PARAMETRIC CEPSTRAL MEAN NORMALIZATION FOR ROBUST SPEECH RECOGNITION

Ozlem Kalinli[1], *Gautam Bhattacharya*[2*], *Chao Weng*[3†]

[1]Apple Inc., [2]McGill University, [3]Tencent AI Lab.

ABSTRACT

This paper proposes a new channel normalization algorithm called parametric cepstral mean normalization (PCMN) to increase robustness of speech recognition to varying acoustic conditions. Rather than using a simple average of input speech features as channel estimate, as done in the traditional CMN, PCMN weighs the running average of input speech frames in a frequency dependent manner. These weights are jointly optimized together with parameters of the acoustic model training. Experimental results show that, in contrast to traditional CMN, which degrades performance on clean data, PCMN provides 5% relative improvement on clean data, while also providing 11.2% relative improvement on far-field test data. We also propose an adaptive version of PCMN, called aPCMN, where both input speech features and channel estimates have weights. These weights are computed at run time and they change dynamically based on the input speech. aPCMN provides 13.0% relative improvement on far-field test set, while still maintaining 5% relative improvement on clean data.

Index Terms— Robust automatic speech recognition, cepstral mean normalization, channel normalization.

1. INTRODUCTION

Automatic speech recognition (ASR) has advanced significantly in recent years thanks to advancements in deep learning based modeling [1, 2, 3]. Indeed, ASR technology has reached a stage where challenging applications have become a reality and has become available to millions of users in the form of personal assistants in smart phones and voice-controlled home devices. Even though ASR systems work reasonably well in clean, close-talking conditions, its performance degrades severely for noisy far-field settings. Far field ASR is known to be a challenging problem due to enviromental factors such as background noise, acoustic reverberation, and acoustic echo, and remains as an active area of research in the speech community [4, 5, 6].

A common strategy for tackling far-field ASR is to incorporate a front-end digital signal processing (DSP) block before speech signal is fed to the acoustic model. The DSP block attempts to clean the noisy far-field data by using techniques such as echo cancellation, de-reverberation, noise suppression, and beam forming etc. While the DSP block enhances far-field speech and improves ASR performance substantially on such data, this performance is still significantly worse than its counterpart in clean or near-field condition. Part of the reason for this is that the DSP block is not perfect and enhanced data still contains some residual noise and artifacts. Cepstral mean normalization (CMN) and spectral subtraction, which are traditional methods used to reduce noise and improve robustness, can also be used here to further reduce the residual noise and improve ASR performance.

Recently, there have been quite a few studies that focused on joint optimization of some of the front-end processing stages with acoustic model training. [7] proposed joint multi-microphone enhancement and acoustic modeling, where beamforming filter coefficients are predicted by neural network layers which are jointly trained with a CLDNN acoustic model, and achieved significant WER reduction. In [8], a front-end called per-channel energy normalization (PCEN) is proposed to improve robustness to loudness variation, and it outperformed log-mel features on keyword spotting task. In PCEN, parameters are jointly optimized with the acoustic model. To address robsutness against interfering speech, [9] used an encoder network projecting an anchor word onto a fixed size embedding, which is appended to acoustic feature vector. The jointly trained encoder network and acoustic model provided significant WER improvements over casual spectral subtraction and hand-crafted anchored mean subtraction methods in the presence of background speech. Our work is inspired by the successes of the aforementioned work, whose common point was joint optimization of the front-end processing with the acoustic model training.

In this work, we focus on the CMN step in far-field ASR pipeline. We propose a novel approach called parametric cepstral mean normalization (PCMN) that learns a channel normalization factor jointly with the acoustic model training. PCMN takes a sliding window cepstral mean estimate and weighs it in a frequency dependent manner. To keep the model generic and handle dynamic changes in input speech, we also introduce weights for the input cepstral features. These weights in PCMN are learned jointly with the rest of the acoustic model parameters to optimize ASR loss function. We also propose an improved version of PCMN, called adaptive PCMN (aPCMN), where weights are time dependent. In other words, these weights are computed at runtime for each input frame and changes dynamically based on the input signal. We perform extensive experiments on a variety of data sets to validate the effectiveness of PCMN and aPCMN, and show that they are able to improve over traditional CMN on a range of challenging far-field datasets, while at the same time not causing any degradation on clean data.

