THE USE OF CEPSTRAL MEANS IN CONVERSATIONAL SPEECH RECOGNITION

Martin Westphal

Interactive Systems Laboratories
University of Karlsruhe — 76128 Karlsruhe, Germany
westphal@ira.uka.de

ABSTRACT

Environmental robustness and speaker independence are important issues of current speech recognition research. Channel and speaker adaptation methods do the best job when the adaption is done towards a normalized acoustic model. Normalization methods might make use of the model but primarily influence the signal such that important information is kept and unwanted distortions are cancelled out. Most large vocabulary conversational speech recognition systems use Cepstral Mean Subtraction (CMS), a channel normalization approach to compensate for the acoustic channel (and also the speaker). In this paper we discuss the basic algorithm and variations of it in the context of conversational speech and report our experience using different approaches on two widely used conversational speech recognition tasks.

1. INTRODUCTION

As speech recognition on clean, read speech has become better, recent research efforts concentrate more on conversational (spontaneous) speech as well as on channel and microphone robustness. An example for conversational speech is the Spontaneous Scheduling Task (SST) that was collected for different languages like English, German, Japanese, Spanish and Korean. Switchboard and Callhome are even more spontaneous speech tasks recorded over local and international telephone lines.

Due to variations introduced by different channels, microphones and speakers, state-of-the-art recognizers use normalization and adaptation methods to compensate for these distortions. Although there is a variety of adaptation algorithms, most systems use the simple but effective Cepstral Mean Subtraction (CMS) for channel normalization. While adaptation methods require an acoustic model that has to be trained and is not necessarily normalized, algorithms like CMS and RASTA are purely signal based and try to eliminate disturbing channel and speaker effects before the signal is used to train a recognizer.

They only make use of channel and speech production model assumptions.

Despite its simplicity it was proven many times that CMS is very effective. We will show that there are some concerns when it is used for conversational speech and propose some variations to the basic algorithm. They will be discussed on a unified theoretical background and completed with some case studies and experimental results. All results, if not mentioned otherwise, are obtained with the Janus Speech Recognition Engine (Janus-III) on the German SST (Verbmbil evaluation test set 96) or the Switchboard/Callhome task (a 418 utterances test set). Janus-III was among the best systems in last year’s Switchboard evaluation and had the best result in the last Verbmbil evaluation using new algorithms like VTLN and MLLR. For more details see [6][8].

2. THE BASIC PRINCIPLE

When a speech signal passes a linear time invariant channel, this convolutional distortion becomes multiplicative in the spectral domain and additive in the log-spectral domain. Since the cepstrum is just a linear transformation of the log-spectrum both can be treated equally in this context. For speech recognition, a short time analysis is performed, resulting in the speech spectrum $S_t(\omega)$ and the measured spectrum $Y_t(\omega)$. The time index $t$ indicates the time dependence.

\[
\begin{align*}
S_t(\omega) &= C(\omega) \cdot X_t(\omega) \\
\log-\text{spec. or cepstrum: } y_t &= c + s_t
\end{align*}
\]

The assumption of a constant channel $C(\omega)$ allows to compensate for it by subtracting the mean, leading to a cepstral mean subtracted feature $z_t$:

\[
z_t = y_t - \bar{y} = c + s_t - (c + \bar{y}) = s_t - \bar{y}
\]

Here we see that also a speech mean $\bar{y}$ is subtracted. When we divide the speech spectrum in two parts $S_t(\omega) = V(\omega) \cdot X_t(\omega)$ with $v = \bar{y}$ and $\bar{y} = 0,$
then \( V(\omega) \) can be seen as part of the channel. While 
\( C(\omega) \) depends on the acoustic channel and recording 
environment, \( V(\omega) \) is characteristic for the current 
speaker and the uttered speech. If we had enough 
samples to build a long time statistic, \( V(\omega) \) would 
depend mainly on the speaker. This is very critical 
for conversational speech since some utterances 
contain only single words. Whenever possible, the mean 
should be estimated over all available utterances of 
a certain speaker.

