Wavelet Scattering Transform and CNN for Closed Set Speaker Identification

Wajdi GHEZAIEL
Normandie Univ
ENSICAEN,UNICAEN
CNRS NormaSTIC
14000 Caen, France
wajdi.ghezaiel@ensicaen.fr

Luc BRUN
Normandie Univ
ENSICAEN,UNICAEN
CNRS Greyc, UMR 6072
14000 Caen, France
luc.brun@ensicaen.fr

Olivier LÉZORAY
Normandie Univ
ENSICAEN,UNICAEN
CNRS Greyc, UMR 6072
14000 Caen, France
olivier.lezoray@unicaen.fr

***Abstract*—In real world applications, the performances of speaker identification systems degrade due to the reduction of both the amount and the quality of speech utterance. For that particular purpose, we propose a speaker identification system where short utterances with few training examples are used for person identification. Therefore, only a very small amount of data involving a sentence of 2-4 seconds is used. To achieve this, we propose a novel raw waveform end-to-end convolutional neural network (CNN) for text-independent speaker identification. We use wavelet scattering transform as a fixed initialization of the first layers of a CNN network, and learn the remaining layers in a supervised manner. The conducted experiments show that our hybrid architecture combining wavelet scattering transform and CNN can successfully perform efficient feature extraction for a speaker identification, even with a small number of short duration training samples.**

***Index Terms*—Speaker identification, short utterances, wavelet scattering transform, convolutional neural network, hybrid network.**

I. Introduction

Smart home speakers have recently become very popular and allow the voice-based launching of applications. This can be problematic for security reasons, especially for applications that enable online purchase. In such cases, the speaker has to be identified to ensure a secure transaction. To enable this, the speaker has first to be enrolled. Such an enrollment is usually performed from few short sentences. In such a practical scenario there may be no constraints on such sentences. Moreover, these sentences are usually short and only few utterances are provided to the system. Different studies [1], [2] have shown that the use of short segments may induce a drastic drop of the performances of speaker identification systems. This drop in performance is mainly due to the low amount of information extracted for each speaker. Speaker identification with only few and short utterances is thus a challenging problem. Moreover, this work takes place within the framework of the HomeKeeper [1] project which aims at providing smart home speakers dedicated to regional needs (healthcare, local radios,...). In order to avoid the need of an external identification server, the consortium of the project has decided to perform the identification step directly on the terminal. This means that each terminal has at its disposal only a closet set reduced database of short sentences corresponding typically to the local members of a family (less than 10 persons).

Traditionally, speaker identification systems are based on the extraction of features relying on speech production and perception such as Mel-Frequency Cepstral Coefficients (MFCCs). During the training phase, such features are extracted from a large amount of speakers. A Gaussian Mixture model (GMM) is then trained to build a Universal Background Model (UBM) [3]. During the enrollment phase of a speaker, the mean of the GMM is adapted to fit the speaker's data. In GMM-UBM systems, the stacked mean vectors are directly used as the representation of the speaker. However, it has been shown [4], [5] that it is beneficial to further process this vector by extracting intermediate vectors called i-vectors. During the identification phase, an i-vector is extracted from a given speech sample and is compared to the reference i-vector, either with a simple cosine distance or with more complex techniques such as Probabilistic Linear Discriminant Analysis (PLDA) [6].

In recent years, deep learning has appeared in many pattern recognition fields. It has shown remarkable success in many fields such as image recognition [7] and natural language processing [8]. In speaker identification, a similar trend has been observed. Deep Neural Networks (DNNs) have been used with the i-vector framework to compute Baum-Welch statistics [9], or for frame-level feature extraction [10]. DNNs have also been proposed for direct discriminative speaker classification, as witnessed by the recent literature on this topic [11], [12]. Lately, there was an increasing number of studies using convolutional neural networks (CNNs) [13] in numerous speech tasks [14]. Some works have proposed to directly feed networks with spectrogram bins [15] or even with raw waveforms [16], [17]. Among DNNs, CNNs have the most suitable architecture for processing raw speech samples, since weight sharing, local filtering, and pooling constitute precious tools to discover robust and invariant representations.

This work was supported by BPI France, project HomeKeeper.

[1] https://home-keeper.io/

978-1-7281-9320-5/20/$31.00 ©2020 IEEE

However, CNN networks require numerous labeled training examples along with considerable computational resources and time to achieve effective learning. In a setting where only few labeled data of short duration are available, the training becomes difficult and requires a lot of regularization.

