



**HAL**  
open science

# Optimized Power Normalized Cepstral Coefficients Towards Robust Deep Speaker Verification

Xuechen Liu, Md Sahidullah, Tomi Kinnunen

► **To cite this version:**

Xuechen Liu, Md Sahidullah, Tomi Kinnunen. Optimized Power Normalized Cepstral Coefficients Towards Robust Deep Speaker Verification. ASRU 2021 - IEEE Automatic Speech Recognition and Understanding Workshop, Dec 2021, Cartagena, Colombia. hal-03359173

**HAL Id: hal-03359173**

**<https://hal.inria.fr/hal-03359173>**

Submitted on 30 Sep 2021

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# OPTIMIZED POWER NORMALIZED CEPSTRAL COEFFICIENTS TOWARDS ROBUST DEEP SPEAKER VERIFICATION

Xuechen Liu<sup>1,2</sup>, Md Sahidullah<sup>2</sup>, Tomi Kinnunen<sup>1</sup>

<sup>1</sup>School of Computing, University of Eastern Finland, Joensuu, Finland

<sup>2</sup>Université de Lorraine, CNRS, Inria, LORIA, F-54000, Nancy, France

## ABSTRACT

After their introduction to robust speech recognition, power normalized cepstral coefficient (PNCC) features were successfully adopted to other tasks, including speaker verification. However, as a feature extractor with long-term operations on the power spectrogram, its temporal processing and amplitude scaling steps dedicated on environmental compensation may be redundant. Further, they might suppress intrinsic speaker variations that are useful for speaker verification based on deep neural networks (DNN). Therefore, in this study, we revisit and optimize PNCCs by ablating its medium-time processor and by introducing channel energy normalization. Experimental results with a DNN-based speaker verification system indicate substantial improvement over baseline PNCCs on both in-domain and cross-domain scenarios, reflected by relatively 5.8% and 61.2% maximum lower equal error rate on VoxCeleb1 and VoxMovies, respectively.

**Index Terms**— acoustic feature extraction, speaker verification, power normalized cepstral coefficients.

## 1. INTRODUCTION

*Automatic speaker verification* (ASV) [1, 2] aims at verifying speaker’s identity from a speech waveform. To this end, an ASV system extracts acoustic features used in speaker comparison. *Mel-frequency cepstral coefficients* (MFCCs) [3] and *mel filterbank* outputs are two widely-used feature sets. The acoustic features are used to obtain a recording-level speaker embedding, such as an *i-vector* [4].

Thanks to extensive studies on deep neural network (DNN), the performance of ASV system has been substantially improved by replacing different parts of the system by DNN-based components. Improvements to speaker embeddings have been obtained through models such as *deep speaker* [5], *recurrent end-to-end network* [6], and *x-vector* [7]. DNN-based methods have been applied to scoring back-end as well [8] to improve upon statistical classifiers [9].

Furthermore, *end-to-end* front-end methods have been investigated in various applications, including ASV. In these approaches, the feature extraction and speaker information extraction are integrated together. Representative examples in-

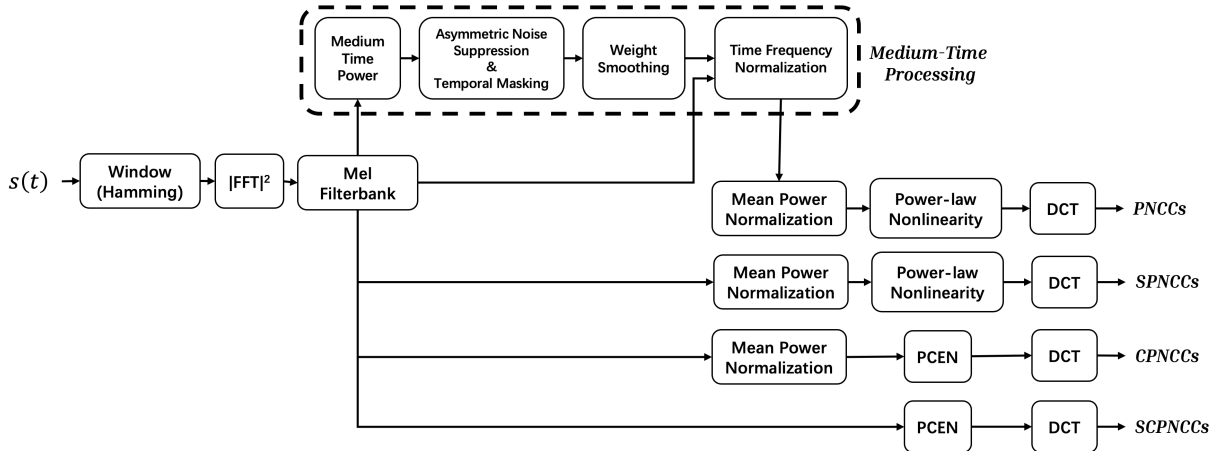
clude SincNet [10, 11] and its variations [12]. Even if promising results have been reported, end-to-end methods may have increased computational complexity, be more dependent on data, and be more challenging to interpret compared to traditional hand-crafted features.

