Noise Robust Feature Extraction for ASR using the Aurora 2 Database

Qifeng Zhu, Markus Iseli, Xiaodong Cui, Abeer Alwan

Department of Electrical Engineering
University of California, Los Angeles
{qifeng, iseli, xdcui ,alwan}@icsl.ucla.edu

Abstract

Four front-end processing techniques developed for noise robust speech recognition are tested with the Aurora 2 database. These techniques include three previously published algorithms: variable frame rate analysis [Zhu and Alwan, 2000], peak isolation [Strope and Alwan, 1997], and harmonic demodulation [Zhu and Alwan, 2000], and a new technique for peak-to-valley ratio locking. Our previous work has focused on isolated digit recognition. In this paper, these algorithms are modified for recognition of connected digits. Recognition results with the Aurora 2 database show that a combination of these four techniques results in 40% error rate reduction when compared to the baseline MFCC front-end for the clean training condition, with no significant increase in computational complexity.

1. Introduction

This paper focuses on front-end feature extraction approaches for noise robust automatic speech recognition (ASR). Four front-end processing techniques are tested with the Aurora 2 database. These techniques include three previously published algorithms: variable frame rate analysis [3], peak isolation [2], and harmonic demodulation [4], and a new technique for peak-to-valley ratio locking. Our previous work has focused on isolated digit recognition and mainly computer-generated additive noise.

Here, training and testing followed the specifications described in [1]. A word-based ASR system for digit string recognition where each HMM word model has 16 emitting states is adopted. Training is done with either 8440 clean utterances (referred to as clean training) or with 8440 clean and noisy utterances (multi-condition training). A 3-state silence model and a one state short pause model are used. Test data included different kinds of realistic background noise at various SNRs.

The Aurora 2 database CD included a program (FE2.0) to compute the MFCCs and log energy. The front-end used by HTK is MFCC_E_D_A, which contains 12 MFCCs and log energy, and their first and second derivatives. Each feature vector thus contains 39 components. Reference recognition results are computed with FE2.0. The techniques used in this paper were implemented by modifying the code in FE2.0.

2. Noise robust frond-end feature extraction

Three previously published front-ends and a new algorithm are described in this section.

Work supported in part by NSF, HRL and Broadcom through the UC MICRO program, and ST Microelectronics.

2.1. Variable frame rate analysis (VFR) [3]

Variable frame rate (VFR) analysis in [3] is motivated by the fact that changes in spectral characteristics are important cues for discriminating and identifying speech sounds. These changes can occur over very short time intervals. Computing frames every 10 ms, as commonly done in ASR, is not sufficient to capture such dynamic changes. The VFR algorithm increases the frame rate for rapidly-changing segments with relatively high energy and decreases the frame rate for steady-state segments, based on a weighted log energy Euclidean MFCC distance. The smallest frame step can be 2.5 ms. An example is shown in Figure 1. The current implementation uses an average frame rate which is less than 100 frames per second.

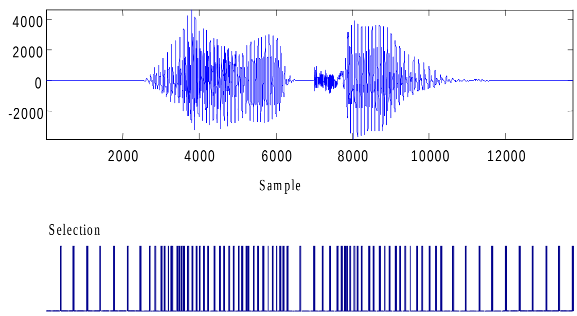

Figure 1: The upper panel shows the utterance “one two”. The lower panel shows the selected frames.

MFCCs computed with the VFR technique reduce the error rate, when compared to reference results, in the clean training condition by 18.59% for Set A, 31.77% for Set B, and -5.58% for Set C. The overall error reduction is 20.33%.

2.2. Peak isolation (PKISO) [2]

This technique involves MFCC liftering, an inverse DCT (IDCT), and half wave rectification. After the IDCT, spectral valleys are often less than 0 and formants are larger than 0. Figure 2 shows an example of the recovered log Mel filter-bank output from liftered MFCCs for a clean and noisy frame of /i/. Half-wave rectification is then applied on the recovered log Mel filter-bank output so that the valleys are effectively removed. A DCT is applied on the rectified log Mel filter-bank output to obtain feature vectors which will be referred to as PKISO_MFCCs.