Feature extraction and representation is a critical point of classification systems which, if correctly handled, can allow to reduce the size of labeled training datasets. The wavelet scattering transform, a rich representation, has enjoyed significant success in various audio [18] and biomedical [19] signal classification tasks. Its structure is that of a convolutional neural network [18], but with fixed filters. This last point is important when only few training samples are available. Specifically, this transformation alternates convolutions with wavelet filters and pointwise nonlinearities to ensure time-shift invariance and time-warping stability [20]. Scattering representations can be plugged into any classification or regression system, be it shallow or deep. The original experiments of Andén and Mallat [18] on deep scattering spectrum relied on support vector machines (SVM) with linear or Gaussian kernels. For supervised large-vocabulary continuous speech recognition, replacing these locally linear classifiers by five layers of deep neural networks (DNN) or deep convolutional networks (ConvNets) only brought marginal improvements in accuracy [21]. However, in the Zero Resource Speech Challenge [22], whose aim is to discover sub-word and word units from continuous speech in an unsupervised way, associating scattering representations with deep siamese network provided a substantial gain in the trade off between inter-class discriminability and inter-speaker robustness [23]. Wavelet scattering transform features demonstrated promising results on the dataset TIMIT for phonetic classification [18] and recognition [21]. In this paper, we explore the use of the Wavelet Scattering Transform (WST) for feature extraction along with convolutional neural network for closed set speaker identification. In this hybrid deep learning network, wavelet scattering coefficients generated in first few layers capture the dominant energy contained in the input data patterns.

The reminder of the paper is organized as follows. Section II discusses wavelet scattering transform. Section III describes the proposed hybrid architecture, which is a cascade of a wavelet scattering transform and a convolutional neural network. Section IV discusses the experimental setup and the corresponding results obtained by the proposed system as well as the ones provided by related systems.

II. Wavelet Scattering transform

The wavelet scattering transform (WST), introduced in [18], [20], is a deep representation, obtained by iterative application of the wavelet transform modulus. It has been defined so as to be invariant to translations of the input signal, and stable to small deformations. The authors of [18], [20] have demonstrated how the WST transform can extract significant features at different scales. WST has been successfully applied to different classification tasks, textures [24], [25], small digits [24], sounds [18] or complex image datasets with unsupervised representations [19]. Moreover, it has been proven [26] that the WST coefficients are more informative than a Fourier transform when dealing with short variation signals or small deformation and rotation. WST consists in a cascade of wavelet transforms and modulus nonlinearities. To produce a wavelet scattering transform of an input signal x, three successive operations are required: convolution, nonlinearity, and averaging as described in Figure 1. The WST coefficients are obtained with the averaging of wavelet modulus coefficients by a low-pass filter ϕ.

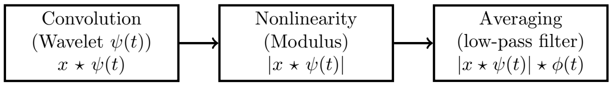

Fig. 1. Wavelet scattering transform processes, where x is the input data, ψ a wavelet function and ϕ an averaging low-pass filter.

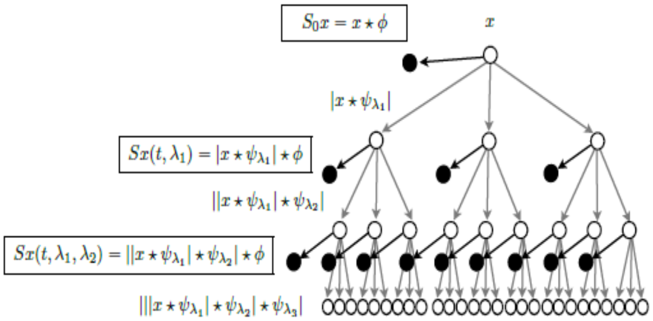
$S_0x = x \star \phi$
x
$|x \star \psi_{\lambda_1}|$
$Sx(t, \lambda_1) = |x \star \psi_{\lambda_1}| \star \phi$
$||x \star \psi_{\lambda_1}| \star \psi_{\lambda_2}|$
$Sx(t, \lambda_1, \lambda_2) = ||x \star \psi_{\lambda_1}| \star \psi_{\lambda_2}| \star \phi$
$|||x \star \psi_{\lambda_1}| \star \psi_{\lambda_2}| \star \psi_{\lambda_3}|$

Fig. 2. Hierarchical representation of wavelet scattering coefficients at multiple layers [18].

