

# IMPROVED FEATURE VECTOR NORMALIZATION FOR NOISE ROBUST CONNECTED SPEECH RECOGNITION

*J. Häkkinen, J. Suontausta, R. Hariharan, M. Vasilache, K. Laurila*

Nokia Research Center, Speech and Audio Systems Laboratory, Tampere, Finland

Email: {juha.hakkinen, janne.suontausta, ramalingam.hariharan,  
marcel.vasilache, kari.laurila}@research.nokia.com

## ABSTRACT

Feature vector normalization has been successfully used to improve the noise robustness of speech recognizers. Unfortunately, it may cause additional insertion errors in connected digit recognition in clean environments. We propose two methods to reduce the number of insertions. Based on estimated instantaneous signal-to-noise ratio we form a reliability measure for the recognized digits. We discard unreliable digits from the beginning and the end of the recognized digit sequence. Since the proposed reliability hypotheses are independent of the likelihoods produced by an HMM classifier, we are capable of bringing new useful information into the classification process. In addition, we constrain the normalization process on the basis of statistics obtained from the training data. Experimental results show that we are capable of achieving an average 32% string level error rate reduction in simulations of a noisy car environment.

## 1. INTRODUCTION

Connected digit recognition is a fundamental building block in human-machine spoken dialogue systems. Compared to many other tasks digit recognition is very challenging since tolerance to recognition errors is minimal. Errors that occur in connected digit recognition include substitutions, deletions and insertions. Substitution is a case when the digit spoken by the user is recognized as another digit. A deletion occurs when the digit spoken by the user is not recognized at all. Insertion means that a digit is recognized though the user did not say any digit.

Earlier we proposed a feature vector normalization scheme which was shown to reduce error rates in noise robust connected digit recognition [1]. More detailed analysis of the scheme revealed that a more significant performance gain was hindered by an increased number of insertion errors due to the scheme.

In the literature, several methods to minimize the amount of classification errors have been proposed [2].

Also methods dealing directly with the insertion error problem have been designed. Many insertion errors can be avoided by digit model adaptation based on speech-silence discrimination as in [3]. Robust digit string verification using filler models has been explored by Rahim *et. al.* [4]. The objective of using filler models is to model all out-of-vocabulary sounds. Rahim *et. al.* used a likelihood ratio test defined as the difference between the log-likelihoods of the digit models and a combination of a generic filler model and digit-dependent filler models. They also experimented with MCE/GPD training of the filler models achieving less overlapping histograms for the confidences of in-class/out-class strings. Sukkar *et. al.* [5] have also used discriminative training of the filler models to improve their rejection performance. They also extended the verification framework to include error correction when  $N$ -best string hypotheses are available.

By extending our connected digit recognizer presented in [6], to include discriminative training against insertion errors we achieved only modest improvements. Also, methods based on the confidence score did not produce good results. In this paper, we propose a method to detect and correct insertion errors. Moreover, by introducing a gain constraint to the normalization process, we are able to further reduce the number of insertion errors.

## 2. CONNECTED DIGIT RECOGNIZER

The connected digit recognizer that we are using is based on HMMs. We use feature vectors consisting of 13 FFT-based MFCCs (including the zero'th cepstral coefficient) and their first and second order time derivatives. To increase noise robustness the feature vectors are normalized using the scheme presented in [1]. Essentially, the normalization scheme is aimed at removing the short-term mean component and adjusting the gain to unity. This is done as follows:

Initial mean and standard deviation estimates are determined for each feature vector component  $i$  as

$$m_N(i) = \frac{1}{N} \sum_{f=1}^N o_f(i) \quad \text{and} \quad (1)$$

$$\sigma_N(i) = \sqrt{\frac{1}{N} \sum_{f=1}^N [o_f^2(i)] - [m_N(i)]^2} = \sqrt{s_N^2(i) - [m_N(i)]^2}$$

where  $o_f(i)$  denotes the  $i$ 'th feature vector component at frame (time)  $f$ .