The remainder of this paper is structured as follows. PCMN and aPCMN are described in Section 2 and 3, which are followed by experiments and their results in Section 4. A conclusion is presented in Section 5.

2. PARAMETRIC CEPSTRAL MEAN NORMALIZATION

Environment factors such as convolutive channel and additive noise cause a shift in mean and variance of the input cepstral features [10].

*Part of this work was done while Gautam Bhattacharya was interning with Siri team at Apple.

†Part of this work was done while Chao Weng was working in Siri team at Apple.

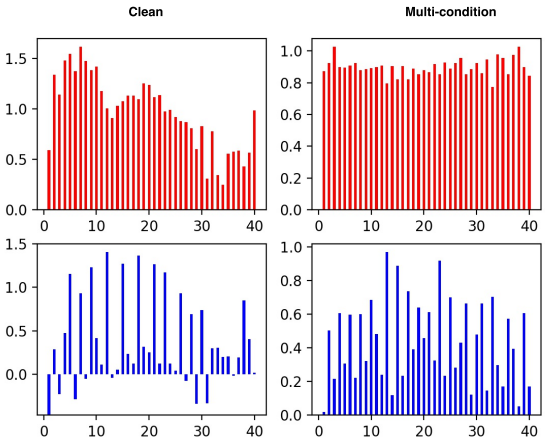

Fig. 1: Comparison of learned β (red) and α (blue) weights on different datasets. x-axis shows index i, which is $i = 1, \ldots, 40$, and y-axis shows learned values.

Cepstral mean normalization is a widely used technique to compensate the shift in mean caused by such environmental factors. One can estimate the mean over an entire utterance, however this will cause an unacceptable delay making it unfeasible for real time recognition applications. Alternatively, the cepstral mean, $\boldsymbol{\mu}_n$, can be computed over a sliding window of N frames as follows:

$$\mu_n[i] = \frac{1}{N} \sum_{m=n-N}^{n} X_m[i] \tag{1}$$

where $X_n[i]$ is the i^{th} component of the cepstral feature vector at time frame index n. Then, ceptral features are normalized as follows:

$$\hat{X}_n[i] = X_n[i] - \mu_n[i] \tag{2}$$

CMN is essentially a blind estimate; in a sense that it is not learned and it doesn't have any interaction with the acoustic modeling loss function. In this work, our goal is to learn a channel normalization factor jointly with the acoustic model. One way to do this is to learn a channel embedding using a neural network and use it as an additional input to the acoustic model as done in [9]. Here, we develop an approach that integrates the standard CMN channel estimate into acoustic model training. This is achieved by replacing the conventional CMN estimate $\boldsymbol{\mu}_n$, with a scaled and shifted version $\boldsymbol{\alpha}\boldsymbol{\mu}_n + \boldsymbol{\mu}_0$. The intuition behind this choice is that the CMN estimate may potentially be noisy for certain frequency components. Also, having $\boldsymbol{\mu}_0$ is inspired by a warm start sliding window CMN, where a cepstral mean computed on the training/dev set is used for initialization. However, here all parameters including $\boldsymbol{\mu}_0$ are learned jointly as part of acoustic model training. We also found it useful to scale the input cepstral features, $\mathbf{X}$, to the acoustic model as explained later in Section 3. Hence, the proposed PCMN model takes the following generic form:

$$\hat{X}_n[i] = \beta[i].X_n[i] - \{\alpha[i].\mu_n[i] + \mu_0[i]\} \tag{3}$$

Note that parameters $\boldsymbol{\beta}, \boldsymbol{\alpha}$, and $\boldsymbol{\mu}_\mathbf{o}$ are all frequency dependent and they are learned jointly as part of acoustic model training via back-propagation using the acoustic modeling loss function.

