The DKU-LENOVO Systems for the INTERSPEECH 2019 Computational Paralinguistic Challenge

Haiwei Wu[1,2], *Weiqing Wang*[1], *Ming Li*[1]

[1] Data Science Research Center, Duke Kunshan University, Kunshan, China
[2] School of Electronics and Information Technology, Sun Yat-sen University, Guangzhou, China
ml442@duke.edu

Abstract

This paper introduces our approaches for the orca activity and continuous sleepiness tasks in the Interspeech ComParE Challenge 2019. For the orca activity detection task, we extract deep embeddings using several deep convolutional neural networks, followed by the Support Vector Machine (SVM) based back end classifier. Both STFT spectrogram and log mel-spectrogram are explored as input features. To increase the size of training data and deal with the data imbalance, we propose four kinds of data augmentation. We also investigate the different ways of fusion for multi-channel input data. Besides the official baseline system, to better evaluate the performance of our deep embedding system, we employ the Fisher Vector (FV) encoding on various kinds of acoustic features as an alternative baseline. Experimental results show that our proposed methods significantly outperform the baselines and achieve 0.948 AUC and 0.365 Spearman's Correlation Coefficient on the orca activity and continuous sleepiness evaluation data, respectively.

Index Terms: ComParE challenge, deep neural networks, data augmentation, fisher vector

1. Introduction

The goal of paralinguistic speech attribute recognition [1] is to classify the paralinguistic attributes of audio data automatically. This technology can be applied in many different applications, such as affective computing, disease detection, and various kinds of interdisciplinary studies [1, 2]. In this year, the Interspeech 2019 Computational Paralinguistic Challenge (ComParE) introduces new attributes, namely Styrian Dialects, Continuous Sleepiness, Baby Sounds, and Orca Activity.

In the baselines provided by the challenge organizers, the OpenSMILE acoustic feature set [3], Bag-of-Audio-Word (BoAW) features [4], and AuDeep representations [5, 6] are extracted for the subsequent Support Vector Machine (SVM) based classifier [7]. We extract the ComParE acoustic feature set by applying functionals on low-level descriptors (LLDs) with OpenSMILE [3] which is proven very effective in many paralinguistic recognition tasks. To extract BoAW features, we quantize the audio signals with a codebook learned from LLDs at the beginning. After that, signals can be represented by histograms of their acoustic LLDs. The baseline system generates codebooks of different sizes to find the optimal setting. AuDeep features are extracted by recurrent sequence-to-sequence autoencoders in an unsupervised manner.

In addition to the features given by the baseline system, we also introduce features extracted by the Fisher Vector (FV) encoding method [8, 9]. Researchers have successfully applied FV encoding in many paralinguistic tasks [10, 11]. For the orca activity detection and the continuous sleepiness tasks, we adopt FV encoding on various kinds of acoustic features, including MFCC (Mel Frequency Cepstral Coefficient), LFCC (Linear Frequency Cepstral Coefficient), IMFCC (Inverted Mel Frequency Cepstral Coefficient), MGDF (Modified Group Delay Function), and PLP (Perceptual Linear Prediction) [12, 13, 14].

Previous works show that convolutional neural networks trained with the STFT spectrogram inputs achieved good performance on many paralinguistic problems [15, 16, 17]. The concept of deep embedding has also been widely used in many tasks such as speaker verification [18, 19] and language identification [18, 20]. However, deep learning based methods may suffer from over-fitting due to small scale and imbalance training data [1]. And applying SVM on top of these deep embeddings is shown to be effective and robust against over-fitting on some paralinguistic speech attribute recognition tasks [21].

In this work, we propose a deep embedding system with SVM based back end classifier to detect paralinguistic attributes. The orca activity detection is a binary classification task with a large scale multi-channel training database, which makes it possible for us to address the problem with deep neural networks. First, we extract STFT spectrogram and log mel-spectrogram as the inputs of our systems. Second, we employ several data augmentation methods to further increase the size of training samples as well as reduce the data imbalance. To extract the deep embeddings, we train three deep convolutional neural networks, containing ResNet [22], Inception [23] and DenseNet [24]. In this way, we can obtain discriminative utterance-level embeddings for each signal. SVM is employed for classification in our approach. In this task, official multi-channel audio files enable us to compare different levels of fusion strategies in terms of channels.