Error rate reductions with PKISO_MFCCs in the clean training condition are, when compared to reference results, 28.45% for Set A, 44.50% for Set B, and -7.8% for Set C. The overall error reduction is 29.74%.

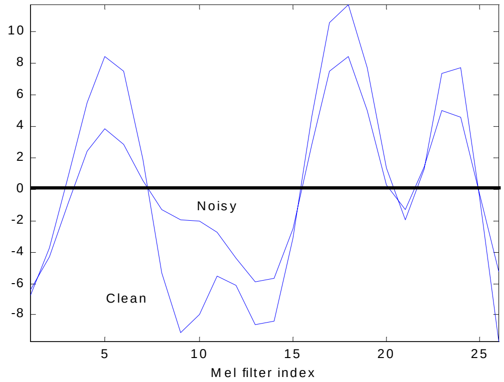

*Figure 2:*Peak isolation. Log Mel filter-bank output recovered from liftered MFCCs for a clean and noisy (0 dB SNR) frame of /i/. After half-wave rectification only the upper part is retained.

2.3. Peak-to-valley ratio locking

We introduce in this paper the concept of peak-to-valley ratio locking. In the presence of noise, spectral valleys will be buried by noise, but formants are, on average, not affected as much. An example can be seen in Figure 3, which shows spectra of a clean and noisy speech frame. The frame is from /i/ in “zero” (female talker) and the additive noise is speech shaped at 0 dB SNR. The noisy spectrum is an average over 150 frames. Note that the spectra are nearly the same at harmonic peaks around the formants, where the amplitude is about 3 –5 times higher than the average noise spectrum. At frequencies where the signal amplitude is low, as in spectral valleys, the average noisy speech spectrum is nearly the same as the average noise spectrum. Because of the difference of the noise effects on valleys and peaks, often the peak-to-valley ratio in spectra of noisy speech is lower than that in clean speech, hence leading to a mismatch between the clean and noisy data.

After obtaining the recovered log Mel filter output from liftered MFCCs (without C_0), as shown in Figure 2, both peaks and valleys will be affected. One approach to addressing this problem is peak-to-valley ratio locking. We set the highest peak of amplitude x to a fixed number p. The entire recovered Mel filter output is then scaled accordingly by a factor of p/x. In our implementation p was set to 10. This number is approximately the average amplitude of the highest peaks across the database.

When used together with peak isolation, only the positive part in the recovered Mel filter output is scaled, and the negative part is set to zero. An example of the result of combining PKISO with peak-to-valley ratio locking is shown in Figure 4.

The error rate reduction with peak-to-valley ratio locking only (without PKISO) in the clean training are: 27.66% for Set A, 41.21% for Set B, and -26.16% for Set C. The overall error reduction is 24.54%.

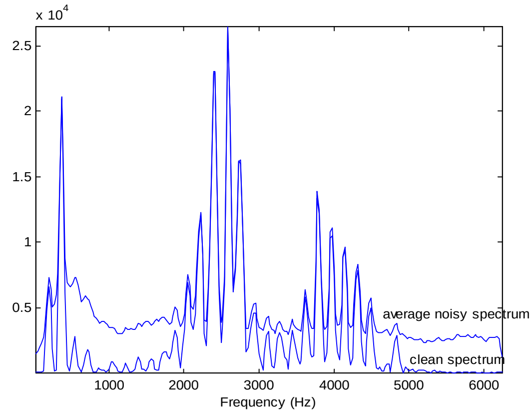

Figure 3: Linear clean and noisy (0 dB SNR) speech spectra.

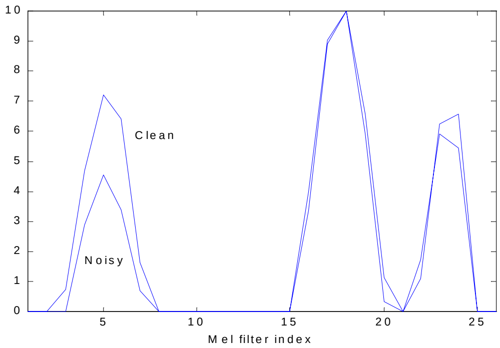

Figure 4: Log Mel filter-bank output after liftering, rectification and peak-to-valley ratio locking for the same frame of /i/ in Figure 2. Notice that the highest peaks are set to 10.

