interfaces', Philips J. Res., vol. 49, pp. 381-397, 1995.

[10] S. Young & P. Woodland, 'HTK v1.4 User Manual', Speech Group, Cambridge University Engineering Department, UK, 1992.

[11] E. A. den Os, T. I. Boogaart, L. Boves & E. Klabbers, 'The Dutch Polyphone corpus', in Proc. Eurospeech-95, pp. 825-828, 1995.