The rest of this paper is organized as follows. In Section 2, we introduce the Fisher Vector encoding method on various kinds of acoustic features. In Section 3, we present our end-to-end deep learning system, including feature extraction, data augmentation, embedding extraction, and multi-channel fusion schemes. Experimental results and discussion are provided in Section 4 followed by the conclusion in Section 5.

2. Fisher Vector Encoding

Fisher Vector encoding (FV) [8, 9] is a widely used representation method for image classification. In recent years, it has been successfully applied in several paralinguistic speech attribute recognition problems [10, 11]. In this paper, we employ FV encoding on several kinds of acoustic features. Different from BoAW representation, FV encoding makes use of the first and second statistics of input features which are informative. Compared to the GMM mean-supervector method [25], the FV Encoding extracts both the mean and the variance on each Gaussian component. To classify these utterance-level representations, We apply the SVM and GBDT (Gradient Boosting Decision Tree) algorithm.

To acquire FV encodings [8, 9], in the first step, we extract the acoustic features and train a K-component GMM (Gaussian Mixture Model). The GMM model can be parameterized as $\{\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, \pi_k\}_{k=1}^{K}$. $\boldsymbol{\mu}_k$, $\boldsymbol{\Sigma}_k$, π_k represent the mean vector, covariance matrix and weight of the k^{th} component of the GMM model. Then, according to the GMM, we are able to represent the occupancy probability $\boldsymbol{q}_k(t)$ of a given feature $\boldsymbol{x}_t$ by

$$\boldsymbol{q}_k(t) = \frac{\pi_k \text{exp}\{-\frac{1}{2}(\boldsymbol{x}_t - \boldsymbol{\mu}_k)^T \boldsymbol{\Sigma}_k^{-1}(\boldsymbol{x}_t - \boldsymbol{\mu}_k)\}}{\sum_{i=1}^{K} \pi_i \text{exp}\{-\frac{1}{2}(\boldsymbol{x}_t - \boldsymbol{\mu}_i)^T \boldsymbol{\Sigma}_i^{-1}(\boldsymbol{x}_t - \boldsymbol{\mu}_i)\}}. \quad (1)$$

Next, we compute the derivations of means and covariances of the probability function (1) resulting:

$$\boldsymbol{u}_k^d = \frac{1}{T\sqrt{\pi_k}} \sum_{t=1}^{T} \boldsymbol{q}_k(t) [\frac{\boldsymbol{x}_t^d - \boldsymbol{\mu}_k^d}{\boldsymbol{\sigma}_k^d}], \quad (2)$$

$$\boldsymbol{v}_k^d = \frac{1}{T\sqrt{2\pi_k}} \sum_{t=1}^{T} \boldsymbol{q}_k(t) [(\frac{\boldsymbol{x}_t^d - \boldsymbol{\mu}_k^d}{\boldsymbol{\sigma}_k^d})^2 - 1]. \quad (3)$$

T represents the number of frames. Diagonal covariances are computed with $\boldsymbol{\sigma}_i^2 = \text{diag}(\boldsymbol{\Sigma}_i)$. The number of GMM components is K, and k refers to the k^{th} component. The dimension of the acoustic features is D, and d denotes the d^{th} dimension of a vector.

Finally, a sequence of acoustic features can be represented by a fixed dimensional vector with size $2 \times K \times D$:

$$\boldsymbol{\Phi} = [\cdots \boldsymbol{u}_k^T \cdots \boldsymbol{v}_k^T \cdots]^T. \quad (4)$$

Traditionally, We use MFCC and PLP Cepstrum/Spectrum to train the GMM for FV encoding [10, 11]. Here, we further explore more acoustic features including IMFCC [12], LFCC [12], and MGDF [13]. Besides training a GMM separately for each type of acoustic features, we concatenate different kinds of features on the feature level and train a single large dimensional GMM for FV encoding. We employ both the SVM and the GBDT algorithm for the back end classifiers. [26].