Let a wavelet $\psi(t)$ be a band pass filter with a central frequency normalized to 1, and $\psi_\lambda(t)$ a wavelet filter bank, which is constructed by dilating the wavelet:

$$\psi_\lambda(t) = \lambda\psi(\lambda t) \tag{1}$$

where $\lambda = 2^{\frac{j}{Q}}$, $\forall j \in \mathbb{Z}$ and Q is the number of wavelets per octave. The bandwidth of the wavelet $\psi(t)$ is of the order $\frac{1}{Q}$, and as a result, the filter bank is composed of band pass filters which are centered in the frequency domain in λ and have a frequency bandwidth $\frac{\lambda}{Q}$. At the zero order, we have a single coefficient given by $S_0x(t) = x \star \phi(t)$, where $\star$ is the convolution operator. This coefficient is close to zero for speech signals. At the first order, we set $Q_1 = 8$ for speech signals, which defines wavelets having the same frequency resolution as mel-frequency filters. Approximate mel-frequency spectral coefficients are obtained by averaging the wavelet modulus coefficients with ϕ:

$$S_1x(t, \lambda_1) = |x \star \psi_{\lambda_1}| \star \phi(t) \tag{2}$$

The second order coefficients capture the high-frequency amplitude modulations occurring at each frequency band of the first layer and are obtained by:

$$S_2x(t, \lambda_1, \lambda_2) = ||x \star \psi_{\lambda_1}| \star \psi_{\lambda_2}| \star \phi(t) \quad (3)$$

The wavelets ψ_{λ_2} have an octave resolution Q_2 which may be different from Q_1. We set $Q_2 = 1$ for speech signals, to defines wavelets with more narrow time support, which are better adapted to characterize transients and attacks. We get a sparse representation which means concentrating the signal information over as few wavelet coefficients as possible. These coefficients are averaged by the low pass filter ϕ, which ensures local invariance to time-shifts, as with the first-order coefficients.

Fig. 2 shows the hierarchy of wavelet scattering coefficients. This somewhat resembles to the structure of deep neural networks, although in the WST, each layer provides some output, while the only output of most of deep neural networks is provided by the last layer. This decomposition on first and second orders scattering coefficients, is applied to the time domain signal. Features of the second order are normalized by features of the first order, just to ensure that the higher order of scattering depends on the amplitude modulation component of the speech signal. The first and the second orders of the WST are concatenated to form a scattering feature vector for a given frame. The scattering features include log-mel features together with higher order features to preserve greater detail in the speech signal [18]. This representation is invariant to time shifts and is stable to deformations. To ensure invariability to frequency translation on a logarithmic scale like translation of speaker formants, the logarithm is applied to each coefficients of the scattering feature vector. It is thus locally translation invariant in time and log frequency, and stable to time and frequency deformations.

III. Hybrid Network Architecture

The proposed hybrid network is composed of a WST for feature extraction and a convolution neural network for classification in the back end. We thus propose to initialize the first layer of our CNN with WST feature maps. This map of features shows n_f feature vectors with n_s coefficients of scattering. Each feature vector is defined as a log of normalized scattering feature vectors (section II) for each frame. It acts as a subsampled feature map for the first CNN layer. We use Gabor and Morlet wavelet to obtain the scattering features map. As stated in section II we set the quality factor to 8 filters per octave for the first level and set 1 filter per octave at the second level. This configuration was chosen to match the frequency resolution of Mel filters at the first level [18]. The second order of the scattering transform recovers the lost information. Therefore, the representation of speech signals using the first and the second orders of the scattering transform extends the MFCC representation and doesn't loose information. These scattering coefficients are computed using 50% overlapping windows using a publicly available toolbox [18]. The size of final scattering feature maps is $n_s \times n_f$.

