Odyssey 2020 The Speaker and Language Recognition Workshop
1-5 November 2020, Tokyo, Japan

Zero-Time Windowing Cepstral Coefficients for Dialect Classification

Rashmi Kethireddy [1], *Sudarsana Reddy Kadiri* [2], *Santosh Kesiraju* [1,3], *Suryakanth V. Gangashetty* [1]

[1]Speech Processing Laboratory, International Institute of Information Technology, Hyderabad, India
[2]Department of Signal Processing and Acoustics, Aalto University, Finland
[3]BUT, Speech@FIT, Czech Republic

rashmi.kethireddy@research.iiit.ac.in; sudarsana.kadiri@aalto.fi; kesiraju@fit.vutbr.cz; svg@iiit.ac.in

Abstract

In this paper, we propose to use novel acoustic features, namely zero-time windowing cepstral coefficients (ZTWCC) for dialect classification. ZTWCC features are derived from high resolution spectrum obtained with zero-time windowing (ZTW) method, and were shown to be useful for discriminating speech sound characteristics effectively as compared to a DFT spectrum. Our proposed system is based on i-vectors trained on static and shifted delta coefficients of ZTWCC. The i-vectors are further whitened before classification. The proposed system is compared with i-vector baseline system trained on Mel frequency cepstral coefficient (MFCC) features. Classification results on STYRIALECT database (German) and UT-Podcast (English) database revealed that the system with proposed features outperformed aforementioned baseline system. Our detailed experimental analysis on dialect classification shows that the i-vector system can indeed exploit high spectral resolution of ZTWCC and hence performed better than MFCC features based system.

1. Introduction

Speech in a language can vary in pronunciation, vocabulary, and grammar based on the geographical spread. These systematic variations in speech due to regional diffusion are termed as dialect. Determining the dialect of the speaker from the speech signal is the dialect classification problem. The applications of automatic dialect classification include personalized computer assistant which adapts to user's dialect. Also, the dialect information can be used to improve the performance of automatic speech recognition (ASR) and speaker recognition systems [1, 2]. The origin of the speaker can be determined by dialect classification and this information is useful for profiling and forensics [3].

Dialect classification is similar to language identification, however, the distribution of phones and allophones across dialects is relatively smaller than across languages. This makes dialect classification rather more challenging. Majority of the methods for dialect classification are borrowed from language identification [4–7]. Previous studies on dialect classification can be categorized into two areas: Studies in first category focused on dialect discriminant feature extraction from speech signal. For example, studies such as [8–10] were focused on exploring the temporal and spectral characteristics across dialects. The features can be further categorized into two; i.e., acoustic or phonotactic based features. Acoustic features usually represent characteristics of speech signal in time or spectral domain, while phonotactic-based [5,6,11,12] features are discrete and capture the distribution of phoneme sequences. In [13], the characteristics of sound sequence are captured from the spectral features using stochastic trajectory model. For acoustic-based features, static Mel frequency cepstral coefficients (MFCC) along with shifted delta cepstral (SDC) features of MFCC are widely used [4, 14].

Studies in the second category focuses on finding the best preprocessing methods which can find distribution of features, decorrelate, and compress the features by retaining the components which are non-overlapping across dialects. Initially, Gaussian mixture model–universal background model (GMM-UBM) based preprocessing methods were used to extract supervectors from both the acoustic and phonotactic based features [15, 16]. Later, i-vectors were introduced to convert high dimensional supervectors to low dimension i-vectors. Using i-vector method as backend preprocessor has improved the performance enormously [4,17–19] for dialect classification. Classifiers such as support vector machine (SVM), linear discriminant analysis (LDA) and its variants such as QDA (quadratic discriminant analysis), PLDA (probabilistic linear discriminant analysis), HLDA (heteroscedastic linear discriminant analysis) were also used [13,19–21] with i-vector based features. In [22], convolutional neural networks (CNN) were explored for dialect classification over phonotactic features.