3. End-to-End Deep Learning Based System

This section introduces the details of our deep embedding system for detecting the orca activity. First, spectral features are extracted. Second, we augment the training data using four different schemes. Then, we employed three different deep neural network structures to obtain the embeddings followed by the SVM for classification. Finally, multi-channel information fusion methods are performed on different levels. The illustration of the proposed framework shows in Figure 1.

3.1. Feature Extraction

Short Time Fourier Transform (STFT) spectrum is a common choice as the input of CNN network [20, 27].

For variable STFT spectrograms, we adopt the structure of the Global Average Pooling (GAP) layer [28]. The GAP layer can deal with variable lengths in both the time and frequency axis, allowing batches to have different sizes of the feature map. In the evaluation phase, we do not have to resize the input features. This structure can extract the utterance-level representations for audio signals with different durations, which is suitable for utterance-level audio classification [20, 28].

Inspired by the bird sound detection and bird species classification works [29], we select the log mel-spectrogram as our input feature for the reason that animal sounds could share some common patterns. The feature used for detecting birds can also be effective for detecting orca [29].

We extract the log mel-spectrogram following the process described in [29]. First, we apply STFT on audio signals to obtain the power spectrum. Then, the linear spectrogram is transformed into mel-spectrogram. Next, we normalize the mel-spectrogram and convert it to decibel units. At last, we resize the log mel-spectrogram to a fixed size using Lanczos filter [29, 30].

3.2. Data augmentation

Data augmentation is a common approach when training classification models with an insufficient amount of training data. By data augmentation, we manage to reduce the data imbalance problem as well as enhance the system's performance.

Traditionally, data augmentation is performed by adding external noises to positive samples. In this task, however, we already have the real environmental noises as negative samples, which we can utilize directly. We use several different methods of data augmentation, and all of them are carried out on the time domain.

For each authentic orca audio sample, we randomly select another sample in the noise set or the orca set to add on top of it. And we augment negative samples by adding two random noise signals together. The additional file's amplitude is assigned with an arbitrary weight between 0 and 0.5. Since the lengths are variable, we perform clipping and repeating. Besides these methods, we also apply speed perturbation with factors 0.9, 1.0, and 1.1.

3.3. Embedding

Deep convolutional neural network plays an important role in many areas, including paralinguistic speech attribute recognition in recent years [15, 21]. The convolutional structure can capture the patterns on the images, and more generally, on spectrograms or other time-frequency representations.

The deep convolutional neural network is considered as a local pattern extractor [21]. Here, instead of directly using the original fully-connected layer for classification, we adopt the SVM as our classifier.

In our work, we train ResNet, Inception, and DenseNet as our deep embedding extractors. ResNet [22] learns residual functions concerning the layer inputs, which reduces the difficulty of training a large neural network. Inception [23] network consists of several well-designed Inception modules aiming to reduce the parameters and best utilize computational resources. DenseNet [24] connects every layer with other layers in a feedforward manner thus has the potential to reduce the problem of gradient vanishing. Compared to other network structures applied in previous paralinguistic challenges [15, 21], the networks we adopt here are much deeper. Therefore we focus on the orca activity detection task due to the relatively large size of training data.

3.4. Fusion on channels

The orca activity detection dataset contains multi-channel signals. Four or eight hydrophones collect the underwater signals in towed-array on a 15-meter trimaran from different directions and distances [1]. Intuitively, designing a system using multi-

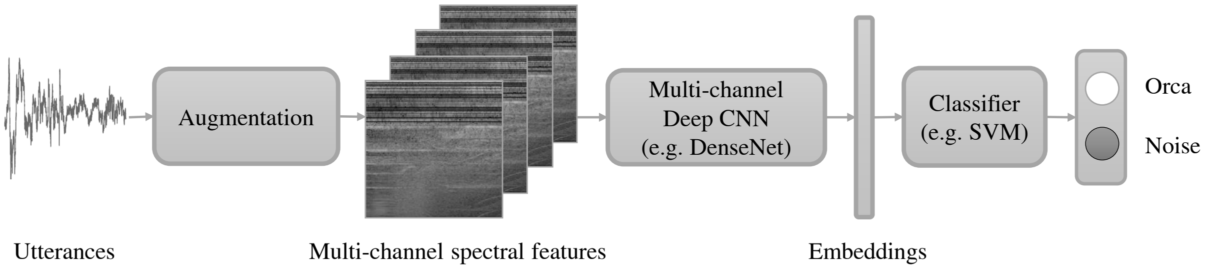

Figure 1: *Structure of the proposed deep embedding system for orca activity detection.*

channel data would have a better performance than the ones built upon single channel data.