Finally, for each new feature vector at time  $t$ , the mean and sample square estimates are recursively updated

$$m_t(i) = \lambda \cdot m_{t-1}(i) + (1 - \lambda) \cdot o_t(i) \quad (2)$$

$$s_t^2(i) = \lambda \cdot s_{t-1}^2(i) + (1 - \lambda) \cdot o_t^2(i)$$

and the normalized feature vector is

$$\hat{x}_{t-N}(i) = \frac{x_{t-N}(i) - m_t(i)}{\sigma_t(i)} \quad (3)$$

The recognizer uses duration-constrained left-to-right HMMs without state skips [7]. The garbage model is a one state HMM. Looped grammar, i.e., the exact number of digits to be recognized is unknown, is used.

### 3. INSERTION ERROR PROBLEM

Insertions are usually a problem when looped grammar is used. They are often caused by breath noise, clicks or out-of-vocabulary speech. The problem is amplified by the feature vector normalization as illustrated in Figure 1. In quiet conditions and when there are no digits within the normalization window, even tiny ripples in the waveform produce significant changes in the normalized feature vector trajectories (see Figure 1, in the middle, at around 1.2 seconds).

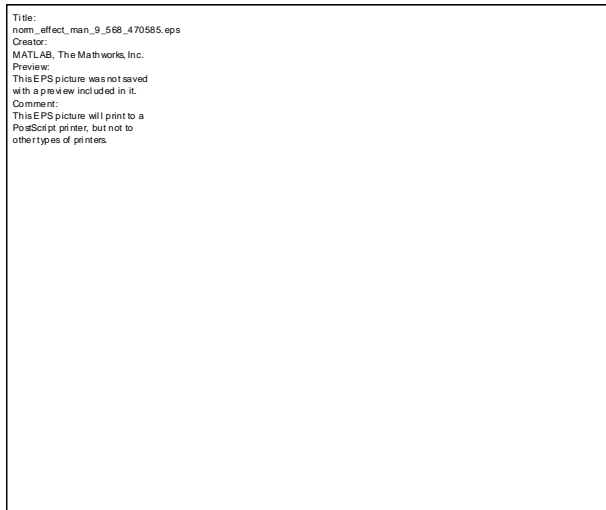


Figure 1. The effect of normalization of feature vectors (MFCC  $C_0$ ). An insertion was observed at around 1.2 sec

When the feature vector normalization algorithm is used there is a tendency towards insertion errors in clean conditions. There is a clear trade-off in balancing the amount of insertion and deletion errors in various environments, since a reduction of insertions in clean

conditions leads to more deletions in noisy environments.

### 4. INSERTION ERROR CORRECTION ALGORITHM AND GAIN CONSTRAINT

Insertion errors due to feature vector normalization can be detected if knowledge about the absolute power level of the signal is available. We consider a digit to be reliably recognized when its estimated signal-to-noise ratio (SNR) is high compared to the other digit candidates. Insertion error detection and correction will be carried out as a post-processing step after the recognition has ended and the recognition result and the alignment of speech frames with the digit models is available.

Most of the insertion errors in the digit strings are in the beginning or in the end of the recognized digit sequence. Therefore, the insertion error correction algorithm is allowed to remove insertions only in the beginning and/or in the end of the digit string. At least two digits should be recognized before the method can be applied. The method uses the difference between the estimated digit SNRs as a reliability measure that is compared against a threshold  $T$ . Assuming that the SNR values  $S_1, S_2, \dots, S_N$  of the  $N$  recognized digits  $d_1, d_2, \dots, d_N$  are available, the insertion error correction (IEC) algorithm works as follows:

1. If  $N \geq 2$  go to step 2, otherwise terminate the algorithm and accept the recognition result.
2. Sort the SNR values  $S_1, S_2, \dots, S_N$  in increasing order. Let the ordered sequence be  $S'_1, S'_2, \dots, S'_N$ .
3. In the sorted sequence, find the maximum SNR difference  $\max\{S'_i - S'_{i-1}\}$ ,  $i = 2, \dots, N$ , and store the corresponding index  $i_{\max}$ . If  $i_{\max} = 3$ , go to step 4, else if  $i_{\max} = 2$ , go to step 5. Otherwise terminate the algorithm and accept the recognition result.
4. Insertions in the beginning and in the end: If  $S'_3 - S'_2 < T$ , terminate the algorithm and accept the recognition result. Otherwise check if  $S'_1$  and  $S'_2$  are the SNR values of the first and last digit. In such case remove the first and last digit from the recognition result.
5. Insertion in the beginning or in the end: If  $S'_2 - S'_1 < T$ , terminate the algorithm and accept the recognition result. Otherwise check if  $S'_1$  is the SNR value of the first/last digit. In such case remove the first/last digit from the recognition result.

#### 4.1 Estimation of SNR values

The estimation of SNR values for each recognized digit is based on the subband powers obtained by subsampling the FFT of the front end. The subband powers are stored and after the recognition has ended, the SNR estimates for the recognized digits are

computed. The maximum power for digit  $d_i$  in the subband  $s$  is denoted by  $P_i^{\max}(s)$ . The mean power of the garbage frames in the subband  $s$  is denoted by  $P^{garb}(s)$ . The SNR estimate of digit  $d_i$  for subband  $s$  is obtained from

$$S_i(s) = 10 \log_{10} \left( P_i^{\max}(s) / P^{garb}(s) \right). \quad (4)$$

Finally, the SNR estimate of digit  $d_i$  is the mean of the  $M$  subband SNR values

$$S_i = \frac{1}{M} \sum_{s=1}^M S_i(s). \quad (5)$$

## 4.2 Computation of the detection threshold

The differences between the SNR estimates of the recognized digits are compared against the threshold  $T$ . The SNR,  $S$ , can be defined as

$$S = 10 \log_{10} \left( \frac{P_S + P_N}{P_N} \right), \quad (6)$$

where  $P_S$  is the signal power and  $P_N$  is the noise power. By rewriting equation (6) the noise power becomes

$$P_N = r P_S, \quad (7)$$

where  $r = 1 / (10^{S/10} - 1)$ . Let two digits  $d_1, d_2$  have powers  $P_S$  and  $c P_S$ ,  $0 < c < 1$ . Then the SNR difference between the digits is

$$10 \log_{10} \left( (P_S + P_N) / P_N \right) - 10 \log_{10} \left( (c P_S + P_N) / P_N \right). \quad (8)$$

By substituting the noise power into the equation, the condition for rejecting the digit with lower power becomes

$$10 \log_{10} \left( (1+r) / (c+r) \right) \geq T. \quad (9)$$

Optimum thresholds are determined for a set of SNRs on a training database. Linear interpolation is used during recognition to obtain the threshold between these values.

## 4.3 Normalization gain constraint

The amplification of feature vector components outside speech regions is due to the large normalization gain (small variance) in these regions. Hence an additional way to reduce these insertions is to limit the gain factor

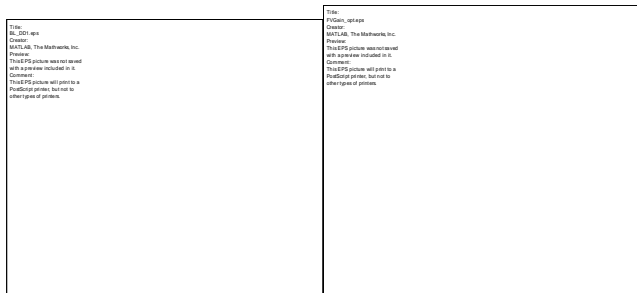


Figure 2. Feature vector component ( $C_0$ ) trajectories of a digit string without (left) and with the gain constraint.

Title: snrHistograms2.eps  
 Creator: MATLAB, The Mathworks, Inc.  
 Preview: This EPS picture was not saved with a preview included in it.  
 Comment: This EPS picture will print to a PostScript printer, but not to other types of printers.