In [17], spectral features in the i-vector approach were replaced by speech attributes such as manner of articulation and place of articulation. This approach reduced the

10.21437/Odyssey.2020-5

relative error rate significantly as compared to MFCC i-vector based approach. This shows that there is a need for better features for dialect processing that can differentiate different speech sound characteristics.

In this paper, we use the high resolution spectrum provided by the zero-time windowing (ZTW) method that can differentiate different speech sound characteristics effectively compared to the DFT spectrum [23–25]. In order to capture the articulation/sound characteristic variations, we propose to use cepstral coefficients derived from ZTW spectrum and they are referred as ZTWCC. We further use ZTWCC features for training an i-vector system. The extracted i-vectors are then fed into a classifier for identifying the dialect. We achieve significantly better performance over MFCC based i-vector system.

The organization of this paper is as follows: The zero time windowing (ZTW) method is explained in Section 2, which forms the basis for extracting the desired ZTW cepstral coefficients (ZTWCC). Our system pipeline based on the extracted ZTWCC features is given in Section 3, followed by the experimental details in Section 4. Detailed analysis and comparison of results are presented in Section 5. Finally, Section 6 gives a summary of the study.

2. Extraction of Zero-time windowing cepstral coefficients (ZTWCC)

This section describes the ZTW method and extraction of ZTWCC features from ZTW spectrum.

2.1. Zero-time windowing (ZTW) method

The objective of the ZTW method is to capture the time varying characteristics of the speech production mechanism by deriving the instantaneous spectrum. The ZTW spectrum was shown to be useful for discriminating several speech sounds such as burst, aspiration, nasalized vowels, trill, and a transition from vowel to consonant as compared to DFT spectrum [23, 24].

In this method, the speech signal is windowed with heavily decaying window at each instant of time. This operation highlights the samples at the beginning of the window (near 0^{th} instant), and hence the name zero-time windowing, which provides the higher temporal resolution. The spectrum is estimated using group delay function, which provides higher spectral resolution. The steps involved in deriving ZTW spectrum are as follows:

- The speech signal ($s[n]$) is pre-emphasized to remove the effects of low frequency trend in the signal.
- Speech segment of L ms at each instant is considered. That is, $s[n]$ is defined for $n = 0, 1, \ldots, M-1$, where the number of samples $M = L * f_s/1000$ and f_s is the sampling frequency.
- The segment is multiplied with a heavily decaying window $w_1^2[n]$, where:

$$w_1[n] = \begin{cases} 0, & n = 0 \\ 1/(4\sin^2(\pi n/2N)), & n = 1, \ldots, N-1. \end{cases} \quad (1)$$

  Here N is the number of samples used for DFT ($N >> M$). Multiplying $s[n]$ with window $w_1^2[n]$ is equivalent to four times integration in the frequency domain. Truncation of the signal at the instant $n = M-1$, may result in a ripple effect in the frequency domain. This effect is reduced by using window ($w_2[n]$), which is square of half cosine window.

$$w_2[n] = 4\cos^2(\pi n/2M), n = 0, \ldots M-1. \quad (2)$$

- The spectrum is estimated using the numerator of the group delay (NGD) function ($g[k]$) for the windowed signal (i.e., $x[n] = w_1^2[n]w_2[n]s[n]$) and is given by:

$$g[k] = X_R[k]Y_R[k] + X_I[k]Y_I[k], k = 0, \ldots, N-1, \quad (3)$$

  where $X_R[k]$ and $X_I[k]$ are the real and imaginary parts of the N-point DFT $X[k]$ of $x[n]$. Likewise, $Y_R[k]$ and $Y_I[k]$ are the real and imaginary parts of the N-point DFT $Y[k]$ of $y[n] = nx[n]$.
- The NGD function is double differentiated to highlight the peaks in the spectrum corresponding to formants of the vocal tract system. The Hilbert envelope of the double differenced NGD is called the HNGD/ZTW spectrum, denoted by $X[n, k]$.