Inspired by [31], we investigate the multi-channel fusion strategies on different levels using four channels of signals. The first one is the multi-channel feature-level fusion. We extract spectral features on each channel and feed the multi-channel spectrograms directly into the multi-channel CNN networks. The second level is the embedding level. We train four separate models for embedding extraction, take the mean of the four embeddings, and then use the SVM as the back end classifier. For score-level fusion, we fuse the SVM scores for the four subsystems (each subsystem is for one channel).

4. Experiments

For the orca activity task, our goal is to build a robust binary classification system to distinguish between environmental noise and orca sounds. Area Under the Receiver Operating Characteristic Curve (AUC) is used as the metric. For the continuous sleepiness task, we need to estimate the level of sleepiness. In this task, Spearman's Correlation Coefficient is utilized as the metric.

The structure of this section is as follows. First, experiments setup including feature extraction, as well as model training are presented in 4.1. Second, we compare and analyze the results on different features, classifiers, augmentation schemes, and fusion methods. Lastly, fused scores on the development and evaluation set are shared in 4.6.

If not explicitly mentioned, in this section, we augment audio signals with approaches in Section 3.2 and extract fixed size log mel-spectrogram as input features. In general, SVM back end, DenseNet, and multi-channel feature-level fusion are adopted as our best single system.

4.1. Experiments setup

In the official baseline, ComPare Acoustic Feature set, BoAW presentations, AuDeep features are extracted by OpenSMILE [3], OpenXBOW [4], and Audeep toolkit [5, 6].

For the orca activity task, each audio file is down-sampled from 44100 Hz to 22050 Hz at the very beginning. We implement the FV encoding with vlfeat library [32] on the MATLAB platform. And MFCC, IMFCC, LFCC, MGDF, PLP/RASTA-PLP spectrum, and cepstrum all have 20 coefficients respectively. We compute delta and delta-delta coefficients for these features. The number of GMM components used for FV encoding is selected to be 128 and the resulted dimensionality of Fisher Vectors is $2 \times 60 \times 128 = 15360$.

To classify the baseline and FV encoding representations, we employ SVM with scikit-learn and GBDT algorithm with LightGBM [26]. As for GBDT, the L1 loss and binary cross entropy loss are chosen for the continuous sleepiness and orca activity task, respectively.

Table 1: *Comparison of different deep embeddings and classifiers on the development set*

Model	FC	SVM
ResNet	0.8408	0.923
Inception	0.8403	0.927
DenseNet	0.8430	**0.934**

In the deep embedding system, STFT spectrogram and log mel-spectrogram are extracted with librosa [33] and scipy library [34]. As for the STFT spectrogram, we apply a 25 ms sliding window with a step of 10 ms and employ 1024-points FFT on each frame. The STFT spectrogram is then normalized by mean subtraction.

For the log mel-spectrogram, we apply 256 mel-filters on the STFT spectrogram, and the output power spectrogram is then converted to decibel units. Furthermore, for each audio file, the log mel-spectrogram is resized to a fixed shape of (299, 299).

Three kinds of convolutional neural networks, including ResNet, Inception, and DenseNet are explored. The structures follow the Pytorch [35] implementation of ResNet34, InceptionV3, and DenseNet121. Categorical cross entropy is taken as the loss function. Networks are optimized using Stochastic Gradient Descent (SGD) with Nesterov momentum 0.9. During the training process, the learning rate is first initialized as 0.01 and reduced by a factor of 10 every 12 epochs. We train each CNN network for 30 epochs. After training, we directly extract the embeddings from the penultimate layer of neural networks.