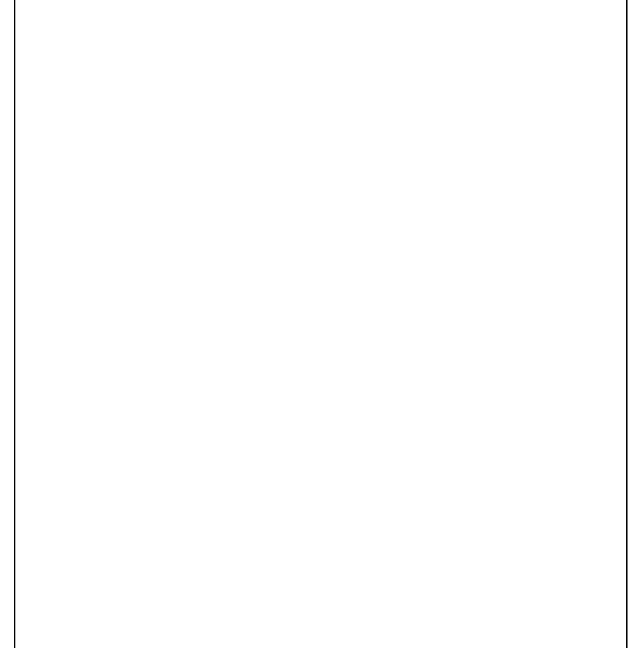


Figure 3. The histograms of the SNR differences of the detected insertion errors and the rest of the utterances in clean (two histograms on the top) and in 10 dB noise (two histograms on the bottom).

to a predetermined value. The maximum gain (minimum variance value in equation (3)) during the speech portion is determined from the complete set of training utterances. A threshold, computed as a percentage of the maximum gain obtained from the training utterances is then used as the upper limit for the gain for speech and non-speech regions during recognition. Figure 2 shows the feature vector component trajectories of a digit string before and after the application of the gain constraint. It can be clearly seen that there is drastic reduction in the gain during the initial and final non-speech regions. This reduces the risk of false insertions in these areas.

## 5. EXPERIMENTS

The IEC algorithm and the gain constraint were tested in a connected digit recognition task on a database composed of utterances spoken by Finnish speakers. The database included 700 speakers whose utterances were divided into equally sized and well balanced training and test sets. The vocabulary was composed of the Finnish digits (*yksi, kaksi, kolme, neljä, viisi, kuusi, seitsemän, kahdeksan, yhdeksän* and *nolla*).

The test set comprised 3686 six-digit strings. The algorithm was tested in clean conditions, and in noise with three SNR values (10, 5 and 0 dB). The noisy

Table 1. Error statistics for the test set (3686 6-digit strings). The results are first shown when the IEC has been switched off. Insertion errors at the beginning, at the middle and at the end of the string are listed separately.

Error type	Clean	Clean/IEC	5 dB	5dB/IEC
Deletions	45	49	150	163
Subst.	37	37	140	139
Insertions	627	59	113	79
Beg-Ins.	398	28	52	27
Mid-Ins.	50	27	45	45
End-Ins.	179	4	16	7

waveforms were obtained from the clean utterances by mixing car noise. The HMMs were estimated using duration constrained maximum likelihood training [7].

Figure 3 shows the SNR difference histograms for the test set in two environments. In the figure, the label *detected insertion* refers to the cases when an insertion has been detected by the IEC algorithm, and the label *other* refers to the cases when no insertions were detected. It can be seen that in clean conditions the SNR difference histograms of the detected insertions and others are bimodal and an optimum threshold can be found. In noise the SNR difference histogram for the detected insertions is shifted to the left and the optimum threshold is smaller than in clean. Therefore, at higher SNRs, the insertion errors can be well corrected with the method, but the correction becomes more difficult as the SNR decreases.

The error statistics for the test set are illustrated in Table 1. For reference, the statistics are also shown for the case when the IEC algorithm was switched off. It is evident that the IEC algorithm was able to remove most

Table 2. String recognition rates using the test set. The baseline versus normalized feature vectors and normalized feature vectors with gain constraint (GC).