The proposed CNN is composed with three convolution blocks and one fully connected layer. Each convolution block

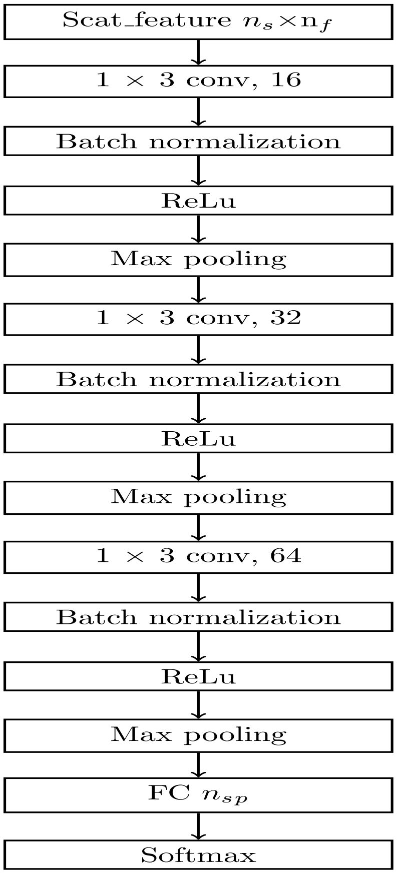

Fig. 3. Proposed Hybrid Network.

TABLE I
SCATTCNN ARCHITECTURE. EACH ROW SPECIFIES THE # OF CONVOLUTIONAL FILTERS, THEIR SIZES, AND THE # FILTERS. THIS ARCHITECTURE HAS 18,1 MILLIONS PARAMETERS FOR SCATTERING FEATURE MAP OF SIZE $433 \times 16 \times 1$.

Layer name	**ScatCNN**	**Output**
Input	inputlayer	$n_s \times n_f \times 1$
Conv1 block	conv1D, $1 \times 3, 16$ bn relu	$n_s \times n_f \times 16$
Pooling	maxpool, 1×2, stride (1,2)	$n_s \times n_f/2 \times 16$
Conv2 block	conv1D, $1 \times 3, 32$ bn relu	$n_s \times n_f/2 \times 32$
Pooling	maxpool, 1×2, stride (1,2)	$n_s \times n_f/4 \times 32$
Conv3 block	conv1D, $1 \times 3, 64$ bn relu	$n_s \times n_f/4 \times 64$
Pooling	maxpool, 1×2, stride (1,2)	$n_s \times n_f/8 \times 64$
Embedding	fc,n_{sp}	n_{sp}
Loss	softmax	$--$

is formed by 1D convolution layer of length 3 and batch normalization. Each convolutional layer is followed by a max-pooling layer, with pooling size 1×2 and stride 1×2. The network has 16, 32 and 64 filters, respectively. A fully connected layer with n_{sp} hidden neurons, where n_{sp} is the number of speakers to be identified, is connected to categorical softmax layer. We use rectified linear units as activation functions in all layers. This architecture takes raw speech with fixed length, to produce speaker embedding at frame-level. The proposed architecture is shown in Figure 3. Details such as number of filters and kernel sizes are summarized in Table I. The amount of parameters in this neural network is 18,1 millions.

IV. Experiments

This section describes the experiments and the results obtained with our approach and related systems.

A. Dataset and experimental setting

Two datasets are used in the experiments, TIMIT [27] and LibriSpeech [28]. The TIMIT dataset contains studio quality recordings of 630 speakers (192 female, 438 male), sampled at 16 kHz, covering the eight major dialects of American English. Each speaker reads ten phonetically rich sentences. We consider only 462 speakers from TIMIT. We use only 8 sentences for each speaker, the "SX" (5 sentences) and the "SI" (3 sentences). The "SX" sentences are used to train the system, while the "SI" sentences are used to test. The average duration of "SX" sentences is about 4s and test sentences "SI" duration is about of 2-6 seconds. The LibriSpeech database consists in audio books read-out-loud by 2484 speakers, 1283 male and 1201 female volunteers who recorded their voices spontaneously. The speech signal is usually clean, but the recording device and channel conditions vary a lot between different utterances and speakers. 7 utterances have been randomly selected to exploit 12-15 seconds of training for each speaker, and 3 utterances lasting from 5 to 12 seconds as a fixed test set for evaluation. Experiments are only conducted on the short length conditions: our system uses raw waveforms of length of 2 and 4 seconds as input for training and testing phases. Speech utterances having initial duration upper to 4s are splitted on small speech chunk of 2s or 4s. Then utterances having duration lower than 2s or 4s are augmented with speech chunks of same speaker to reach desired duration. To validate the effectiveness of our model, we built 8 kHz and 16 kHz versions of our system. The Timit and Librispeech datasets are thus downsampled to 8 kHz. We did not apply any pre-processing to the raw waveforms, such as pre-emphasis, silence removal, detection and removal of unvoiced speech. Non speech intervals at the beginning and end of each sentence were conserved. Scattering transform was computed to the depth of 2. The first layer contained 8 Gabor wavelets per octave and the second had one Morlet wavelet per octave. The averaging window was set to 500ms of length. Later, coefficients are normalized and log-transformed. The number of the coefficients of a single frame for this setting is 77 and 356 for the first and the seconds orders of the scattering transform respectively. Stochastic gradient descent was used as an optimizer with a learning rate of 0.001 and 0.9 momentum. The network is trained with mini batches of 64 for 30 epochs. Our implementation is based on Scatnet [18] and deep learning Matlab toolboxes.