In Figure 1, we present an analysis of the learned $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ weights for clean training (150 hours of data) and multi-style training (450 hours of data). We use 40 log Mel-filterbank features in our experiments, hence i varies from 1 to 40 in Eq. 1-3. The top row of Fig. 1 shows the learned $\boldsymbol{\beta}$ parameters for two training set-up. We see that as the amount of training data increases, $\boldsymbol{\beta}$ weights seem to converge to 1.0. The second row of Fig. 1 shows the learned $\boldsymbol{\alpha}$ parameters. We can see that only a few $\alpha[i]$ values are close to 1.0, while most bins are scaled by smaller weights, which supports our hypothesis that cepstral mean should be weighted in a frequency dependent manner. Also, even though there was no constraint on the values of parameters during training, with the larger set-up of 450 multi-style training, learned $\boldsymbol{\alpha}$ parameters converged to $0 \leqslant \alpha[i] \leqslant 1$ for all $i = 1, ..., 40$.

These findings led us to consider a simplified version of PCMN model where we set $\boldsymbol{\beta} = \mathbf{1}$ and $\boldsymbol{\mu}_\mathbf{o} = \mathbf{0}$, and only learn $\boldsymbol{\alpha}$. From our experiments we found that the performance of the simplified model was comparable to that of the full model. In this simplified model, if we further assume that $\boldsymbol{\alpha}$ is replaced with a scalar α such that $0 \leqslant \alpha \leqslant 1$, then the model essentially acts as a gate deciding either to subtract the cesptral mean or not. In other words, when $\alpha = 0$, the method is equivalent to no-CMN, and when $\alpha = 1$ it is equivalent to traditional CMN; hence PCMN becomes an interpolation of no-CMN and CMN. In the rest of the paper, the complete set of PCMN parameters, $\boldsymbol{\alpha}$, $\boldsymbol{\beta}$, and $\boldsymbol{\mu}_\mathbf{o}$, is learned as part of training while reporting experimental results with PCMN.

3. ADAPTIVE PCMN

Trainable parameters of the PCMN layer are fixed after training and do not change for different input signals. This feature is interesting because the learned global set of parameters from the training data can improve CMN-based channel estimation. However, instead of learning a set of global parameters, we could also formulate the model so that parameters adapt dynamically to the input speech to better handle dynamic channel and/or noise changes. Hence, we propose adaptive PCMN (aPCMN) method, where parameters $\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\mu}_\mathbf{o}$ are time dependent, and they are conditioned on the input signal. This dependency is achieved by using a linear projection layer as follows:

$$[\boldsymbol{\beta}_n, \boldsymbol{\alpha}_n, \boldsymbol{\mu}_{n,0}] = \boldsymbol{W} \cdot \boldsymbol{Y}_n + \boldsymbol{b}, \tag{4}$$

where $\boldsymbol{Y}_n = [\boldsymbol{X}_{n-10}, \boldsymbol{X}_{n-9}, ..., \boldsymbol{X}_n, \boldsymbol{X}_{n+1}, ..., \boldsymbol{X}_{n+10}]$ spliced cepstral features. $\boldsymbol{W}$ and $\boldsymbol{b}$ are learned jointly with the rest of the acoustic model parameters. One trick we use during training is a similar strategy to conditional batch-normalization proposed in [11], where instead of directly outputting $\boldsymbol{\beta}_n$ we output $\hat{\boldsymbol{\beta}}_n = \mathbf{1} + \boldsymbol{\beta}_n$ to avoid potentially zeroing out the input cepstral features.

With aPCMN, our goal is to better handle dynamic changes that may happen in channel and/or users' speech characteristics. One particular case we would like to tackle is ducking, which is a feature used with home speaker devices. While the device is playing back audio, once it is triggered using a trigger phrase, the playback volume is ducked, in other words lowered drastically. This ducking causes dynamic changes both in users' speech and also in the DSP residual noise. For example, we observed that when the far-field device is playing music, usually users tend to start speaking to the device loudly, as soon as the device ducks, then users adjust and lower their voice. Similarly, ducking makes the DSP residual noise dynamically varying. aPCMN can handle these kind of dynamic changes that may happen in the channel and in users' speech characteristics by adjusting $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ for each time frame.