For Switchboard, we found that using the speaker 
based mean instead of the utterance based mean leads to 
relative word error reductions of 4\% in 
GSST, very short utterances are rare and the perform-
ance difference is not significant. Since the aim 
here is an online application, the speaker based mean 
estimate over all the speaker’s utterances of a con-
versation is out of question anyway.

Delta coefficients not only provide the recognizer 
with context information, but are also very robust to 
channel variations:

\[
\Delta z_t = z_{t+\tau} - z_{t-\tau} = s_{t+\tau} - s_{t-\tau}
\]

The channel \( e \) was eliminated in the delta com-
putation, however possibly relevant, local static com-
ponents of the signal were lost. Similar effects occur 
when adjacent frames are used as input features to-
gether with an LDA transformation.

3. THE CHANNEL MODEL WITH NOISE 
AND THE CEPSTRAL MEAN

When we replace \( C(\omega) \cdot V(\omega) \) by the overall channel 
\( H(\omega) \) and consider additive noise \( N_t(\omega) \), we obtain the 
following well known channel model:

\[
Y_t(\omega) = H(\omega) \cdot X_t(\omega) + N_t(\omega)
\]

For the log-spectral or cepstral domain we will use 
two alternative terms for speech and non-speech

\[
y_t = \begin{cases} 
  x_t + h + r_t & \text{speech}, \\
  n_t + t_t & \text{pause},
\end{cases}
\]

with the following substitutions (here given for the 
log-spectral domain):

\[
r_t = \log(1 + \frac{N_t(\omega)}{H(\omega) \cdot X_t(\omega)}) \\
t_t = \log(1 + \frac{H(\omega)}{N_t(\omega)})
\]

For the cases of dominant speech \((N_t(\omega) \ll H(\omega) \cdot 
X_t(\omega))\), pause \((X_t(\omega) = 0)\), and silence \((X_t(\omega) = 0, 
N_t(\omega) \ll \alpha)\), we can simplify (1)

speech: \( y_t \approx x_t + h \)

pause: \( y_t = n_t \)

silence: \( y_t \approx \log(\alpha) \)

where \( \alpha \) is a small constant factor and \( N_t(\omega) \) the 
real noise in \( N_t(\omega) = N_t(\omega) + a \) such that the log’s 
argument never becomes zero.

With this model the mean \( m \) of the received sig-
nal \( y_t \) can be expressed in two terms containing the 
mean over speech and over non-speech frames weighted 
by the proportion of speech frames \( \alpha \) and pause frames 
\( \beta = 1 - \alpha \). For static noise \((n_t = n)\) we get 

\[
m = \alpha \cdot \bar{y}_s + \beta \cdot \bar{y}_p \\
= \alpha \cdot (\bar{x} + h + \bar{p} + \alpha n) + \beta \cdot n
\]

For long utterances with a high signal-to-noise-
ratio (SNR) we can neglect \( \bar{y}_s \) and \( \bar{y}_p \) and get the following approximation:

\[
m \approx \alpha \cdot h + \beta \cdot n
\]

4. VARIATIONS OF CMS

Based on the standard Cepstral Mean Subtraction 
(CMS), we will introduce three variations, discuss 
their effects on the original input \( y_t \) and present re-
results on different conversational speech tasks.

4.1. CMS

For standard CMS, i.e. taking \( z_t = y_t - m \) as new 
input feature, we substitute (2) and assume static 
opt to study the effects of CMS on the mentioned 
cases.

speech: \( z_t \approx x_t + \beta \cdot h - \alpha \cdot n \)
pause: \( z_t \approx \alpha \cdot n - \alpha \cdot h \)

Note that for segmented speech with not many pauses 
(\( \beta = \alpha \)), the compensation works well for the speech 
case although we introduced some noise dependence. 
For the pause case, we introduced a shift that is re-
akn to the channel. In conversational speech we 
have a greater variance of the pause proportion \( \beta \) 
that will reduce the desired channel compensation.