Meanwhile, MFCCs obtained through *short-term Fourier transform* (STFT) lack specificity to long-term speech characteristics and may lack robustness. Several alternatives have been developed for robust ASV and evaluated on the previous-generation statistical ASV methods. Feature extraction methods including some form of *long-term processing* steps — such as *power-normalized cepstral coefficients* (PNCCs) [13] — have indicated their potential in improving robustness over short-term features. After their introduction, PNCCs have been studied in different tasks, including ASV [14, 15]. Nonetheless, while showing promise in speech recognition and keyword spotting tasks, the computational operations of PNCC are heavily dedicated to compensating environmental variations. While environment compensation is important in ASV as well, some of the PNCC operations may suppress intrinsic speaker variations useful in ASV tasks. Meanwhile, such variations have been studied earlier and demonstrated its efficacy in deep speaker embeddings [15].

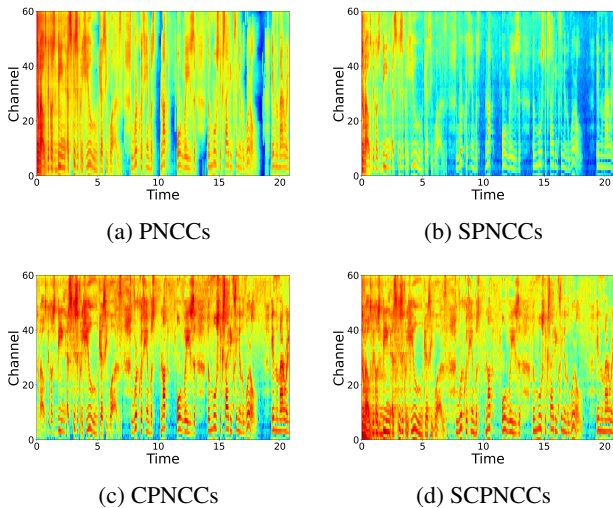
With the intention of developing more proper compensation techniques for ASV feature extraction, we leverage from the PNCCs as a case study. We extend its use with recent data and framework as a starting point, comparing it to MFCCs. We first try to simplify its medium-time processing module, with hypothesis on its redundancy and negative impact on ASV performance. Furthermore, we address the usefulness of channel-wise robust normalization techniques, compensating the potential loss and providing more robustness. To the best of our knowledge, this is the first work on introducing and optimizing PNCCs for speaker verification systems using deep speaker embeddings.

## 2. POWER NORMALIZED CEPSTRAL COEFFICIENTS

The full pipeline of PNCCs [13] is illustrated in Fig. 1. Its original design is motivated by the practical need on robust



**Fig. 1:** Block diagram of PNCC and proposed feature extractors. PNCCs are adopted from [13]. SPNCCs = Simple PNCCs [16]; CPNCCs = Channel-normalized PNCCs; SCPNCCs = Simple Channel-normalized PNCCs.



**Fig. 2:** Effect of different feature extractors on mel spectrogram representation on a audio sample from VoxCeleb1. Number of filters is 60 for all cases. No DCT applied.

features with respect to undesirable acoustic variations. Compared with established feature sets such as MFCCs [3] and RASTA-PLPs [17], PNCCs introduce additional signal enhancement and noise compensation operations on the spectrogram. As illustrated, those operations are represented as four steps: i) medium-time power analysis, ii) asymmetric noise suppression and temporal masking, iii) weight smoothing, and iv) time-frequency normalization. These operations are collectively referred to as *medium-time processing* [18] to distinguish them from conventional short-term processing operations such as the Fourier transform.

