Look-a-head Sequential Feature Vector Normalization for Noisy Speech Recognition

Rathinavelu Chengalvanayan
Speech Processing Group,
Lucent Speech Solutions Department
Lucent Technologies, Naperville, IL 60566, USA
Email: rathi@lucent.com

ABSTRACT

Cepstral mean subtraction (CMS), which is a simple long-term bias removal, is used to compensate for transmission and linear fixed channel effects. In order to process the non-linear channel, a two-level CMS was proposed where separate channel compensation is performed for segments that are classified as speech and for segments classified as background. In this paper, methods for extending the two-level CMS to real-time implementation is proposed using a finite number of look-a-head frame delay, which further reduces computation and memory requirements of the compensation process. The on-line bias compensation shows similar characteristic curve as that of batch-mode and has the effect of greatly reducing the sensitivity of the recognizer to transmission noise variability.

1. INTRODUCTION

When speech recognizers are deployed in telephone services, they often encounter variable transmission and background noise conditions, which significantly deteriorate their performance level [5, 12]. To account for the variability due to transmission and noise, we consider batch and sequential non-linear cepstral mean subtraction (CMS) techniques. CMS is a standard channel compensation technique which can remove the time-invariant part of channel distortion [8, 1, 11]. The effectiveness of CMS is severely limited when the environment can’t be adequately modeled by a linear channel [12]. In order to process the non-linear channel, the two-level CMS method is proposed and further the system performance depends on the signal classification accuracy [2, 7, 9].

Bias is usually estimated in a batch-mode, assuming that its parameters are constant for the whole utterance [10]. In many practical systems, a speech utterance is commonly analyzed on a frame-by-frame basis (i.e. frame synchronously), and acoustic features are passed to the recognizer at every frame. This process of dealing with each frame individually is rather crucial in order to ensure real-time implementation and minimal memory requirements[1]. It implies that not all frames corresponding to the test utterance are simultaneously available for the computation of the cepstral mean. In this paper, methods for extending the two-level CMS to real-time implementation is proposed using a finite number of frame look-a-head delay [6]. The proposed sequential two-level cepstral normalization achieves the same level of accuracy as batch estimation with potentially improved convergence and with smallest implementation costs.

2. BATCH-MODE TWO-LEVEL CMS

CMS based upon the time-invariant linear channel model has been used to equalize the channel difference between training and test data. The noisy speech is

\[ Y(w) = [X(w) + N_1(w)] H(w) + N_2(w) \]  

(1)

X(w) refers to the input speech component. N1(w) refers to the environmental noise, while channel H(w) refers to the microphone distortion. The noise N2(w) is a relatively small component due to electric circuitry and is ignored. For clean training and test conditions N1(w) is negligible, the distortion becomes additive in the cepstral domain. Therefore CMS removes the time-invariant part of channel distortion. When high level of additive noise (N1(w) being non-negligible) is present, we can only ignore noise component in speech segments with high signal-to-noise ratio and use CMS in those segments. In the same manner, we can ignore the speech component in segments of very low signal-to-noise ratio (very low level speech or no speech is present).

Therefore, recognition performance can be improved further by using a two-level CMS, where separate channel compensation is performed for segments that are classified as speech and for segments classified as background [2, 7, 9]. The two-level or non-linear CMS technique is implemented in several steps:

- Determine the maximum frame energy \( E_{\text{max}} \) and minimum frame energy \( E_{\text{min}} \) for every utterance.
- Separating the frames of current utterance into two classes: if \( E_t < \alpha \times E_{\text{max}} + (1 - \alpha)E_{\text{min}} \), then the frame \( t \) belongs to class-I (silence class), else to class-II (speech class), where \( \alpha \) is a constant determined by a fast experiment. In all the following experiments, we tend to choose the same \( \alpha \) value (\( \alpha \) is set to 0.3 in our current study), for every database.
- The background and the speech cepstral mean vectors are calculated for the whole utterance.
- Finally the normalized cepstral features for each frame are computed by subtracting them by their respective cepstral means.

The above procedure is applied in both training and recognition [5].

3. SEQUENTIAL TWO-LEVEL CMS

Basically, the batch-mode technique suffers from the fact that we need some amount of speech to get a first estimate of the long-term cepstrum which makes the approach off-line [1]. To overcome this problem, we have implemented
the on-line cepstral normalization as follows. We assume a
time-delay of seven frames for computing the first and sec-
tond order derivatives of feature vectors as well as a 20 frame
delay during real-time energy normalization [6]. Therefore,
multi-level CMs can be applied by taking full advantage
of the availability of the whole utterance during training,
while only imposing a finite time delay of 20 frames during
recognition. A flow chart of the on-line two-level CMS is
illustrated in Figure 1. An initial silence/background and
speech cepstral mean vectors (Y_{int} and Z_{int}) are gener-
ated by averaging all cepstral vectors of the training data that
belong to silence and speech classes respectively. The in-
stantaneous \( E_{max} \) and \( E_{min} \) are estimated according to [6]
so that they will be subsequently used in the silence and
speech classification stage. The resultant vectors are used
for bootstrapping the two-level CMS at the start of each ut-
terance during recognition. The initial cepstral mean vec-
tors (\( Y_{int} \) and \( Z_{int} \)) are updated by looking into the first 20
frames and thereafter it is updated once for every frame un-
til the utterance ends. The cepstral mean vectors for silence
and speech classes are updated as follows.