4.2. SCMS

To overcome the dependence on \( \beta \), the Speech-based 
Cepstral Mean Subtraction (SCMS) estimates the 
mean only on speech frames \((z_t = y_t - \mu_{\text{sps}})\) using

\[
m_{\text{sps}} = \frac{\sum w_t \cdot y_t}{\sum w_t}
\]

where \( w_t \) is the probability \( p(\text{speech}|y_t) \) or the output 
of a speech detector (1 for speech, 0 for pause).

For this method we get the following approxima-
ctions.

speech: \( z_t \approx x_t \)
pause: \( z_t \approx n_t - h \)
We achieved some improvements using SCMS on our GSST development test set reducing the error rate from 21.2% to 19.9%. For Switchboard and Callhome the error rates increased a bit which we think is due to the introduction of pause in the pause case and the suboptimal speech detection.

4.3. 2CMS

To solve the first problem, a 2 level Cepstral Mean Subtraction (2CMS) can be used. The input vector $z_t$ is then calculated as

$$z_t = y_t - w_t \cdot m_{spe} - (1 - w_t) \cdot m_{pau}$$

In [4], 7% to 20% relative improvements of the error rate for digit recognition in a car environment were reported using 2CMS with an energy based speech detector. On the other hand, [5] reported a 3.5% to 6.5% relative increase of the error rate when using 2CMS instead of SCMS. When we take a look at the approximations for 2CMS

speech: $z_t \approx x_t$
pause: $z_t \approx x_t - m_{pau}$

we see that $h$ was eliminated. However we produce large errors if speech frames are detected as pause and vice versa. Even if we don’t, speech and pause vectors have a zero mean with the result that speech was shifted towards the pause region in feature space ($z_{spe} \approx z_{pau} \approx 0$) making it harder for the recognizer to distinguish them. For Switchboard and even more on our Callhome test set, we indeed observed an increase of the error rate of 1% to 3.5% compared to standard CMS.

4.4. 2CDMS

Due to these problems with 2CMS we propose to use a 2 level Cepstral Delta Mean Subtraction (2CDMS) or 2 level Cepstral Mean Normalization (2CMN) according to

$$z_t = y_t - w_t (m_{spe} - m_{pau}) - (1 - w_t) (m_{pau} - m_{pau})$$

and a continuous estimate of the weighting factor $w_t$. Here $m_{pau}$ is similar calculated as $m_{spe}$ in (3). $m_{spe}$ and $m_{pau}$ are their averages over the whole data base. The advantage of 2CDMS is that the input $y_t$ is only corrected by a linear combination of the two delta means, compensating channel and noise effects but leaving much of the original structure allowing to distinguish between speech and pause frames.

5. SPEECH DETECTION

For some of the CMS variations a speech detector is required. For SCMS we simply used a threshold for the smoothed signal power to indicate whether a frame $y_t$ is counted as speech ($w_t = 1$) or as non-speech ($w_t = 0$). We refer this method as discrete SCMS.

For a continuous estimation of $w_t = p(speech|y_t)$ we trained an acoustic model with Gaussian mixtures for speech and silence based on the alignment of our recognizer. The probability estimation could then be calculated for each frame as:

$$p(speech|y_t) = \frac{p(speech) \cdot p(y_t|speech)}{p(speech) \cdot p(y_t|speech) + p(sil) \cdot p(y_t|sil)}$$

The a priori values were estimated on the training data, $p(speech)$ was between 0.8 and 0.9. Figure 1 shows $p(speech|y_t)$ for a Switchboard (SBW) segment. Since this measure is frame based (using $y_t$, not $y_1 ... y_T$) or uses only a small context we also tried a smoothed version.

![Figure 1: audio signal and output of speech detector](image)

6. TASKS AND RESULTS

Experiments were performed using two conversational speech recognition tasks: The English Switchboard/Callhome (SWB/CH) task with conversations recorded over national and international telephone lines and the German Spontaneous Scheduling Task (GSST) that was recorded in an office environment with a Sennheiser headset.

To train the systems with the different normalization methods, we used fixed, speaker adapted viterbi alignments precalculated with our standard system. The training steps included LDA calculation, k-means initialization of the Gaussian mixtures and 4 training iterations. Warp factors for VTLN and the polypHONE cluster tree were taken from the standard system. For both tasks we used mel frequency cepstral coefficients (MFCC) together with their first and second order delta coefficients.