We consider time-frequency normalization as part of

medium-time processing since it can be simply expressed as a dot-product:  $\bar{E} = E \cdot \bar{S}$ , where  $E$  represents the short-time spectral energies after mel integration and  $\bar{S}$  denotes the output transfer function from weight smoothing. It is followed by mean power normalization, inspired from *automatic gain control* (AGC) [19] theory. Also, the logarithmic nonlinearity in MFCC processing is replaced by a power-law nonlinearity before discrete cosine transform (DCT). In the original PNCC study [13], the authors also promote the usage of Gammatone filterbanks [20] by noting that it may have small impact on performance compared with mel integration. Those remarks are supported by their pilot experiments. We use mel filterbank outputs for all experiments in this study.

### 3. OPTIMIZATION STRATEGIES ON PNCCS

In this section we describe our PNCC optimization strategies for robust deep ASV systems. First, we ablate the medium-time processing to obtain the *simple power-normalized cepstral coefficients* (SPNCCs) [16]. We then measure the efficacy of *per-channel energy normalization* (PCEN) [21, 22]. PCEN is designed to either replace, or to jointly operate with mean power normalization. These operations act as normalizers on the channel variations and other intrinsic variations.

#### 3.1. Ablation of Medium-time Processing

The medium-time processor is intended for compensating potential environmental degradation, by producing a smoothed spectral envelope. Weight smoothing, in turn, helps to suppress small perturbations. These operations can enhance speech recognition (ASR) performance [18]. However, for modern DNN-based ASV some of these operations are questionable. In particular, the smooth spectral envelope may suppress speaker-related cues, especially in voiced regions.

This raises questions on the necessity of the noise suppression and smoothing operations.

Removing the medium-time module results in the SPNCCs, which has been applied earlier to robust ASR [16]. Because of mean power normalization, the residual pattern with respect to time and frequency can be preserved. This can be seen in Fig. 2 as well, from the perspective of time-frequency representations — the contrast between the onsets and offsets remains. This motivates us to re-evaluate SPNCCs in robust ASV.

### 3.2. Mean Power Normalization

The module right after the medium-time processor is mean-power normalization. The mean power estimate is computed by  $\mu[t] = \lambda_\mu \mu[t-1] + (1-\lambda_\mu) \sum_{f=0}^{F-1} E[t, f]/L$ , where  $t$  and  $f$  are, respectively, the time and frequency indices.  $F$  denotes number of frequency bins.  $E[t, f]$  represents power spectral energies, either directly from mel integration or after medium-time processing.  $\lambda_\mu = 0.999$  is known as *forgetting factor*, resembling the smoothing factor in infinite impulse response (IIR) filtering. The mean power estimate is a running average of the current and preceding frame energies. The normalized power is then calculated as  $\tilde{E}[t, f] = E[t, f]/\mu[t]$ .

As noted above, mean power normalization has the effect of scaling the power spectrum while retaining the local contrast. This echoes the main aim of developing such module in PNCC, which is to compensate the effect of power-law non-linearity. However, what also can be indicated from the figures is that there are useful speech onsets that have been normalized. This may due to the power law scaling as discussed, which motivates the usage of advanced parameterized methods to properly normalize the spectrogram energies. As one of the methods whose original version have been validated effective for various speech tasks [21, 23], PCEN is discussed in next sub-section.

### 3.3. Per-channel Energy Normalization (PCEN)

As mentioned, PCEN was developed originally as a replacement of logarithmic nonlinearity. The PCEN operation is expressed as:

$$\text{PCEN}[t, f] = \left( \frac{E[t, f]}{(M[t, f] + \epsilon)^\alpha} + \delta \right)^r - \delta^r, \quad (1)$$

where  $E[t, f]$  represents the power spectral energies,  $M[t, f] = (1-s)M[t-1, f] + sE[t, f]$  is the smoothed energies using an auto-regressive filter. Here we set  $s$  as the reciprocal of number of frequency bins.  $\alpha \in (0, 1]$  is the main parameter for compressing smoothed energies.  $\delta$  and  $r$  parameterize dynamic range compression, mainly aiming for properly modeling the loudness variations.