SNR	Baseline (%)	Norm(%)	Norm+GC(%)
Clean	97.15	88.88	92.00
10 dB	95.79	93.60	95.80
5 dB	91.96	90.37	92.84
0 dB	65.36	75.61	77.62
Mean	87.57	87.12	89.57

Table 3. String recognition rates using the test set for the normalized feature vectors (no IEC) vs. normalized feature vectors with the IEC (NIEC) and the gain constraint (GC).

SNR	Baseline (%)	NIEC (%)	NIEC+ GC(%)
Clean	97.15	96.88	96.64
10 dB	95.79	95.39	95.39
5 dB	91.96	91.48	93.22
0 dB	65.36	79.41	80.96
Mean	87.57	90.79	91.56

of the insertions in the beginning and in the end of the digit strings with a marginal number of deletions of correct digits.

Table 2 compares the string recognition rates of the system using the baseline (un-normalized) feature vectors against the one with normalized feature vectors. Also the case when the gain constraint was added is shown. When the SNR was decreased the baseline system stayed ahead until the 0 dB case, where the normalization scheme excelled. The addition of the gain constraint improved the rates further, especially in the clean environment as was expected. However, it is clearly visible in the results obtained in clean conditions that the normalization method could not be balanced to produce good results in noise without producing insertion errors in clean.

A comparison between the baseline with the system using normalized feature vectors together with the IEC algorithm is shown in Table 3. Also the results with the gain constraint are listed in the table. The combination of the IEC algorithm and the gain constraint achieved an error rate reduction of more than 32% over the baseline system. The improvement in noisy environments can be explained by the fact that as we are now able to cope with an increase in the amount of insertion errors in clean conditions, the system can be tuned to produce less deletions in noise. With the basic normalization scheme, this would have resulted in very bad performance in clean. When the new approach is compared with this basic scheme, the relative improvement is 70% in clean conditions and over 20% at 0 dB SNR.

## 6. CONCLUSIONS

We presented an insertion error correction algorithm for digit string recognition. The algorithm together with the proposed normalization gain constraint makes the feature vector normalization method realize its potential better. The good performance in noisy conditions is complemented with a good performance in clean conditions since the excessive number of insertion errors is reduced. Our recognition test results showed an average 32% improvement against the baseline.

## 7. REFERENCES

- [1] O. Viikki, D. Bye, K. Laurila, "A recursive feature vector normalization approach for robust speech recognition in noise," *IEEE International Conference on Acoustics, Speech, and Signal Processing*. Seattle, WA, USA, May 1998, pages 733-736.
- [2] B.-H. Juang, W. Chou, C.-H. Lee, "Minimum classification error rate methods for speech recognition", *IEEE Transactions on Speech and Audio Processing*, vol. 5, no. 3, May 1997, pages 257-265.
- [3] I. Zeljković, S. Narayanan, A. Potamios, "Unsupervised HMM adaptation based on speech-silence

discrimination,” *Proceedings of Eurospeech Conference*. Rhodes, Greece, Sep 1997, pages 2055-2058.

- [4] M. G. Rahim, C.-H. Lee, B.-H. Juang, “Robust utterance verification for connected digits recognition,” *IEEE International Conference on Acoustics, Speech, and Signal Processing*. Detroit, Michigan, USA, May 1995, pages 285-288.
- [5] R. A. Sukkar, A. R. Setlur, C.-H. Lee, J. Jacob, “Verifying and correcting recognition string hypotheses using discriminative utterance verification,” *Speech Communication*, vol. 22, 1997, pages 333-342.
- [6] K. Laurila, M. Vasilache, O. Viikki, “A combination of discriminative and maximum likelihood techniques for noise robust speech recognition”, *IEEE International Conference on Acoustics, Speech, and Signal Processing*. Seattle, WA, USA, May 1998, pages 85-88.
- [7] K. Laurila, “Noise Robust Speech Recognition with State Durations Constraints”, *IEEE International Conference on Acoustics, Speech, and Signal Processing*. Munich, Germany, 1997, pages 871-874.