TABLE II
Number of parameters and epochs for our system and related systems.

	SincNet	**CNN**	**Proposed**
Parameters $\times 10^6$	26,5	27,6	**18,1**
Epochs	2900	2900	**30**

B. Related systems

In this paper, we compare our system with SincNet [29], CNN-Raw [30] and deep system based on hand-crafted features for speaker identification. SincNet is a novel end-to-end neural network architecture, that directly receives raw waveforms as inputs. The first 1D convolutional layer of SincNet is composed by Sinc functions. SincNet convolves the waveform with a set of parametric sinc functions that implement band-pass filters. The filters are initialized using the Mel-frequency filter bank and their low and high cutoff frequencies are adapted with standard back-propagation as any other layer. The first layer performs Sinc based convolutions, using 80 filters of length 251. The remaining two layers use 60 filters of length 5. Next, three fully-connected layers composed of 2048 neurons and normalized with batch normalization are applied. All hidden layers use leaky-ReLU non-linearity. Frame-level binary classification is performed by applying a softmax classifier and cross-entropy criteria [29]. The number of parameters in Sincnet is about 26.5 millions.

In the CNN-Raw system, the raw waveform is fed directly to the first layer. This network sets on the same architecture as SincNet, but the sinc-based convolution layer is remplaced with standard convolution layer. Three convolution layers are used to perform feature mapping. Each convolution layer is composed of 80 filters followed by a max pooling. Next, three fully-connected layers composed of 2048 neurons and normalized with batch normalization are applied. All hidden layers use leaky-ReLU non-linearities. Frame-level binary classification is performed by applying a softmax classifier and cross-entropy criteria [30]. The number of parameters in CNN-raw is about 27.6 millions.

The first row of Table II summarizes the number of learning parameters of all tested methods. We observe that the number of learning parameters required by our method is lower than the ones of SincNet and CNN-Raw by about 33%. The second raw of Table II shows the number of epoch required by each method on the Librispeech dataset. Our architecture requires only 30 epochs for training while both SincNet and CNN-Raw networks require 2900 epochs.

A comparison with popular hand-crafted features was also performed. To this end, we computed 39 MFCCs and 40

FBANKs. These features, were computed every 25 ms with 10 ms overlap. FBANK features were fed to the same CNN architecture used in our system (Section III), while a Multi-Layer Perceptron (MLP) was used for MFCCs.

C. Results

In order to evaluate our proposed speaker identification system we use the identification accuracy rate which is equal to the number of correct identification over the number of tests. Accuracy are only computed per frame for all tested methods. In Table III, we report the effect of sampling frequency on system performance. As expected, results show that our system performs better on 16 kHz than 8 kHz data. However for both datasets, correct identification rates with 8K data are smaller than rates with 16k by only 3.5%. Our system remains thus competitive for low sampling frequency rate.

TABLE III
IDENTIFICATION ACCURACY RATE (%) OF THE PROPOSED SPEAKER IDENTIFICATION ON 8K AND 16K DATA- TRAINED AND TESTED WITH 4S OF UTTERANCES DURATION.

	8k	**16k**
LibriSpeech	84.79	88.04
TIMIT	60.86	64.29

TABLE IV
IDENTIFICATION ACCURACY RATE (%) OF THE PROPOSED SPEAKER IDENTIFICATION AND RELATED SYSTEMS ON LIBRISPEECH 16K.