In its original study [21], the authors treat PCEN as a replacement for logarithmic nonlinearity to address the problem of singularities and time-varying loudness. Therefore, when integrated to PNCCs, PCEN is regarded as either a replacement of power-law nonlinearity, or mean power normalization and nonlinearity. We implement both cases, resulting in *channel-normalized power cepstral coefficients* (CPNCCs) and *simple channel-normalized power cepstral coefficients* (SCPNCCs). Their effect is illustrated in Fig. 2c and 2d. Compared to PNCCs and SPNCCs, we can see that by applying PCEN, the spectrogram is enhanced, especially for higher frequency components. Meanwhile, CPNCCs preserve relatively more offsets at lower frequency bins and cast more enhancing effect relatively at higher ones than SCPNCCs. This may indicate the instability of AGC operation in PCEN, which can be compensated by placing mean power normalization as a predecessor.

Eval sets	Positive.	Negative.	#target	#non-target
E-1	D-M ( <i>same</i> )	D-M	10286	10286
E-2	D-I, D-M	D-I, D-M	23289	23289
E-3	D-I, D-M	D-I	23402	23402
E-4	D-I, D-M	D-M	23433	23433
E-5	D-M ( <i>diff</i> )	D-M	20545	20545

**Table 1:** Trial statistics of VoxMovies, taken from [24]. The speech material originates from two domains: interview videos (D-I) and movie clips (D-M). Enroll and test files may come from same movie (*same*) or different movies (*diff*).

## 4. EXPERIMENTS

In this section, we describe the experimental protocol on multiple descendants of the VoxCeleb datasets [25, 26], with varied level of difficulty. We analyze the performance of different optimized PNCCs in both in-domain and cross-domain scenarios.

### 4.1. Data

**Training.** For all experiments we acquire speech data from the *dev* of VoxCeleb2 [26] to train DNN speaker embedding extractors. It is sourced from online interview videos and consists of 5994 speakers in total. We perform data augmentation by creating copies using room impulse response (RIR) [27] and multiple sound sources from the MUSAN corpus [28], including noise, babble and music.

**Evaluation.** We follow the official protocols of the VoxCeleb speaker recognition challenge 2019 [29], generating two trials from VoxCeleb1 [25] and denoted as *VoxCeleb1-E* and *VoxCeleb1-H*, representing the *in-domain* scenario. We also include the recent VoxMovies [24] corpus in our experiments. It is sourced from movie clips with speakers selected

from the VoxCeleb datasets. This creates an interesting *cross-domain* speaker verification scenario. There are different trials defined based on whether audio source originates from the interviews or movie clips. There are five different trial lists, marked as E-1 to E-5. Their statistics are summarized in Table 1. More details about such dataset can be found in [24], according to which the level of mismatch and verification difficulty increases from E-1 to E-5. The trial list of *pooled* condition is simply the union of the five trial lists.

## 4.2. System Configuration

For all the experiments, the dimensionality of the acoustic features is set as 30, including MFCCs, without dynamic features. For all systems with the power-law nonlinearity, we set the scaling factor to  $1/15$ , following original PNCCs. Extended x-vector based on time-delay neural network (TDNN) [30] with attentive statistics pooling [31] and angular margin softmax training loss [32] forms the speaker embedding extractor. Adam [33] optimizer is applied for training, with minibatch size of 128. All speaker embeddings are extracted from the first fully-connected layer after the pooling layer. They are subsequently length-normalized and centered, prior to being transformed by a linear discriminant analyzer (LDA). Probabilistic linear discriminant analysis (PLDA) [9] is applied to produce log-likelihood scores.

## 4.3. Evaluation

We employ equal error rate (EER) and minimum detection cost function (minDCF) to evaluate ASV performance. MinDCFs were computed with target speaker prior  $p_{tar} = 0.01$  and detection costs  $C_{fa} = C_{miss} = 1.0$ .