4.2. Results on classifiers

We extract deep embeddings from three neural networks. From Table 1, we can see that compared to the original fully-connected layer (FC), SVM performs much better. In this task, SVM is more robust and less likely to be over-fitting than the FC layer based classifiers in the end-to-end system. Comparing the performances of different networks, we can also find that DenseNet performs the best.

4.3. Results on features

We can notice in Table 2 that the log mel-spectrogram achieves better results than the STFT spectrogram. Log mel-spectrogram can capture the patterns of bird sounds as well as orca sounds [29].

Table 2: *Comparison of different features on the development set*

Model	Log mel-spec	STFT-spec
ResNet	0.923	0.900
Inception	0.927	0.917
DenseNet	**0.934**	0.912

4.4. Results on data augmentation

Table 3: *Comparison of different data augmentation methods on the development set*

Methods	AUC
Origin	0.925
Orca + Noise → Orca	0.929
Orca + Orca → Orca	0.932
Noise + Noise → Noise	0.928
Speed perturbation	0.932
All the aforementioned methods	**0.934**

Table 3 shows that all four data augmentation methods help improve performance. It is worth noting that adding positive samples to other positive samples seems to be a practical approach. Speed perturbation is also useful for improving performance. Increasing negative samples has a relatively minor impact on the final results.

4.5. Results on different fusion methods

Table 4: *Comparison of different levels of multi-channel fusion on the development set with log mel-spectrogram input and SVM back end classifier*

Methods	DenseNet	Inception
Single channel (best)	0.926	0.924
Feature-level fusion (multi-channel CNN)	0.934	0.927
Embedding-level fusion	0.935	0.933
Score-level fusion	0.932	0.933

In our experiments, we investigate three levels of multi-channel fusions. To fuse in the feature-level, we directly feed the multi-channel spectral features into the CNN model. And for embedding-level fusion, the four embeddings are averaged as the input of SVM. As for score-level fusion, we compute the mean of the four SVM scores.

We find that the results between these three methods are similar as shown in Table 4. Embedding-level fusion shows a slightly better result. It is worth mentioning that the scheme of multi-channel feature-level fusion consumes fewer computational resources than the score-level and embedding-level method which require generating deep embeddings separately. Multi-channel feature-level fusion makes more sense when the computational resources are limited.

4.6. Comparison with the baseline

In this part, we compare our proposed methods with the baseline systems. Results of both the development and evaluation set are illustrated in Table 5.

For the official baseline, we select the scores generated with ComParE acoustic feature set, BoAW, and AuDeep features. And the results of the FV system are produced with Fisher Vectors encoding MFCC, LFCC, IMFCC, MGDF, and spectrum/cepstrum of PLP/RASTA-PLP [14]. For both official and FV representations, we adopt the SVM and GBDT algorithm to perform classification. The official baseline + FV utilize both their final output scores. Our results of deep embedding system are obtained by fusing the output scores of models considering different features, classifiers, augmentation schemes, and multi-channel fusion methods.

Table 5: *Comparison with the baseline*

Orca Activity (AUC)	Devel	Test
Official baseline [1]	0.817	0.866 [1]
FV system	0.876	–
Official baseline + FV	0.880	–
Deep embedding with SVM	0.945	**0.948**
Deep embedding with SVM + FV	**0.946**	–
Continuous Sleepiness (Spearman)	**Devel**	**Test**
Official baseline [1]	0.308	0.343 [1]
FV system	0.316	–
Official baseline + FV	**0.326**	**0.365**

The scores on development set generated with different classifiers and representations are fused with weights, and challenge organizers provide the baseline results on the evaluation set [1].

In the tasks of both orca activity and continuous sleepiness, FV features show better performances than baseline representations, which prove the effectiveness of the FV encoding method. For orca activity detection, the deep embedding with SVM system outperforms the baseline on both the development and evaluation set, which means our proposed algorithm is robust and effective.