2.2. ZTWCC extraction

ZTWCCs are derived from the ZTW spectrum [26, 27]. The cepstrum $c[n, k]$ is given by

$$c[n, k] = \text{IDFT}(\log(X[n, k])). \quad (4)$$

From $c[n, k]$, the first 14 cepstral coefficients are considered (including 0^{th} coefficient) and they are referred as ZTWCCs. Figure 1 shows the block diagram describing the steps involved in the ZTWCC feature extraction.

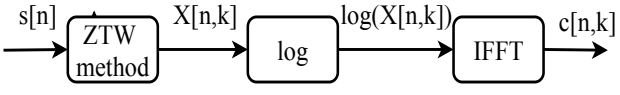

Figure 1: *Block diagram of ZTWCC feature extraction.*

3. Proposed system

This section describes the details of the proposed dialect classification system. Figure 2 shows the block diagram of the proposed system which consists of three stages:

front-end feature extraction, back-end preprocessing, and classification. Feature extraction stage involves extraction of ZTWCC from the ZTW spectrum and extraction of shifted delta coefficients from ZTWCC. This is followed by i-vector extraction and whitening transformation. Finally, predicting dialects from whitened i-vectors using a classifier.

3.1. Feature extraction

3.1.1. Configuration for ZTWCC feature extraction

For ZTW spectrum estimation, a window of length $L = 5$ ms was used with a N-point DFT ($N = 2048$). ZTWCCs are extracted with an interval of 6.25 ms rather than considering every sample, and only first 14 cepstral coefficients are extracted.

3.1.2. Shifted delta cepstra (SDC)

In [14], it was shown that cepstral features vary temporally across dialects. There was a significant improvement in language identification after using SDC features rather than delta and double delta coefficients [14]. SDC features are computed over the ZTWCCs for each frame. N-d-p-K defines the configuration for the SDC computations. At every time instant t, delta computations between cepstral coefficients at $(t + ip - d)^{th}$ and $(t + ip + d)^{th}$ are done. These delta coefficients computed with i varying from 1 to K, and are stacked to get delta coefficients at each instant in time t. SDC vector $\mathbf{\Delta}c(t, i)$ for cepstral coefficients at time t for i^{th} shift is given by:

$$\mathbf{\Delta c(t, i)} = \mathbf{c(t + ip + d)} - \mathbf{c(t + ip - d)}, \quad (5)$$

where N denotes dimension of static cepstral coefficients, d denotes delay or advance from the current frame, p is the shift between consecutive delta computations, and K such delta computations are concatenated to form $N * K$ dimensional SDC coefficients.

In our experiments, the configuration for SDC was set to 14-1-3-7, which resulted in 98 SDCs. Combining both static (14-dimensional ZTWCC) and SDCs (98 dimension) resulted in 112-dimensional feature vector for each frame.

3.2. i-vector system

Factor analysis is a method of expressing the variability of the observed variables (data) in terms of low-dimensional (latent) vectors. I-vector modeling is one of the factor analysis methods to represent low-dimensional total variability factors for each utterance in a single vector [28].

Stacked means of GMM are termed as supervectors. Stacked means of a GMM-UBM are represented by $\mathbf{m}$ and stacked means of an utterance adapted GMM are represented by $\mathbf{M}$. The supervectors of each utterance $\mathbf{M}$ can be expressed by mean components and offset which is given by:

$$\mathbf{M} = \mathbf{m} + \mathbf{Tw}, \quad (6)$$

where $\mathbf{T}$ represents low-rank total variability matrix, $\mathbf{w}$ is utterance specific latent factor vector known as i-vector with a prior distribution of $\mathcal{N}(0, \mathbf{I})$.