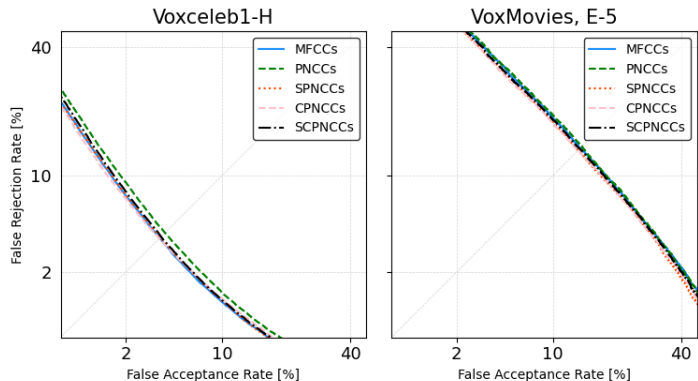
Feature	VoxCeleb1-E		VoxCeleb1-H	
	EER(%)	minDCF	EER(%)	minDCF
MFCCs	<b>1.88</b>	0.1883	3.74	0.5808
PNCCs	2.39	0.1524	3.91	0.6428
SPNCCs	1.92	0.142	3.77	0.5721
CPNCCs	2.08	<b>0.0867</b>	<b>3.52</b>	<b>0.5698</b>
SCPNCs	2.22	0.2874	3.84	0.5993

**Table 2:** Results on VoxCeleb1 evaluation sets.

## 5. SPEAKER VERIFICATION RESULTS

### 5.1. Voxceleb1

The results on VoxCeleb1 reported in table 2 indicate that the performance of the feature extractors differs between the two conditions. Let us first look at *VoxCeleb1-E*, the least challenging condition across all covered in this study. While CPNCCs achieve the lowest minDCF among all systems, in terms of EERs none of the proposed features outperform



**Fig. 3:** DET visualization of systems with different feature extractors on *VoxCeleb1-H* (left) and E-5 from VoxMovies (right). Best viewed in color.

conventional MFCCs. This confirms the efficacy of MFCCs under in-domain or relatively matched conditions. Among PNCC variants, the lowest EER is obtained from SPNCCs.

We then move on to the more challenging *VoxCeleb1-H*. While standard PNCCs still remain behind MFCCs, removing the medium-time processing component provides a 3.6% relative improvement in EER. Applying PCEN furthers such improvement and results in best performance among all features in both metrics, outperforming MFCCs by relatively 5.8% in EER and 18.9% in minDCF. These results indicate the robustness provided by PCEN in the more challenging *VoxCeleb1-H* condition. Meanwhile, removing mean power normalization, again degrades performance. Mean power normalization thus is speculated to have an improvement role in normalizing in-domain acoustic variations.

Feature	<i>Pooled</i>	E-1	E-2	E-3	E-4	E-5
MFCCs	13.15	11.04	12.04	9.22	18.93	14.43
PNCCs	13.72	10.70	12.84	6.79	20.17	14.76
SPNCCs	13.09	<b>10.14</b>	12.04	4.91	19.53	13.08
CPNCCs	<b>13.0</b>	10.39	<b>11.72</b>	<b>3.57</b>	<b>18.3</b>	<b>13.03</b>
SCPNCs	13.30	10.30	12.20	5.65	18.42	14.36

**Table 3:** EER(%) results on VoxMovies.

### 5.2. VoxMovies

Performance of different feature extractors on VoxMovies trial sets in terms of EER is presented in table 3. The standard PNCCs outperforms MFCCs slightly on first three conditions but not on E-4 and E-5. Removing medium-time processing results in improvement across all conditions, including *pooled*. VoxMovies contains more intrinsic speaker variations and domain mismatch compared to the VoxCeleb1 conditions. Therefore, based on results from MFCCs, PNCCs

and SPNCCs, medium-time processing is suspected to have a negative impact.

Moving on to the two variants with PCEN, best performance on all conditions and *pooled*, excluding E-1, is achieved by CPNCCs, with maximum relative improvement of 61.2% observed in the E-3 condition. Since E-1 contains least domain mismatch according to Table 1, this result is expected. For the most challenging E-5 it reaches 9.5% relative improvement in EER over MFCCs. Therefore, integrating mean power normalization and PCEN thus may have positive impact on normalizing the intrinsic variations, either as an addition or replacement from the perspective of medium-time processing. Ablating mean power normalization gain (SCP-NCCs) leads to performance drop on all conditions except E-1, but this time SCPNCCs outperforms MFCCs on four out of the five conditions. This indicates the efficacy of channel normalization on reducing domain mismatch, without requiring extensive parameter tuning. Future work thus can focus on domain adaptation on related parameters via fine-tuning or making data-adaptive version of PCEN, for either improving CPNCCs or refining SCPNCCs, as described in [21].