Figure 2 shows the evolution of $\alpha[i]$ weights for $i = 6, 18, 32$ frequency bins over the duration of a noisy recording. These plots

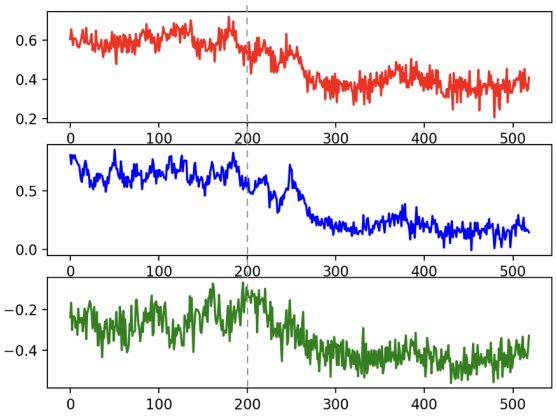

Fig. 2: Evolution of α weights over time for different frequency bins out of 40 bins. Red = bin 6, blue = bin 18, green = bin 32. X-axis shows time in terms of frame number.

.

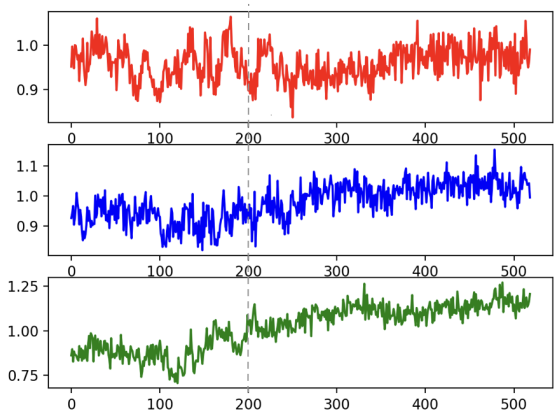

Fig. 3: Evolution of β weights over time for different frequency bins out of 40 bins. Red = bin 6, blue = bin 18, green = bin 32. X-axis shows time in terms of frame index.

change from utterance to utterance since the model is adaptive . For this recording, there was music playing back from the far-field home speaker, and the device was triggered by the user around 200^{th} frame, hence the music playback level was drastically reduced after that. As seen in the graph, the model is adapting to this change by lowering α weights over time, basically reducing the effect of cepstral mean being subtracted. Similarly, the evolution of $\beta[i]$ is shown in Fig. 3 for $i = 6, 18, 32$ frequency bins. The β weights are behaving in an opposite way; β increases after ducking to compensate for the lowered volume of the users voice and this is more prominent with the green plot, which is the 32^{nd} bin. Also, we see that the scaling for a given frequency changes significantly over the course of an utterance, showing that the adaptive model is able to adapt to changing channel and user speech characteristics.

4. EXPERIMENTAL SETUP

Initial experiments were conducted using an in-house multi-style dataset consisting of 450 hours of speech. Later results obtained using 6,000 hours of multi-style training data, and 9,000 hours of real far field home-speaker data are also presented. For the multi-style training, training data consisted of three subsets, including $1/3^{rd}$ relatively clean speech. We made one copy of this set by convolving the data with 3500 room impulse responses (RIRs) collected from a variety of real rooms to simulate reverberated speech, and another copy by adding echo-residual to reverberated speech. The echo residual contains the residual noise that remains after the DSP block when the device is playing back varying audio such as music and podcast.

For the sliding-window mean estimation, we estimate the mean using up to 600 past frames (e.g. 6 sec) with a minimum window of 100 frames for the first 100 frames. 40 dimensional log mel-filterbank features used in experiments, which are extracted every 10ms with a 25ms window. The following four systems were built: (a) Baseline DNN based acoustic model without any channel normalization, (b) with sliding-window CMN normalization (c) with PCMN, and (d) with adaptive-PCMN (aPCMN). All fully connected DNNs had 6 layers, sigmoid activation, and 1024 hidden units per layer. The size of the output layer was 8419. Inputs to DNN were whitened with the corresponding global mean and variances and spliced with +/- 10 frames of context; hence DNN input layer takes 840 dimensional feature vector. All models were trained using the cross-entropy loss function.