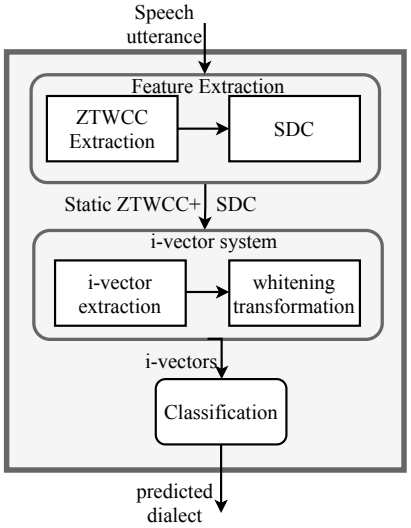

Figure 2: *Block diagram of the proposed system for dialect classification.*

Means and variances of GMM-UBM were initialized using k-means clustering, then UBM is trained using expectation-maximization (EM) algorithm with train data from all dialects. To obtain i-vectors, a process similar to estimation of eigen voice in [29, 30] is followed. In this approach, first Baum-Welch statistics per utterance are accumulated then total variability matrix $\mathbf{T}$ is iteratively trained. Finally, i-vector $\mathbf{w}$ is estimated for each utterance which can be used for classifying dialects.

We experimented with various number of Gaussian mixture components $\in \{256, 512, 1024, 2048\}$ and i-vector dimensions $\in \{100, 200, 300, 400\}$. Both the GMM-UBM and the total variability matrix $\mathbf{T}$ are trained for five iterations. Further, extracted i-vectors are whitened using zero-phase component analysis-correlation. This process decorrelates the variables and align along largest variance by maximally retaining the information [31]. In [7], it is mentioned that whitening transformation contributes to the reduction of channel variability component in the obtained i-vectors. The details of the experiments showing the importance of centering and whitening transformation can be seen in Section 5. Matlab toolbox for speaker recognition (MSR) identity toolkit [32] is used for implementing i-vector framework.

3.3. Classification

The classifiers are trained on the whitened i-vectors. More specifically, we explored support vector machines (SVM), multi-class logistic regression (MCLR), and

Gaussian linear classifier (GLC). The SVM was trained with a linear kernel in one-vs-rest fashion. We used the standard publicly available implementation of SVM and MCLR [33].

4. Experimental setup

This section describes the database, evaluation metrics and baseline systems used for comparison.

4.1. Database description

We have considered two databases with dialects in different languages, so that the proposed system's robustness to language can also be evaluated. The first database is STYRIALECT which has the Styrian dialects of German language. The database contains a total of 9732 utterances, with 5227, 2570, and 1935 utterances in training, development and test sets, respectively. Average duration of utterances is 2 sec, and the sampling frequency of the data is 16 kHz. Database consists of three dialect classes, and the distribution of the classes is uneven in the data splits. More information about the database can be found in [34].

The second database is UT-Podcast which has three major dialects (US, UK, AU) of English [35]. To have variability in vocabulary, it is collected from different websites within each dialect covering wide range of topics. The data is more spontaneous and not very well structured as it is collected from podcast websites. The audio recordings are segmented such that each utterance is 17 sec and contains 46 words on an average. The sampling frequency of the data is 8 kHz. Corpus is divided into train data with 1101 utterances and test data with 661 utterances.

4.2. Evaluation metrics

The primary evaluation metric is the unweighted average recall (UAR) which considers all the classes equally. Additional metrics such as F1-score and accuracy are also reported. F1-score balances between false positives and false negatives which makes it unbiased to the majority class.

4.3. Baseline systems for comparison

The first three baseline systems are provided by the organizers of the ComParE challenge [34]. The next one is a standard i-vector system trained using MFCC features [7, 18]. The configurations for the baseline systems are defined below.

- ComParE-2019 baseline systems have two stages: feature extraction followed by classification using SVM. Three baseline systems were provided: The first uses the "ComParE" acoustic features derived using openSMILE toolkit [36]. The second uses bag-of-audio-words as features derived using openXBOW toolkit [37], and the third uses features extracted from auto-encoder that was trained on spectrogram with the help of AuDeep toolkit [38].
- MFCC i-vector system uses i-vector modeling over MFCC features [7,18,19]. 13-dimensional MFCCs are extracted using a window size of 20 msec and a shift of 10 msec. The mean normalized 13 static MFCC features are then used to obtain 84 SDC features using 13-1-3-7 configuration. The i-vector configuration is similar to the configuration of our proposed system as described in Section 3.2.