### 5.3. Analysis with DET plot

We analyze the performance of different feature extractors on the two challenging conditions: *VoxCeleb1-H* and E-5 from *VoxMovies*. For *VoxCeleb1-H*, we can observe that while the gain from CPNCCs are reflected, for scenario where systems are less strict on false alarms (right bottom), MFCCs may be a better choice over the proposed PNCC variants. This is different from what we see at the other plot, where the gain from CPNCCs and SPNCCs over other systems are consistent, echoing what has been reported via EER.

## 6. CONCLUSIONS

We have re-evaluated and optimized power normalized cepstral coefficients (PNCCs) for deep neural network based speaker verification. We first ablated the medium-time processing module, which results in simple PNCCs (SPNCCs) and indicates promising performance. We did further simplification by applying PCEN. We performed experiments in both in-domain (*VoxCeleb1*) and cross-domain (*VoxMovies*) scenarios. Among all the observed improvements, we highlight particularly the successful combination of mean power normalization and PCEN as a replacement of power-law non-linearity and medium-time processing (CPNCCs). In the cross-domain scenario, the relative maximum improvement over the baseline was 61.2%.

In future work, we may focus on creating data-adaptive or pre-trained variants with optionally sequential training criteria, similar to what has been done in [21]. Exploring individual PCEN components (AGC and DRC) as alternative choices

from nonlinearities such as logarithmic and power-law is also left as future work.

## 7. ACKNOWLEDGEMENTS

This work was partially supported by Academy of Finland (project 309629) and Inria Nancy Grand Est.

## 8. REFERENCES

- [1] J.H.L. Hansen and T. Hasan, "Speaker recognition by machines and humans: A tutorial review," *IEEE Signal Processing Magazine*, vol. 32, no. 6, pp. 74–99, 2015.
- [2] Z. Bai and X. Zhang, "Speaker recognition based on deep learning: An overview," *Neural Networks*, vol. 140, pp. 65–99, 2021.
- [3] S. Davis and P. Mermelstein, "Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 28, no. 4, pp. 357–366, 1980.
- [4] N. Dehak, P. J. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, "Front-end factor analysis for speaker verification," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 4, pp. 788–798, 2011.
- [5] C. Li, X. Ma, B. Jiang, X. Li, X. Zhang, X. Liu, Y. Cao, A. Kannan, and Z. Zhu, "Deep Speaker: an end-to-end neural speaker embedding system," *CoRR*, vol. abs/1705.02304, 2017.
- [6] L. Wan, Q. Wang, A. Papir, and I. Moreno, "Generalized end-to-end loss for speaker verification," in *Proc. ICASSP*, 2018, pp. 4879–4883.
- [7] D. Snyder, D. Garcia-Romero, G. Sell, D. Povey, and S. Khudanpur, "X-Vectors: Robust DNN embeddings for speaker recognition," in *Proc. ICASSP*, 2018, pp. 5329–5333.
- [8] S. Ramoji, P. Krishnan, and S. Ganapathy, "NPLDA: A deep neural plda model for speaker verification," in *Proc. Odyssey 2020*, 2020, pp. 202–209.
- [9] S. Ioffe, "Probabilistic linear discriminant analysis," in *ECCV 2006*, 2006, pp. 531–542.
- [10] M. Ravanelli and Y. Bengio, "Speaker recognition from raw waveform with SincNet," in *2018 IEEE Spoken Language Technology Workshop (SLT)*, 2018, pp. 1021–1028.
- [11] M. Pariente, S. Cornell, A. Deleforge, and E. Vincent, "Filterbank design for end-to-end speech separation," in *Proc. ICASSP*, 2020, pp. 6364–6368.