2.4. Harmonic demodulation (HD) [4]

Harmonic demodulation is a method that aims at reducing the difference between clean and noisy speech spectra especially at inter-harmonic valleys.

The LTI speech production model is viewed as amplitude modulation in the frequency domain, with the excitation spectrum being the carrier and the spectrum of the vocal tract transfer function being the modulator. Non-coherent demodulation with non-linear envelope detection is used to recover the spectrum of the vocal tract transfer function [4]. Envelopes of the speech spectra, instead of the speech spectra themselves, are used to compute the MFCCs.

3. Modifications for the Aurora 2 database

3.1. Speech/nonspeech detection

Our previous studies on PKISO [2], HD [4], and VFR [3] focused on isolated digits with endpoint detection. All techniques mentioned in this paper assume a speech model, hence speech/nonspeech detection is critical.

A very simple speech/nonspeech detector is implemented by comparing the log-energy of a frame to a threshold for the corresponding utterance. A threshold (T) for one utterance is determined empirically by T=(H+L)*A, where H and L are the average of the 10 highest and 10 lowest log energy values, respectively, and A is either 0.5 (for VFR) or 0.6 (for the other methods).

When the log energy of a frame is higher than T, it is classified as speech, otherwise non-speech. In VFR, frame rate adjustment is only performed on the speech part. For the nonspeech part, the frame step size is set to 25 ms.

3.2. Increasing the variances of the silence model

The HD, PKISO and peak-to-valley ratio locking algorithms remove the mean spectral difference between the clean and noisy speech spectra. For the silence model, however, which is trained with clean data, there will be a large mismatch with the noisy test data. One solution is to increase the variances in the silence and short pause models.

We found that if the reference MFCCs are used, best performance for clean training can be achieved by increasing the variances of the silence model by a factor of 1.1, the overall improvement in accuracy rate is less than 2%. With PKISO, HD, and peak-to-valley ratio locking, we get a better model of the digits, and hence a silence model with larger variance results in a larger increase in recognition performance. The increasing factor we use is 1.2, and the improvement in overall error rate reduction is 3.3%.

3.3. Rasta-like filter in the cepstral domain

The four techniques described in this paper have difficulty with Set C, where the problem is channel mismatch. A Rasta like band pass filter [5] is used at the final stage of the front-end to avoid the mean shift effect caused by channel distortion. This results in a 15% improvement in error rate reduction in Set C, and a 4% improvement for sets A and B.

4. Complexity considerations

Even though our implementation did not optimize for computational cost (but focused on optimizing recognition performance) the increase in computational complexity of PKISO, HD and peak-to-valley ratio locking together is not high. In addition, these algorithms are frame-based and thus do not introduce a delay. The speech/non-speech detection algorithm introduces a delay equal to the utterance duration.

The additional memory cost for HD is 128 floating numbers, which is half of the FFT length and the additional memory for PKISO and peak-to-valley ratio locking together is 23 floating numbers, which is the number of the Mel filters.

The extra operations introduced for processing one frame in HD mainly comes from 7*128 (7 is the length of the filter characteristics and 128 is half the FFT size) floating number multiplications. The extra operations introduced by PKISO and peak-to-valley ratio locking mainly come from the extra IDCT and DCT, which contain 23*12 (23 is the number of the Mel filters and 12 is the length of the MFCC vector) floating number multiplication each. Liftering in PKISO adds 12 multiplications and peak-to-valley ratio locking adds another 23 multiplications in processing one frame.

When tested on a Sun Ultra Sparc 60 workstation, the computation load of PKISO and peak-to-valley ratio locking together is less than 4% of the total computation time of the original front-end (FE2.0) executable. HD adds about 20% more computational time. These measurements only count the front-end computation time, excluding the disk I/O time.

The computational load of VFR is higher than the other algorithms and depends on the number of frames classified as speech in an utterance. The current implementation of VFR requires processing the entire utterance and hence a delay equivalent to the utterance duration is introduced. In VFR, the threshold for frame selection is computed from the inter-frame MFCC distance of the whole utterance, thus a memory size that equals the number of speech frames in an utterance is needed.