5. Conclusions

In this work, We introduce the Fisher Vector encoding scheme for the continuous sleepiness and orca activity task in the Interspeech ComParE2019 Challenge. For orca activity detection, we further propose a deep embedding with SVM system. Our work focuses on feature extraction, classifier design, data augmentation and fusion with multi-channel inputs. We find that the log mel-spectrogram shows a better performance than the traditional STFT spectrogram. SVM is proven to be more effective than the original fully-connected layer. Data augmentation by adding training samples and speed perturbation help improve the results. Finally, the performances of the three multi-channel fusion methods show little differences, while the multi-channel feature-level fusion requires fewer computational resources than others. With these systems, we manage to significantly outperform the baseline on both the development and evaluation set.

6. Acknowledgement

This research was funded in part by the National Natural Science Foundation of China (61773413), Natural Science Foundation of Guangzhou City (201707010363), Six Talent Peaks project in Jiangsu Province (JY-074), Science and Technology Program of Guangzhou City (201903010040) and Lenovo. We thank Lin Yang, Xuyang Wang, Junjie Wang, Yulong Liang and yingjie Li from AI lab of Lenovo research who provided insight and expertise that greatly assisted this research.

7. References

[1] B. W. Schuller, A. Batliner, C. Bergler, F. B. Pokorny, J. Krajewski, M. Cychosz, R. Vollmann, S.-D. Roelen, S. Schnieder, E. Bergelson10 *et al.*, "The interspeech 2019 computational paralinguistics challenge: Styrian dialects, continuous sleepiness, baby sounds & orca activity," in *Proc. INTERSPEECH*, 2019.

[2] B. Schuller, S. Steidl, A. Batliner, A. Vinciarelli, K. Scherer, F. Ringeval, M. Chetouani, F. Weninger, F. Eyben, E. Marchi *et al.*, "The interspeech 2013 computational paralinguistics challenge: social signals, conflict, emotion, autism," in *Proc. INTERSPEECH*, 2013, pp. 148–152.

[3] F. Eyben, M. Wöllmer, and B. Schuller, "Opensmile: the munich versatile and fast open-source audio feature extractor," in *Proc. ACM Multimedia*, 2010, pp. 1459–1462.

[4] M. Schmitt and B. Schuller, "Openxbow: introducing the passau open-source crossmodal bag-of-words toolkit," *The Journal of Machine Learning Research*, vol. 18, no. 1, pp. 3370–3374, 2017.

[5] M. Freitag, S. Amiriparian, S. Pugachevskiy, N. Cummins, and B. Schuller, "audeep: Unsupervised learning of representations from audio with deep recurrent neural networks," *The Journal of Machine Learning Research*, vol. 18, no. 1, pp. 6340–6344, 2017.

[6] S. Amiriparian, M. Freitag, N. Cummins, and B. Schuller, "Sequence to sequence autoencoders for unsupervised representation learning from audio," in *Proc. DCASE Workshop*, 2017, pp. 17–21.

[7] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[8] F. Perronnin, J. Sánchez, and T. Mensink, "Improving the fisher kernel for large-scale image classification," in *Proc. ECCV*, 2010, pp. 143–156.

[9] F. Perronnin and C. Dance, "Fisher kernels on visual vocabularies for image categorization," in *Proc. CVPR*, 2007, pp. 1–8.

[10] H. Kaya and A. A. Karpov, "Introducing weighted kernel classifiers for handling imbalanced paralinguistic corpora: Snoring, addressee and cold," in *Proc. INTERSPEECH*, 2017, pp. 3527–3531.

[11] Z. S. Syed, J. Schroeter, K. Sidorov, and D. Marshall, "Computational paralinguistics: Automatic assessment of emotions, mood, and behavioural state from acoustics of speech," in *Proc. INTERSPEECH*, 2018, pp. 511–515.

[12] M. Sahidullah, T. Kinnunen, and C. Hanilçi, "A comparison of features for synthetic speech detection," in *Proc. INTERSPEECH*, 2015, pp. 2087–2091.

[13] H. A. Murthy and V. Gadde, "The modified group delay function and its application to phoneme recognition," in *Proc. ICASSP*, vol. 1, 2003, pp. I–68.