- [12] S. Pascual, M. Ravanelli, J. Serrà, A. Bonafonte, and Y. Bengio, “Learning problem-agnostic speech representations from multiple self-supervised tasks,” in *Proc. Interspeech 2019*, 2019, pp. 161–165.
- [13] C. Kim and R. Stern, “Power-normalized cepstral coefficients (PNCC) for robust speech recognition,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 24, no. 7, pp. 1315–1329, 2016.
- [14] Ning W. and Lei W., “Robust speaker recognition based on multi-stream features,” in *2016 IEEE International Conference on Consumer Electronics-China (ICCE-China)*, 2016, pp. 1–4.
- [15] X. Liu, M. Sahidullah, and T. Kinnunen, “A Comparative Re-Assessment of Feature Extractors for Deep Speaker Embeddings,” in *Proc. Interspeech 2020*, 2020, pp. 3221–3225.
- [16] M. J. Harvilla and R. M. Stern, “Histogram-based sub-band power warping and spectral averaging for robust speech recognition under matched and multistyle training,” in *Proc. ICASSP*, 2012, pp. 4697–4700.
- [17] H. Hermansky and N. Morgan, “RASTA processing of speech,” *IEEE Transactions on Speech and Audio Processing*, vol. 2, no. 4, pp. 578–589, 1994.
- [18] C. Kim and R. M. Stern, “Power-normalized cepstral coefficients (PNCC) for robust speech recognition,” in *Proc. ICASSP*, 2012, pp. 4101–4104.
- [19] J. Prez, S. C. Pueyo, and B. Lpez, *Automatic Gain Control: Techniques and Architectures for RF Receivers*, Springer-Verlag New York, 2011.
- [20] R. D. Patterson, K. Robinson, J. Holdsworth, D. Mckeown, C. Zhang, and M. Allerhand, “Complex sounds and auditory images,” in *Proc. 9th Int. Symp. Hearing Audit., Physiol. Perception*, 1992, pp. 429–446.
- [21] Y. Wang, P. Getreuer, T. Hughes, R. F. Lyon, and R. A. Saurous, “Trainable frontend for robust and far-field keyword spotting,” in *Proc. ICASSP*, 2017, pp. 5670–5674.
- [22] V. Lostanlen, J. Salamon, M. Cartwright, B. McFee, A. Farnsworth, S. Kelling, and J. Bello, “Per-channel energy normalization: Why and how,” *IEEE Signal Processing Letters*, vol. 26, no. 1, pp. 39–43, 2019.
- [23] C. Ick and B. McFee, “Sound event detection in urban audio with single and multi-rate PCEN,” in *Proc. ICASSP*, 2021, pp. 880–884.
- [24] A. Brown, J. Huh, A. Nagrani, J. Chung, and A. Zisserman, “Playing a part: Speaker verification at the movies,” in *Proc. ICASSP*, 2021, pp. 6174–6178.
- [25] A. Nagrani, J. Chung, and A. Zisserman, “VoxCeleb: A large-scale speaker identification dataset,” in *Proc. Interspeech*, 2017, pp. 2616–2620.
- [26] J. Chung, A. Nagrani, and A. Zisserman, “VoxCeleb2: deep speaker recognition,” in *Proc. Interspeech*, 2018, pp. 1086–1090.
- [27] T. Ko, V. Peddinti, D. Povey, M. L. Seltzer, and S. Khudanpur, “A study on data augmentation of reverberant speech for robust speech recognition,” in *Proc. ICASSP*, 2017, pp. 5220–5224.
- [28] D. Snyder, G. Chen, and D. Povey, “MUSAN: A music, speech, and noise corpus,” *CoRR*, vol. abs/1510.08484, 2015.
- [29] J. Chung, A. Nagrani, E. Coto, W. Xie, M. McLaren, D. A. Reynolds, and A. Zisserman, “VoxSRC 2019: The first voxceleb speaker recognition challenge,” in *ISCA archive*, 2019.
- [30] D. Snyder, D. Garcia-Romero, G. Sell, A. McCree, D. Povey, and S. Khudanpur, “Speaker recognition for multi-speaker conversations using x-vectors,” in *Proc. ICASSP*, 2019, pp. 5796–5800.
- [31] K. Okabe, T. Koshinaka, and K. Shinoda, “Attentive statistics pooling for deep speaker embedding,” in *Proc. Interspeech 2018*, 2018, pp. 2252–2256.
- [32] F. Wang, J. Cheng, W. Liu, and H. Liu, “Additive margin softmax for face verification,” *IEEE Signal Processing Letters*, vol. 25, no. 7, pp. 926–930, 2018.
- [33] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” in *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015.