	2s-2s	**4s-4s**
SincNet-raw	66.52	79.33
CNN-raw	58.48	69.82
MFCC-DNN	52.19	61.94
FBANK-CNN	54.83	65.4
Proposed	**79.86**	**88.04**

In order to compare our system with related ones, speech utterances of either 2s or 4s of duration are used for training and testing without any pre-processing for all systems. Results in Table IV, show that our hybrid network outperforms other systems on Librispeech dataset. For speech utterance with 4s duration, our system achieves a relative improvement of about 20% over CNN-raw and 17% over SincNet. Similarly, with speech utterance of 2s our system provides a relative improvement of about 20% over CNN-raw and 13% over SincNet. Also our system with WST outperforms classical hand-crafted features for both speech utterances of 4s and 2s duration. Results on the original SincNet and CNN-Raw systems [29] were conducted on Librispeech with 12-15 seconds of training utterances for each speaker, and 3 utterances lasting from 5 to 16 seconds for evaluation. It is also important to note that our proposed method does not use any pre-processing such as voice or silence detection as this is performed in [29]. Adding such pre-processings could possibly further enhance our results.

We report in Table V the effect of training utterances duration per speaker on performances. We split the training data to obtain a total duration of 8s or 12s per speaker. Full train duration is about 14s. In this experiment, the duration of training and testing utterances are set equal. Hence, a duration of 2s for testing and training with a total of 8s for training induces the use of 4 samples per speaker for training. As shown in Table V, varying the number of samples per speaker and thus the total duration for training induces a variation of only 2% of the accuracy. On the other hand, using 4s duration instead of 2s induces an increase of the accuracy of about 10%. Our system is thus able to construct discriminating speakers models with few number of training data but provides better results with test and train samples of at least 4s.

TABLE V
IDENTIFICATION ACCURACY RATE (%) OF THE PROPOSED SPEAKER IDENTIFICATION ON LIBRISPEECH 16KHZ DATASET TRAINED AND TESTED WITH UTTERANCES OF 4S AND 2S DURATIONS.

	Total train duration		
Test	**8s**	**12s**	**full**
2s	77.16	78.03	79.86
4s	86.27	87.63	88.04

V. CONCLUSION

In this paper, we proposed a speaker identification system that learns speaker discriminating information directly from short raw speech signals using wavelet scattering transform and CNNs. We have shown the effectiveness of this hybrid architecture for speaker identification under difficult conditions where we use both short utterances for testing and a low number of samples (of the same duration) for training. Experiments on Librispeech and Timit databases have shown that our hybrid method yields significant improvements over SincNet, CNN-Raw and classical feature combined with deep learning methods on the Librispeech database. The wavelet scattering transform has reduced instabilities on the CNN first layers. The proposed hybrid architecture has the ability to reduce the required depth and spatial dimension of the deep learning networks. In future works, we plan to extend this work in order to consider variable length speech utterance.

REFERENCES

[1] A. Kanagasundaram, R. Vogt, D. Dean, and S. Sridharan, “i-vector based speaker recognition on short utterances,” Proc. of Interspeech, 2011.

[2] A. Poddar, M. Sahidullah, and G. Saha, “Speaker verification with short utterances: a review of challenges, trends and opportunities,” IET Biometrics, vol. 7, number 3, pp. 91–101, 2018.

[3] D. A. Reynolds, T. F. Quatieri and R. B. Dunn, “Speaker Verification Using Adapted Gaussian Mixture Models,” Digital Signal Processing, vol. 10. Elseiver, 2000, pp.19–41.

[4] N. Dehak, P. J. Kenny, R. Dehak, P. Dumouchel and P. Ouellet, “Front-end factor analysis for speaker verification,” in IEEE Transactions on Audio, Speech, and Language Processing, vol. 19, number 4, 2011, pp. 788–798.

[5] Wei Li, Tianfan Fu and Jie Zhu, “An improved i-vector extraction for speaker verification,” EURASIP Journal on Audio, Speech and Music Processing, 2015, pp. 1–9.

[6] S. J. Prince and J. H. Elder, “Probabilistic linear discriminant analysis for inferences about identity,” Proc. of International Conference on Computer Vision, 2007.

[7] A. Krizhevsky, I. Sutskever and G. E. Hinton, “Imagenet classification with deep convolutional neural networks,” Proc. of Advances in Neural Information Processing Systems, 2012.

[8] R. Collobert and J. Weston, "A unified architecture for natural language processing: Deep neural networks with multitask learning," Proc. of the International Conference on Machine Learning, 2008.

[9] P. Kenny, V. Gupta, T. Stafylakis, P. Ouellet and J. Alam, "Deep neural networks for extracting baum-welch statistics for speaker recognition," Proc. of Speaker Odyssey, 2014.