We evaluate our models on a relatively clean test set, this data consists of real-world data, and do contain some amount of acoustic and/or label noise. We also test our models on a number of far-field test sets. The dataset named *reverb* was simulated by convolving the clean test set with real RIRs. Care was taken to ensure that there is no overlap between the RIRs used in our training set and test one. We also used a simulated test set, called *duck*, where audio ducking takes place. First, far-field speech and echo data was recorded separately using a home-speaker device in varying room settings. Then speech with echo was created by adding recorded far-field speech to the recorded echo signal which was ducked once the triggering phrase was detected in speech signal. Music and podcast audio was used in echo-only recordings. After the mixture was created, it was passed through the DSP block for echo cancellation, de-reverberation, noise suppression, and beamforming, to enhance speech signal. Finally, two real-world far-field data sets, called *ffield* and *ffield-2*, were used for experimenting. Note that the test-sets referred as clean and reverb haven't been processed through the DSP block.

We present our experimental results using 450 hrs of multi-condition training data in Table 1. As expected, while traditional CMN improves recognition accuracy over the baseline model on all far-field test sets (8.9% relative on average), it degrades performance on the clean set (-3.9%). On the contrary, both PCMN and aPCMN models improve over the baseline on both clean and far-field data sets. PCMN provides 5% relative improvement on clean data, and on average 11.2% relative improvement on far-field test sets. aPCMN improves results further on far-field data sets by providing 13.0% average relative improvement over the baseline. PCMN outperforms CMN by a significant margin on all test sets, and the adaptive extension of the model aPCMN performs better than PCMN especially on the ducking test set that exhibit dynamic channel changes.

In Table 2, we present results when the cepstral mean was estimated using entire utterance instead of sliding-window to see the upper bound of the system. All three methods CMN, PCMN, aPCMN performs better when the cepstral mean is estimated using entire utterance instead of sliding-window approximation for clean and its

Model	clean	reverb	ffield	duck	*Ave.*
NoNorm	17.8	25.6	18.2	26.4	–
CMN	18.5	25.1	15.9	23.2	–
PCMN	16.9	23.6	16.2	22.5	–
aPCMN	16.9	23.4	15.8	21.8	–
Relative Improvement					
CMN	−3.9	+1.9	+12.6	+12.1	+8.9
PCMN	+5.0	+7.8	+11.0	+14.7	+11.2
aPCMN	**+5.0**	**+8.5**	**+13.2**	**+17.4**	**+13.0**

Table 1: Results, where cepstral mean is computed over a sliding window.

reverb data. However, aPCMN that uses a sliding window cepstral mean was the best performing method on ducking and ffield test sets, outperforming its utterance-level counterpart. This observation suggests that for recordings with dynamic channel characteristics, a shorter, more dynamic channel estimate may work better.

Model	clean	reverb	ffield	duck	*Ave.*
NoNorm	17.8	25.6	18.2	26.4	–
CMN	18.6	24.8	16.1	22.4	–
PCMN	16.6	22.8	16.1	22.2	–
aPCMN	16.6	22.9	16.2	22.2	–
Relative Improvement					
CMN	−3.9	+3.1	+11.5	+15.1	+9.9
PCMN	+7.2	+10.9	+11.5	+15.9	+12.8
aPCMN	+7.2	+10.8	+11.3	+15.9	+12.7

Table 2: Results, where cepstral mean is computed over entire utterance.

Next, we present experimental results using 6,000 hours of multi-style training data in Table 3. As shown in Table 3, aPCMN provides significant improvement across all far-field noisy test sets (average 10.6% relative improvement), while still providing 4.7% improvement on clean test-set.

Model	clean	reverb	ffield	duck	ffield-2	*Ave.*
NoNorm	15.0	19.5	11.8	16.4	14.0	–
CMN	14.7	18.9	10.4	15.3	12.2	–
aPCMN	**14.3**	**18.5**	**10.4**	**14.8**	**11.8**	–
Relative Improvement						
CMN	2.0	+3.1	+11.9	+6.7	+12.9	+8.7
aPCMN	**+4.7**	**+5.1**	**+11.9**	**+9.8**	**+15.7**	**+10.6**

Table 3: Results using 6,000 hours of multi-style straining data

Finally, we conducted experiments using 9,000 hours of real far field home-speaker data using sliding-window CMN and aPCMN. For a stronger baseline, we use a model topology that includes two convolution layers and eight fully connected layers with 1024 activation units. Each convolution layer has 128 two-dimensional filters and there is a max pooling in between them. The model was trained using cross entropy loss function. Also, a well matched language model that is trained with in-domain data is used for these experiments. Results are shown in Table 4. In line with other experimental results, aPCMN provides 8.9% relative improvement over sliding-window CMN on real far field home-speaker test data.