\[
Y_{new} = \frac{(\gamma + t_y) Y_{old} + X_t}{\gamma + t_y + 1} \quad t_y \in \{1, 2, 3, \ldots, T\} \tag{2}
\]

\[
Z_{new} = \frac{(\gamma + t_y) Z_{old} + X_t}{\gamma + t_y + 1} \quad t_x \in \{1, 2, 3, \ldots, T\} \tag{3}
\]

where \( X_t \) is the t-th cepstra, \( Y_{new} \) and \( Z_{new} \) are the ac-
cumulated mean vectors for the silence and speech classes
up to the t-th frame and \( \alpha \) is its weighting coefficient. The
\( Y_{old} \) and \( Z_{old} \) are the previous values of the correspon-
ding cepstral means \( Y_{new} \) and \( Z_{new} \). Initially the \( Y_{old} \) and

![Figure 1. A flow diagram of on-line cepstral normalization.](image)

---

**Figure 2. Typical energy measurement contours for the utterance “9876”**. The top plot shows the original speech energy and the bottom plot shows the speech/nonspeech classification.

\( Z_{old} \) are set to \( Y_{int} \) and \( Z_{int} \) at the beginning of each ut-
terance. \( Y_{old} \) or \( Z_{old} \) is updated whenever the d-th look-
a-head frame from t-th cepstra belongs to either silence or
speech class as per equations (2) and (3), where \( t_y \) is the
silence frame counter that gets incremented whenever the
look-a-head frame is classified as silence. Similarly, \( t_x \) is
the speech frame counter that is incremented when the look-a-
head frame belongs to speech class. Note that \( \gamma \) is essen-
tially a forgetting factor which determines the significance
of the cepstral mean vector, at the t-th frame, with respect
to the t-th cepstral vector. The value of \( \gamma \) is typically set
to a finite number of frames ranging from 10 to 100. This
gives the estimated mean vector an equal weighting at ev-
ery frame. After estimating the silence and speech cepstral
mean vectors in equations (2) and (3), it is then subtracted
from each frame cepstral vector to remove the bias based
on current frame class:

\[
\hat{X}_t = \begin{cases} 
X_t - Y_{new} & \text{if } X_t \rightarrow silence \text{- class} \\
X_t - Z_{new} & \text{if } X_t \rightarrow speech \text{- class} \\
1 \leq t \leq T, 
\end{cases}
\]

where \( \hat{X}_t \) is the bias removed cepstral vector at time \( t \). This
on-line iterative process of updating the silence and speech
mean vectors and removing the bias from cepstral vector con-
tinues until the utterance is ended.

To illustrate the nature of the signal classification, Figure
2 shows the actual frame energy trajectory and the corre-
sponding speech index for the connected digit “9876” spok-
en by a female speaker. It is observed that the two-level
CMS provides better speech and silence classification and
further enhances the system performance. In order to show
the behavior of the on-line CMS and to compare it to the
classical batch-mode CMS, the subtracted values with these
techniques from the cepstral coefficients are computed. For
simplicity, we consider only the first and second cepstral co-
efficients. These values are shown in Figures 3 and 4 for the
wireless test digit-string “9876” as a function of the analysis
frame indices. In Figure 3, the value of \( \alpha \) is set to zero to
mimic the one-level CMS and in Figure 4, the plots were
created using the value of 0.3 for \( \alpha \). Both the figures were
Figure 3. Value subtracted from the a) first cepstral coefficient and b) second cepstral coefficient using the batch-mode one-level CMS (flat-line) and online one-level CMS (dotted-line).

Figure 4. Value subtracted from the a) first cepstral coefficient and b) second cepstral coefficient using the batch-mode two-level CMS (flat-line) and online two-level CMS (dotted-line).

Figure 5. Block diagram of feature analysis using batch-mode (for training) and online (for testing) two-level cepstral normalization.

generated using a look-ahead delay of 50 frames. Respective cepstral values are also plotted in each sub-graphs using solid lines. It is observed that the online CMS produces after nearly 50 frames an estimate of the bias introduced by the channel effect close to the one given by the batch-mode CMS.