5. Results and discussion

Table 1 shows the results of STYRIALECT database using ComParE's baseline (rows $1-3$), MFCC i-vector (rows $4-6$), and proposed ZTWCC (rows $7-9$). From the table, it can be clearly seen that proposed ZTWCC-based systems performed significantly better than ComParE's baselines and MFCC i-vector baselines. Among the three ComParE's baseline systems, AuDeep system performance is the better than others. From the table, it can also be observed that using SDC features and whitening the i-vectors improves the performance of both MFCC (row 6) and the proposed ZTWCC (row 9) systems. Note that both the MFCC and ZTWCC i-vector systems were trained with the same configuration, i.e., 256 Gaussian components and 100 dimensional subspace. Usage of SDC features improved the UAR of MFCC i-vectors by 13.0% (42.6% UAR) and the proposed system by 1.64% (45.5% UAR) when compared to Δ and $\Delta\Delta$ coefficients.

Table 1: Performance (in UAR [%], accuracy [%] and F1-score) of MFCC i-vector system and proposed system by varying SDC and WT with SVM classifier over STYRIALECT corpus (dev).

Feature representation	UAR	Acc.	F1
openSMILE [34]	38.3	55.00	0.386
openXBOW [34]	38.2	58.59	0.353
AuDeep [34]	46.7	**66.00**	0.464
MFCC+Δ, $\Delta\Delta$	37.6	48.32	0.389
MFCC+SDC	42.6	51.12	0.437
MFCC+SDC+WT	45.0	55.78	0.448
ZTWCC+Δ, $\Delta\Delta$	45.2	59.76	0.472
ZTWCC+SDC	45.5	61.05	0.472
ZTWCC+SDC+WT	**49.3**	59.99	**0.487**

Further, usage of whitening transformation improved the UAR by 5.60% (45.0% UAR) and 8.35% (49.3% UAR) with MFCC i-vector (MFCC+SDC+WT) and proposed systems (ZTWCC+SDC+WT), respectively. Overall, it is observed that proposed ZTWCC i-vector with

Table 2: Classwise (NS: NorthenS, US: UrbanS, ES: EasternS) accuracies for ComParE's best baseline, MFCC i-vector baseline and proposed ZTWCC system on STYRIALECT database (dev).

System	NS	US	ES
ComParE's baseline (AuDeep)	0.46	89.22	50.55
MFCC i-vector system	19.48	64.3	49.81
ZTWCC i-vector system	21.11	69.81	56.82

SDC and whitening transformation (WT) outperformed all the variants of baseline systems.

The class-wise performance in terms of accuracy is shown in Table 2. It can be observed that ComParE's baseline results are biased towards the majority class "UrbanS (US)", with much worse performance for "NorthernS (NS)". On the other hand, the proposed system is less biased to the majority class and outperformed both the baseline systems in UAR.

Table 3 shows the performance (in UAR%) of MFCC i-vector system and proposed (ZTWCC) system with various classifiers such as SVM, logistic regression (LR), and Gaussian linear classifier (GLC). The results depicts that both ZTWCC and MFCC based i-vector systems with SVM classifier outperformed other classifiers. Also, it can be observed that the proposed system outperformed MFCC i-vector baseline system in all the classifiers. This improvement in the performance with ZTWCC based i-vector system suggests that i-vectors can exploit the high spectral resolution of ZTWCC features.

Table 3: Performance (in UAR%) of MFCC and ZTWCC i-vector systems with SVM, LR, and GLC classifiers on STYRIALECT database (dev).

System \Classifier	SVM	LR	GLC
MFCC i-vector system	45.0	44.8	44.9
ZTWCC i-vector system	**49.3**	**48.8**	**47.5**

In Figure 3, performance of dialect classification (in UAR) for baseline system (MFCC i-vector system) and proposed system (ZTWCC i-vector system) is shown by varying the Gaussian mixture components and i-vector dimension. From the figure, it is observed that lesser number of GMM components and lower dimension of i-vector resulted in better performance. MFCC i-vector system gave best UAR of 45.1% with 256 Gaussian components and 300-dimensional i-vectors. The proposed system gave the best UAR of 49.3% with 256 Gaussian components and 100-dimensional i-vectors. From the figure, it is also observed that in all the different configurations, ZTWCC-based i-vectors outperformed MFCC-based i-vectors.