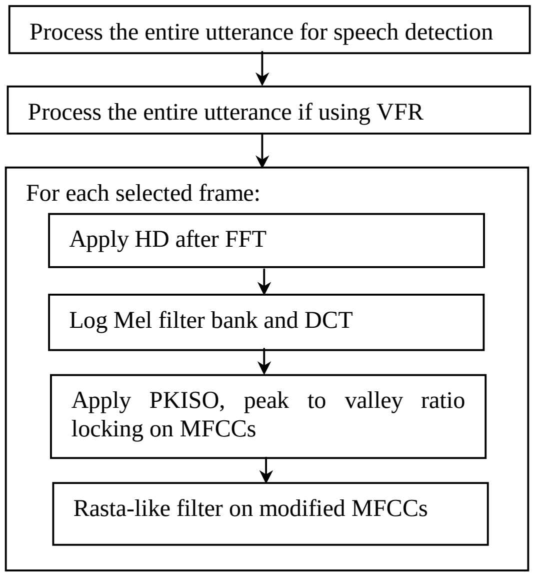

Figure 5: Block diagram of the front-end processing.

5. Recognition results

Recognition experiments were performed with scripts included in the Aurora 2 CD, and with HTK 2.2. Training follows the steps specified in [1]. The four techniques were combined to produce the front-end for ASR as shown in Figure 5. Tables 1 and 2 show the results (word accuracy) for the clean and multi-condition training, respectively, when tested with sets A, B, and C at seven SNRs. Improvements in error reduction, when compared to the reference results with MFCCs, are shown in the last row and rightmost column in each table.

As Table 1 shows, the methods mentioned in this paper (excluding VFR) reduce the overall error rate by 40.47%. Error reductions are 43.4% for Set A, 52.8% for Set B and 1.53% for Set C.

VFR is not included in Table 1 because its computational complexity is high. If VFR is used in combination with the other algorithms, the overall error reduction is 41.74%. Error reductions are 44.5% for Set A, 54.4% for Set B, and 2.34% for Set C.

Aurora 2 Clean Training - Results															
	A					B					C				Percentage
	Subway	Babble	Car	Exhibition	Average	Restaurant	Street	Airport	Station	Average	Subway M	Street M	Average	Overall	Improvement
Clean	98.43	98.46	98.48	98.52	98.47	98.43	98.46	98.48	98.52	98.47	98.43	98.22	98.33	98.44	-61.91%
20 dB	96.10	96.31	96.54	95.37	96.08	96.81	96.37	96.63	96.64	96.61	93.95	94.07	94.01	95.88	13.75%
15 dB	93.34	95.25	94.30	93.30	94.05	95.18	93.92	94.96	94.35	94.60	87.41	90.57	88.99	93.26	42.86%
10 dB	85.17	90.93	89.56	84.94	87.65	89.04	87.48	89.77	89.14	88.86	74.15	78.11	76.13	85.83	52.24%
5 dB	62.97	78.78	78.50	68.19	72.11	70.06	70.59	76.86	72.14	72.41	43.48	60.76	52.12	68.23	44.70%
0 dB	23.24	41.38	57.53	40.70	40.71	31.69	41.81	49.03	49.77	43.08	12.37	31.71	22.04	37.92	23.96%
-5dB	0.49	0.10	22.82	11.26	8.67	0.10	11.94	9.01	13.08	8.53	2.40	8.28	5.34	7.95	-0.67%
Average	72.16	80.53	83.29	76.50	78.12	76.56	78.03	81.45	80.41	79.11	62.27	71.04	66.66	76.22	
	8.78%	61.15%	57.58%	32.09%	43.40%	50.55%	42.92%	60.32%	55.84%	52.80%	-11.50%	14.54%	1.53%		40.47%

Table 1: The clean training results on Aurora 2 database using PKISO, HD and peak-to-valley ratio locking.