Table 4: Results using 9,000 hours of real far-field home-speaker training data

Model	ffield-2
CMN	5.6
aPCMN	5.1

5. CONCLUSION

In this paper, we present a novel parametric cepstral mean normalization method for robust far-field ASR. Our key insight is that by learning the channel normalization jointly with the acoustic model we are able to improve ASR performance for noisy far-field conditions. PCMN model is a flexible framework that improves performance on both clean and far-field data, while only adding a small number (120) of parameters to the model. We found that making the model adaptive leads to further improvement with far-field data.

As part of our future work, we plan to train PCMN and aPCMN models with sequence training criteria.

6. ACKNOWLEDGMENT

We would like to thank Alex Acero and Don McAllaster for fruitful technical discussions.

7. REFERENCES

[1] Dong Yu and Li Deng, *Automatic speech recognition: A deep learning approach*, Springer, 2014.

[2] Wayne Xiong, Jasha Droppo, Xuedong Huang, Frank Seide, Mike Seltzer, Andreas Stolcke, Dong Yu, and Geoffrey Zweig, "The microsoft 2016 conversational speech recognition system," in *Acoustics, Speech and Signal Processing (ICASSP), 2017 IEEE International Conference on*. IEEE, 2017, pp. 5255–5259.

[3] George Saon, Sercu Tom, Hong-Kwang J Kuo, and Steven Rennie, "The ibm 2016 english conversational telephone speech recognition system," *arXiv preprint arXiv:1604.08242*, 2016.

[4] Jon Barker, Ricard Marxer, Emmanuel Vincent, and Shinji Watanabe, "The third chimespeech separation and recognition challenge: Dataset, task and baselines," in *Automatic Speech Recognition and Understanding (ASRU), 2015 IEEE Workshop on*. IEEE, 2015, pp. 504–511.

[5] Keisuke Kinoshita, Marc Delcroix, Takuya Yoshioka, Tomohiro Nakatani, Armin Sehr, Walter Kellermann, and Roland Maas, "The reverb challenge: A common evaluation framework for dereverberation and recognition of reverberant speech," in *Applications of Signal Processing to Audio and Acoustics (WASPAA), 2013 IEEE Workshop on*. IEEE, 2013, pp. 1–4.

[6] Eberhard Hänsler and Gerhard Schmidt, *Speech and audio processing in adverse environments*, Springer Science & Business Media, 2008.

[7] Bo Li, Tara N Sainath, Ron J Weiss, Kevin W Wilson, and Michiel Bacchiani, "Neural network adaptive beamforming for robust multichannel speech recognition.," in *INTERSPEECH*, 2016, pp. 1976–1980.

[8] Yuxuan Wang, Pascal Getreuer, Thad Hughes, Richard F Lyon, and Rif A Saurous, "Trainable frontend for robust and far-field keyword spotting," in *Acoustics, Speech and Signal Processing (ICASSP), 2017 IEEE International Conference on*. IEEE, 2017, pp. 5670–5674.

[9] Brian King, I-Fan Chen, Yonatan Vaizman, Yuzong Liu, Roland Maas, Sri Hari Krishnan Parthasarthi, and Björn Hoffmeister, "Robust speech recognition via anchor word embeddings," in *INTERSPEECH*, 2017.

[10] Sangita Tibrewala and Hynek Hermansky, "Multi-band and adaptation approaches to robust speech recognition," in *Fifth European Conference on Speech Communication and Technology*, 1997.

[11] Ethan Perez, Harm de Vries, Florian Strub, Vincent Dumoulin, and Aaron Courville, "Learning visual reasoning without strong priors," *arXiv preprint arXiv:1707.03017*, 2017.