[14] D. P. W. Ellis, "PLP and RASTA (and MFCC, and inversion) in Matlab," 2005, online web resource. [Online]. Available: http://www.ee.columbia.edu/ dpwe/resources/matlab/rastamat/

[15] D. Cai, Z. Ni, W. Liu, W. Cai, G. Li, M. Li, D. Cai, Z. Ni, W. Liu, and W. Cai, "End-to-end deep learning framework for speech paralinguistics detection based on perception aware spectrum," in *Proc. INTERSPEECH*, 2017, pp. 3452–3456.

[16] D. Tang, J. Zeng, and M. Li, "An end-to-end deep learning framework for speech emotion recognition of atypical individuals," in *Proc. INTERSPEECH*, 2018, pp. 162–166.

[17] M. Li, D. Tang, J. Zeng, T. Zhou, H. Zhu, B. Chen, and X. Zou, "An automated assessment framework for atypical prosody and stereotyped idiosyncratic phrases related to autism spectrum disorder," *Computer Speech & Language*, vol. 56, pp. 80–94, 2019.

[18] W. Cai, J. Chen, and M. Li, "Exploring the encoding layer and loss function in end-to-end speaker and language recognition system," in *Proc. Odyssey Workshop*, 2018, pp. 74–81.

[19] 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," *arXiv preprint arXiv:1705.02304*, 2017.

[20] W. Cai, Z. Cai, W. Liu, X. Wang, and M. Li, "Insights into end-to-end learning scheme for language identification," in *Proc. ICASSP*, 2018, pp. 5209–5213.

[21] S. Amiriparian, M. Gerczuk, S. Ottl, N. Cummins, M. Freitag, S. Pugachevskiy, A. Baird, and B. W. Schuller, "Snore sound classification using image-based deep spectrum features," in *Proc. INTERSPEECH*, 2017, pp. 3512–3516.

[22] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. CVPR*, 2016, pp. 770–778.

[23] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proc. CVPR*, 2016, pp. 2818–2826.

[24] G. Huang, Z. Liu, L. v. d. Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proc. CVPR*, 2017, pp. 2261–2269.

[25] M. Li, K. J. Han, and S. Narayanan, "Automatic speaker age and gender recognition using acoustic and prosodic level information fusion," *Computer Speech & Language*, vol. 27, no. 1, pp. 151–167, 2013.

[26] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, and T.-Y. Liu, "Lightgbm: A highly efficient gradient boosting decision tree," in *Proc. NIPS*, 2017, pp. 3146–3154.

[27] W. Cai, D. Cai, W. Liu, G. Li, and M. Li, "Countermeasures for automatic speaker verification replay spoofing attack : On data augmentation, feature representation, classification and fusion," in *Proc. INTERSPEECH*, 2017, pp. 17–21.

[28] M. Lin, Q. Chen, and S. Yan, "Network in network," in *Proc. ICLR*, 2014.

[29] M. Lasseck, "Audio-based bird species identification with deep convolutional neural networks," *Working Notes of CLEF*, vol. 2018, 2018.

[30] C. E. Duchon, "Lanczos filtering in one and two dimensions," *Journal of applied meteorology*, vol. 18, no. 8, pp. 1016–1022, 1979.

[31] D. Tavarez, X. Sarasola, A. Alonso, J. Sanchez, L. Serrano, E. Navas, and I. Hernáez, "Exploring fusion methods and feature space for the classification of paralinguistic information," in *Proc. INTERSPEECH*, 2017, pp. 3517–3521.

[32] A. Vedaldi and B. Fulkerson, "VLFeat: An open and portable library of computer vision algorithms," http://www.vlfeat.org/, 2008.

[33] B. McFee, C. Raffel, D. Liang, D. P. Ellis, M. McVicar, E. Battenberg, and O. Nieto, "librosa: Audio and music signal analysis in python," in *Proc. SciPy*, 2015, pp. 18–25.

[34] E. Jones, T. Oliphant, P. Peterson *et al.*, "SciPy: Open source scientific tools for Python." [Online]. Available: http://www.scipy.org/

[35] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer, "Automatic differentiation in pytorch," in *NIPS-W*, 2017.