Aurora 2 Multicondition Training - Results															
	A					B					C				Percentage
	Subway	Babble	Car	Exhibition	Average	Restaurant	Street	Airport	Station	Average	Subway M	Street M	Average	Overall	Improvement
Clean	98.37	98.40	98.30	98.61	98.42	98.37	98.40	98.30	98.61	98.42	98.25	98.43	98.34	98.40	-8.57%
20 dB	97.67	97.85	98.21	97.01	97.69	97.33	97.31	97.91	97.28	97.46	97.14	96.40	96.77	97.41	2.21%
15 dB	96.68	97.04	97.49	96.42	96.91	95.95	96.52	97.23	95.80	96.38	95.73	95.25	95.49	96.41	2.43%
10 dB	94.26	94.98	95.62	93.27	94.53	92.82	93.95	94.36	92.53	93.42	92.29	90.66	91.48	93.47	-5.22%
5 dB	88.52	88.03	88.01	85.81	87.59	83.39	85.22	86.37	83.40	84.60	82.62	79.69	81.16	85.11	-2.60%
0 dB	69.39	63.45	59.17	62.08	63.52	57.41	62.27	65.37	58.04	60.77	51.55	53.20	52.38	60.19	2.59%
-5dB	30.64	27.75	22.31	24.16	26.22	24.26	27.09	28.69	23.02	25.77	18.67	23.34	21.01	24.99	0.64%
Average	89.30	88.27	87.70	86.92	88.05	85.38	87.05	88.25	85.41	86.52	83.87	83.04	83.45	86.52	
	4.87%	2.66%	8.73%	-9.29%	1.91%	-0.08%	0.12%	4.90%	2.67%	1.84%	3.71%	-8.09%	-2.00%		0.95%

Table 2: The multicondition training results on Aurora 2 database, only harmonic demodulated MFCCs are used.

As shown in Table 1, the algorithms degrade slightly the performance of the clean and –5 dB SNR conditions (from 99% to 98.4%, and 8.53% to 7.95%, respectively). But for all other SNRs (20 to 0 dB) the performance is improved significantly. Error reduction is best with babble (61%), airport (60%), car (58%), train station (56%), and restaurant (51%) background noise.

Most of the techniques reported in this paper (PKISO, peak-to-valley ratio locking, and HD) aim at reducing the difference of the means between clean and noisy speech spectra. For the multicondition training, the mean shift is not a problem. Hence, these algorithms do not improve recognition results for multicondition training.

Generally, these techniques are designed for mismatched training and testing conditions but not designed for matched conditions. In addition, half-wave rectification in the peak isolation algorithm appears to be harmful and leads to an overall 40% error rate increase for multicondition training. We also found that peak-to-valley ratio locking improves the performance of Set A by 4% but decreases those of Sets B and C by about 7%. Although HD was not designed for multi-condition training, it does help slightly in Sets A and B because it effectively removes pitch information. Table 2 shows the results in multicondition training with HD only. Error reductions are 1.9% for Set A and 1.8% for set B, and –2% for Set C. The overall error reduction is 0.95%. This improvement is small. Cepstral mean subtraction (CMS) can achieve a higher error rate reduction by 13% overall (but does not improve the clean training condition).

VFR also does not improve recognition performance for the multicondition training mainly because of the imperfect speech/nonspeech detection algorithm. When silence is classified as speech, due to high noise background in the training set, a high frame rate is applied, and that results in inaccurate models. VFR resulted in a 28% increase in the error rate for multi-condition training.

We believe that improving the speech/nonspeech algorithm could result in improvements in recognition performance, especially for the VFR algorithm. This is especially true for the subway and street noise, which are bursty. In fact, we tried a computationally costly pitch based speech detection algorithm which was tailored for subway noise and the accuracy rate in Set A for the clean training condition improved from 72% to 77.5%.

6. References

[1] H. G. Hirsch and D. Pearce, "The AURORA Experimental Framework for the Performance Evaluations of Speech Recognition Systems under Noisy Condition", *ISCA ITRW ASR2000 "Automatic Speech Recognition: Challenges for the Next Millennium", France, 2000.*

[2] B. Strope and A. Alwan, "A model of dynamic auditory perception and its application to robust word recognition," *IEEE Trans. on Speech and Audio Processing, Vol. 5, No. 5, p. 451-464, 1997.*

[3] Q. Zhu and A. Alwan, "On the use of variable frame rate analysis in speech recognition", *Proc. IEEE ICASSP, Turkey, Vol. III, p. 1783-1786, June 2000.*

[4] Q. Zhu and A. Alwan, "Amplitude Demodulation of Speech Spectra and its Application to Noise Robust Speech Recognition", *ICSLP 2000. Vol. 1, pp. 341-344.*

[5] H. Hermansky and N. Morgan, "RASTA processing of speech". *IEEE Trans. Speech and Audio Proc., Vol.2, (no.4), Oct. 1994. p.578-89.*