4. EXPERIMENTAL SETUP

Wireless data of connected digit strings containing the English digits one through nine, zero and oh was chosen to verify the effectiveness of the algorithms outlined in this paper. The wireless data are a compilation of databases collected during several independent data collection efforts, field trials, and live service deployments [4]. Wireless database contains connected digit strings recorded over analog AMPS, TDMA and digital cellular channels. The collected wireless data include different channel and noise conditions varying from clean speech to hardly audible speech, contaminated mainly by environmental car noise. The training and test set contain 15488 and 9142 digit strings with lengths range from 1 to 30.

The recognizer feature set consists of 39 features that includes the 12 filtered linear predictive cepstral coefficients, log-energies, their first and second order derivatives [1]. The energy feature is batch normalized during training and testing [6]. Following feature analysis, each feature vector is passed to the recognizer which models each word in the vocabulary by a set of left-to-right continuous mixture density HMM using context-dependent head-body-tail models [4]. Each word in the vocabulary is divided into a head, a body, and a tail segment. To model inter-word coarticulation, each word consists of one body with multiple heads and multiple tails depending on the preceding and following contexts. In this paper, we model all possible inter-word coarticulation, resulting in a total of 276 context-dependent sub-word models. Both the head and tail models are represented with 3 states, while the body models are represented with 4 states, each having multiples of 4 mixture components. Silence is modeled with a single state model having 32 mixture components. This configuration results in a total of 276 models, 837 states and approximately 3904 mixture components for wireless models [5].
Table 1. String accuracy for an unknownlength grammar-based English connected digit recognition task with batch and on-line CMS.

<table>
<thead>
<tr>
<th>Model Type</th>
<th>String Error</th>
<th>Error Rate Reduction</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline (No CMS)</td>
<td>6.91%</td>
<td>—</td>
</tr>
<tr>
<td>One-level (batch)</td>
<td>6.09%</td>
<td>12%</td>
</tr>
<tr>
<td>One-level (on-line)</td>
<td>6.13%</td>
<td>11%</td>
</tr>
<tr>
<td>Two-level (batch)</td>
<td>5.45%</td>
<td>22%</td>
</tr>
<tr>
<td>Two-level (on-line)</td>
<td>5.56%</td>
<td>20%</td>
</tr>
</tbody>
</table>

Training included updating all the parameters of the model, namely, means, variances and mixture gains using one iteration of MLE followed by six epochs of MSE to further refine the estimate of the parameters [13]. The number of competing string models was set to four and the step length was set to one during the model training phase. Each training utterance is signal conditioned by applying batch-mode 2L-CMS prior to being used in MSE training [1]. The integration of feature analysis and the signal conditioning is illustrated in Figure 5. Note that the two-level CMS was only applied on the cepstral features and the delta and delta-delta coefficients remained unaffected. In addition, the delta and delta-delta coefficients were calculated from the bias removed static cepstral features [3]. The length of the input digit strings are assumed to be unknown during both training and testing.

5. EXPERIMENTAL RESULTS

We have conducted experiments to verify the effectiveness of the proposed on-line CMS techniques using the noisy wireless database. Preliminary experiments indicated that an accurate estimation of the cepstral mean vector requires approximately 100 frames. Also setting the delay of 100 frames both in training and recognition, rather than just in recognition, does not lead to a better performance on all the databases [1]. For on-line CMS experiments we set the value of $\gamma$ to be 100 and the look-ahead delay $d$ of 20 frames only during recognition. This ensured a slower forgetting rate of the bootstrapped mean vector or a gradual increase in the cepstral mean weighting with increasing the number of frames. Further, we applied off-line CMS taking full advantage of the availability of the whole utterance during training.

The table presents the string error rate with multi-level CMS tested using both the off-line and on-line mode. We observed a 12% and 22% string error rate reductions when moving from baseline to one-level and from one-level to two-level batch CMS. The non-linear CMS technique outperforms benchmark and one-level CMS and exhibits consistent improvements even for short utterances. Compared to the off-line approach, little or no degradation is observed with the on-line CMS algorithm as illustrated in Table 1. This shows that the behaviour exhibited in Figures 3 and 4 is a dominant one, testifying our conjecture that the on-line CMS fits well with implementation for on the fly adaptation where frames are processed as soon as received.

6. CONCLUSIONS

In order to process the non-linear channel in sequential manner, a method for extending the two-level CMS to real-time implementation is proposed where separate channel compensation is performed for segments that are classified as speech and for segments classified as background. The proposed algorithm is implemented using a finite number of look-ahead frame delay, which further reduces computation and memory requirements of the compensation process with smallest implementation costs. Moreover, the on-line CMS employing a bootstrapped mean is particularly attractive since the results are competitive to those obtained when computing the cepstral mean over the whole utterance interval. Also in speaker-independent connected digit recognition, over 10% and 20% string error rate reductions are obtained with the proposed sequential two-level feature normalization in comparison with one-level and zero-level sequential CMS systems respectively.

REFERENCES