The proposed system is also evaluated using UT-Podcast English dialect database and the results for ComParE's baseline, MFCC-based i-vector, and ZTWCC-based i-vector systems are given in Table 4. The results in the table shows that the proposed ZTWCC-based i-vector system outperformed the MFCC based i-vector and ComParE's best baseline systems by 2.4% and 24.8% (relatively) in UAR. This significant improvement in performance suggests the i-vector system can benefit from ZTWCC features which encode high resolution spectral features.

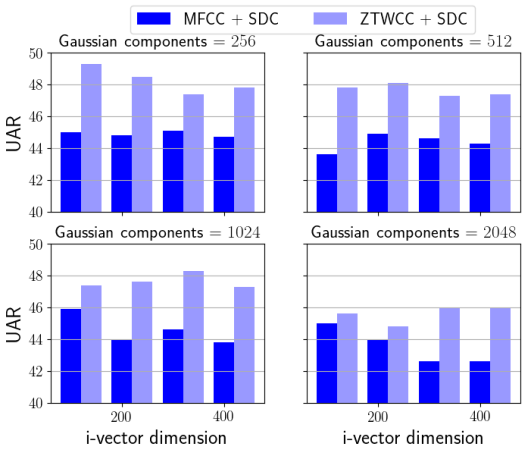

Figure 3: *Performance (in UAR %) of baseline system (MFCC features) and proposed system (ZTWCC features) with varying Gaussian mixture components (256, 512, 1024 and 2048) and i-vector dimension (100, 200, 300 and 400).*

Table 4: Performance (in UAR%, accuracy %, F1-score) of baseline and proposed systems over UT-Podcast (test) database.

System	UAR	Acc.	F1
ComParE's baseline (AuDeep)	60.9	65.3	0.600
MFCC i-vector system	74.2	**79.2**	0.729
ZTWCC i-vector system	**76.0**	78.0	**0.742**

6. Summary

In this paper, we proposed to use ZTWCC features for training i-vector system for dialect classification. Experiments over STYRIALECT and UT-Podcast have shown an improvement of 9.5% and 2.4% in UAR respectively with respect to baseline MFCC features. Our experiments on STYRIALECT and UT-Podcast databases showed that the i-vector system can exploit the high resolution spectral features encoded in ZTWCCs and performed significantly better than standard MFCC based system for dialect classification. Our further analysis showed the importance of SDC features and i-vector whitening which improved the performance of both the baseline and proposed systems.

7. Acknowledgements

The second author would like to thank the Academy of Finland (project no. 312490) for supporting his stay in Finland as a Postdoctoral Researcher.

8. References

[1] V Gupta and P Mermelstein, "Effects of speaker accent on the performance of a speaker-independent, isolated-word recognizer," *J. Acoust. Soc. Am.*, vol. 71, pp. 1581–1587, 1982.

[2] Arlo Faria, "Accent classification for speech recognition," in *Proc. International Workshop on Machine Learning for Multimodal Interaction*, 2005, pp. 285–293.

[3] Fadi Biadsy, *Automatic dialect and accent recognition and its application to speech recognition*, Ph.D. thesis, Columbia University, 2011.

[4] Abualsoud Hanani, Martin J Russell, and Michael J Carey, "Human and computer recognition of regional accents and ethnic groups from British English speech," *Computer Speech & Language*, vol. 27, no. 1, pp. 59–74, 2013.

[5] Maryam Najafian, Saeid Safavi, Phil Weber, and Martin J. Russell, "Identification of British English regional accents using fusion of i-vector and multi-accent phonotactic system," in *Proc. ODYSSEY*, 2016, pp. 132–139.