[10] S. Yaman, J. W. Pelecanos and R. Sarikaya, "Bottleneck features for speaker recognition," Proc. of Speaker Odyssey, pp. 105–108, 2012.

[11] E. Variani, X. Lei, E. McDermott, I. L. Moreno and J. Gonzalez-Dominguez, "Deep neural networks for small foot-print text-dependent speaker verification," Proc. of ICASSP, pp. 4052–4056, 2014.

[12] D. Snyder, D. Garcia-Romero, G. Sell and D. Povey and S. Khudanpur, "X-vectors: Robust dnn embeddings for speaker recognition," Proc. of ICASSP, 2014.

[13] Y. LeCun and Y. Bengio, "Convolutional networks for images, speech, and time series," The hand-book of brain theory and neural networks, vol. 3361, 1995.

[14] O. Abdel-Hamid, M. Abdel-rahman, J. Hui, D. Li, P. Gerald and Y. Dong, "Convolutional neural networks for speech recognition," IEEE Transactions on Audio, Signal, and Language Processing, vol. 22, number 10, pp. 1533–1545, 2014.

[15] C. Zhang, K. Koishida and J. Hansen, "Text-independent speaker verification based on triplet convolutional neural network embedding," IEEE Transactions on Audio, Signal, and Language Processing, vol. 26, number 9, pp. 4052–4056, 2018.

[16] A. Nagrani, J. S. Chung and A. Zisserman, "Voxceleb:a large-scale speaker identification dataset," Proc. of Interspeech, 2017.

[17] H. Muckenhirn, M. Magimai-Doss and S. Marcel, "To-wards directly modeling raw speech signal for speaker verification using CNNs," Proc. of ICASSP, 2018.

[18] J. Andén, S. Mallat, "Deep scattering spectrum," IEEE Transactions on Signal Processing, vol. 62, number 16, pp. 4114–4128, 2014.

[19] V. Chudáček, J. Andén, S. Mallat, and P. Abry, and M. Doret, "Scattering transform for intrapartum fetal heart rate variability fractal analysis: A case-control study," IEEE Transactions on Biomedical Engineering, vol. 61, number 4, pp. 1100–1108, 2014.

[20] S. Mallat, "Group invariant scattering," Communications on Pure and Applied Mathematics, vol. 65, number 10, pp. 1331–1398, 2012.

[21] V. Peddinti, T. N. Sainath, S. Maymon, B. Ramabhadran, D. Nahamoo, and V. Goel, "Deep scattering spectrum with deep neural network," Proc. of ICASSP, pp. 361–364, 2014.

[22] M. Versteegh, and R. Thiollière, and T. Schatz, and X. Nga Cao, and X. Anguera, and A. Jansen, and E. Dupoux, "Deep scattering spectrum with deep neural networks," Proc. of Interspeech, pp. 55, 2015.

[23] N. Zeghidour, and G. Synnaeve, and M. Versteegh, and E. Dupoux, "A deep scattering spectrum—Deep Siamese network pipeline for unsupervised acoustic modeling," Proc. of ICASSP, pp. 4965–4969, 2016.

[24] J. Bruna, and S. Mallat, , "Invariant scattering convolution networks," IEEE transactions on pattern analysis and machine intelligence, vol. 35, number 8, pp. 1872–1886, 2013.

[25] J. Bruna, A. Szlam, and Y. LeCun, "Learning stable group invariant representations with convolutional networks," arXiv preprint arXiv:1301.3537, 2013.

[26] E. Oyallon,E. Belilovsky, and S. Zagoruyko, "Scaling the Scattering Transform: Deep Hybrid Networks," arXiv preprint arXiv:1703.08961, 2017.

[27] L. Lamel, and R. Kassel, and S. Seneff, "Speech Database Development: Design and Analysis of the Acoustic-Phonetic Corpus," Proc. of DARPA Speech Recognition Work-shop, 1986.

[28] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: An ASR corpus based on public domain audio books," Proc. of ICASSP, pp. 5206–5210, 2015.

[29] M. Ravanelli and Y. Bengio, "Speaker Recognition from raw waveform with SincNet," Proc. of SLT, 2018.

[30] H. Muckenhirn, M. Magimai-Doss, and S. Marcel, "On Learning Vocal Tract System Related Speaker Discriminative Information from Raw Signal Using CNNs," Proc. of Interspeech, 2018.