6.1. Switchboard/Callhome

For the experiments on SWB/CH, we used a predecessor of the CMU 1997-DARPA-evaluation sys-
system setup with 25000 mixtures over 5000 codebooks and used 1224 SWB conversation sides for training. The vocabulary size was about 15000 words and the tests were evaluated on a 418 utterance set of SWB and CH data with a total of 4275 words. Table 1 shows the word error rates for the CMS variations on the subsets and the whole test set. On SWB test data, we achieved improvements by estimating the mean over the whole conversation side (speaker-based CMS). Using continuous SCMS or 2CDMS helped for both subsets and the mixed data (BOTH).

<table>
<thead>
<tr>
<th>Method</th>
<th>Word Error in %</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>SWB</td>
</tr>
<tr>
<td>utterance based CMS</td>
<td>39.8</td>
</tr>
<tr>
<td>speaker based CMS</td>
<td>38.3</td>
</tr>
<tr>
<td>discrete SCMS</td>
<td>39.0</td>
</tr>
<tr>
<td>continuous SCMS</td>
<td>37.7</td>
</tr>
<tr>
<td>continuous 2CMS</td>
<td>39.4</td>
</tr>
<tr>
<td>continuous 2CDMS</td>
<td>37.6</td>
</tr>
<tr>
<td>smoothed cont. SCMS</td>
<td>38.3</td>
</tr>
</tbody>
</table>

Table 1: Word error rates for SWB/CALLHOME

6.2. GSST

The GSST system is similar to our 1996-Verb mobil-evaluation system setup [3] with 10000 mixtures over 2500 codebooks trained with nearly 140000 utterances. The vocabulary consists of 5800 words. The first column of table 2 shows error rates for the official 1996 Verbmobil evaluation test set with 343 utterances and 6442 words. This test set was recorded with the same setup (Sennheiser headset) as the training data. None of the variations helped for this matched conditions although we got improvements with SCMS for an internal development set.

In a recent work [2] we recorded a set of GSST conversations simultaneously with 5 different microphones. For a 2151 word test set recorded over a room microphone (column 2), placed on the table between the two speakers, the performance dropped down dramatically. Here the SCMS and the 2CDMS got better results than the standard CMS.

For a similar test set (column 3), the error rate decreased down to 36.9 % [2] by using Codebook Dependent Cepstral Normalization (CDCN) [1]. Note that the same test set recorded over a Sennheiser microphone tested with standard CMS had an error rate of 26% and is thus much more difficult than the evalset 96. The CDCN considers also additive noise in its model assumption (we have a lower SNR for the room microphone) but is computationally much more costly than the simple Mean Subtraction algorithms and requires a clean speech codebook. As for the 2-level Mean Subtraction, the compensation vector is frame dependent. For SWB we could not decrease the word error by using CDCN.

<table>
<thead>
<tr>
<th>Method</th>
<th>GSST Word Error in %</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>evalset 96</td>
</tr>
<tr>
<td>standard CMS</td>
<td>15.1</td>
</tr>
<tr>
<td>discrete SCMS</td>
<td>15.2</td>
</tr>
<tr>
<td>continuous SCMS</td>
<td>15.7</td>
</tr>
<tr>
<td>continuous 2CMS</td>
<td>15.1</td>
</tr>
<tr>
<td>continuous 2CDMS</td>
<td>15.6</td>
</tr>
<tr>
<td>CDCN</td>
<td>-</td>
</tr>
</tbody>
</table>

Table 2: Word error rates for GSST

7. CONCLUSION

We discussed some variations of the widely used Central Mean Subtraction method and presented results for two conversational speech recognition tasks. The performance depends on the channel variation within the data base (different telephone channels for SWB/CH compared to fixed setup for GSST) and whether we have a test environment matching the training condition or not. For an unmatched condition, the word error rate decreased by up to 13% on GSST data recorded with a room microphone using the Speech-based CMS.

8. REFERENCES