[6] Fadi Biadsy, Julia Hirschberg, and Nizar Habash, "Spoken Arabic dialect identification using phonotactic modeling," in *Proc. Workshop on Computational Approaches to Semitic Languages*, 2009, pp. 53–61.

[7] Alberto Abad, Eugénio Ribeiro, Fábio Kepler, Ramón Fernández Astudillo, and Isabel Trancoso, "Exploiting phone log-likelihood ratio features for the detection of the native language of non-native English speakers," in *Proc. Interspeech*, 2016, pp. 2413–2417.

[8] John HL Hansen and Levent M Arslan, "Foreign accent classification using source generator based prosodic features," *Proc. Int. Conf. Acoustics Speech and Signal Processing (ICASSP)*, vol. 1, pp. 836–839, 1995.

[9] Levent M Arslan and John HL Hansen, "A study of temporal features and frequency characteristics in American English foreign accent," *J. Acoust. Soc. Am.*, vol. 102, no. 1, pp. 28–40, 1997.

[10] Liu Wai Kat and Pascale Fung, "Fast accent identification and accented speech recognition," in *Proc. Int. Conf. Acoustics Speech and Signal Processing (ICASSP)*, 1999, pp. 221–224.

[11] Marc A. Zissman, Terry P. Gleason, Deborah Rekart, and Beth L. Losiewicz, "Automatic dialect identification of extemporaneous conversational, Latin American Spanish speech," in *Proc. Int. Conf. Acoustics Speech and Signal Processing (ICASSP)*, 1996, pp. 777–780 vol. 2.

[12] Fred S. Richardson, William M. Campbell, and Pedro A. Torres-Carrasquillo, "Discriminative n-gram selection for dialect recognition," in *Proc. Interspeech*, 2009, pp. 192–195.

[13] Pongtep Angkititrakul and John H. L. Hansen, "Advances in phone-based modeling for automatic accent classification," *IEEE Trans. Audio, Speech, and Language Processing*, vol. 14, no. 2, pp. 634–646, 2006.

[14] Pedro A. Torres-Carrasquillo, Elliot Singer, Mary A. Kohler, Richard J. Greene, Douglas A. Reynolds, and J. R. Deller Jr., "Approaches to language identification using Gaussian mixture models and shifted delta cepstral features," in *Proc. Int. Conf. Spoken Language Processing (INTERSPEECH)*, 2002.

[15] Saeid Safavi, Abualsoud Hanani, Martin J. Russell, Peter Jancovic, and Michael J. Carey, "Contrasting the effects of different frequency bands on speaker and accent identification," *IEEE Signal Processing Letters*, vol. 19, no. 12, pp. 829–832, 2012.

[16] Fadi Biadsy, Julia Hirschberg, and Daniel P. W. Ellis, "Dialect and accent recognition using phonetic-segmentation supervectors," in *Proc. Interspeech*, 2011, pp. 745–748.

[17] Hamid Behravan, Ville Hautamäki, Sabato Marco Siniscalchi, Tomi Kinnunen, and Chin-Hui Lee, "i-vector modeling of speech attributes for automatic foreign accent recognition," *IEEE Trans. Audio, Speech, and Language Processing*, vol. 24, no. 1, pp. 29–41, 2016.

[18] Hamid Behravan, Ville Hautamäki, and Tomi Kinnunen, "Foreign accent detection from spoken Finnish using i-vectors," in *Proc. Interspeech*, 2013, pp. 79–83.

[19] Andrea DeMarco and Stephen J. Cox, "Iterative classification of regional British accents in i-vector space," in *Proc. Symposium on Machine Learning in Speech and Language Processing*, 2012, pp. 1–4.

[20] Karsten Kumpf and Robin W. King, "Foreign speaker accent classification using phoneme-dependent accent discrimination models and comparisons with human perception benchmarks," in *Proc. Eurospeech*, 1997.

[21] Hamid Behravan, Ville Hautamäki, and Tomi Kinnunen, "Factors affecting i-vector based foreign accent recognition: A case study in spoken Finnish," *Speech Communication*, vol. 66, pp. 118–129, 2015.

[22] Maryam Najafian, Sameer Khurana, Suwon Shon, Ahmed Ali, and James R. Glass, "Exploiting convolutional neural networks for phonotactic based dialect identification," in *Proc. Int. Conf. Acoustics Speech and Signal Processing (ICASSP)*, 2018, pp. 5174–5178.

[23] Bayya Yegnanarayana and Dhananjaya N. Gowda, "Spectro-temporal analysis of speech signals using zero-time windowing and group delay function," *Speech Communication*, vol. 55, no. 6, pp. 782–795, 2013.

[24] N. Dhananjaya, B. Yegnanarayana, and Peri Bhaskararao, "Acoustic analysis of trill sounds," *J. Acoust. Soc. Am.*, vol. 131, no. 4, pp. 3141–3152, 2012.

[25] N. Dhananjaya, *Signal processing for excitation-based analysis of acoustic events in speech*, Ph.D. thesis, Dept. of Computer Science and Engineering, IIT Madras, Chennai, Oct. 2011.

[26] Sudarsana Reddy Kadiri and Bayya Yegnanarayana, "Breathy to tense voice discrimination using zero-time windowing cepstral coefficients (ztwccs)," in *Proc. INTERSPEECH*, 2018, pp. 232–236.

[27] Sudarsana Reddy Kadiri, Paavo Alku, and B. Yegnanarayana, "Analysis and classification of phonation types in speech and singing voice," *Speech Communication*, vol. 118, pp. 33 – 47, 2020.

[28] Najim Dehak, Patrick Kenny, Réda Dehak, Pierre Dumouchel, and Pierre Ouellet, "Front-end factor analysis for speaker verification," *IEEE Trans. Audio, Speech, and Language Processing*, vol. 19, no. 4, pp. 788–798, 2011.

[29] Patrick Kenny, Pierre Ouellet, Najim Dehak, Vishwa Gupta, and Pierre Dumouchel, "A study of interspeaker variability in speaker verification," *IEEE Trans. Audio, Speech, and Language Processing*, vol. 16, no. 5, pp. 980–988, 2008.

[30] Howard Lei, "Joint factor analysis (jfa) and i-vector tutorial," *ICSI. Web. 02 Oct*, 2011.

[31] Agnan Kessy, Alex Lewin, and Korbinian Strimmer, "Optimal whitening and decorrelation," *The American Statistician*, vol. 72, no. 4, pp. 309–314, 2018.

[32] Seyed Omid Sadjadi, Malcolm Slaney, and Larry P. Heck, "MSR identity toolbox v1.0: A matlab toolbox for speaker recognition research," 2013.

[33] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al., "Scikit-learn: Machine learning in python," *Journal of machine learning research*, vol. 12, pp. 2825–2830, 2011.

[34] B. W. Schuller, A. Batliner, C. Bergler, F. B. Pokorny, J. Krajewski, M. Cychosz, R. Vollmann, S.-D. Roelen, S. Schnieder, E. Bergelson, A. Cristià, A. Seidl, A. Warlaumont, L. Yankowitz, E. Nöth, S. Amiriparian, S. Hantke, and M. Schmitt, "The interspeech 2019 computational paralinguistics challenge: Styrian dialects, continuous sleepiness, baby sounds and orca activity," in *Proc. Interspeech*, 2019.

[35] John H.L. Hansen and Gang Liu, "Unsupervised accent classification for deep data fusion of accent and language information," *Speech Communication*, vol. 78, pp. 19 – 33, 2016.

[36] Florian Eyben, Martin Wöllmer, and Björn Schuller, "Opensmile: The Munich versatile and fast open-source audio feature extractor," in *Proc. ACM International Conference on Multimedia*, 2010, pp. 1459–1462.

[37] Maximilian Schmitt and Björn 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.

[38] Michael Freitag, Shahin Amiriparian, Sergey Pugachevskiy, Nicholas Cummins, and Björn 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.