Development of CRIM System for the Automatic Speaker Verification Spoofing and Countermeasures Challenge 2015

Md Jahangir Alam, Patrick Kenny, Gautam Bhattacharya, Themos Stafylakis

CRIM, Montreal, Quebec, Canada

{jahangir.alam,patrick.kenny,themos.stafylakis,gautam.bhattacharya}@crim.ca

10.21437/Interspeech.2015-469

Abstract

The automatic speaker verification spoofing and countermeasures challenge 2015 provides a common framework for the evaluation of spoofing countermeasures or anti-spoofing techniques in the presence of various seen and unseen spoofing attacks. This contribution proposes a system consisting of amplitude, phase, linear prediction residual, and combined amplitude - phase-based countermeasures for the detection of spoofing attacks. In this task we use following features: Mel-frequency cepstral coefficients (MFCC), product spectrum-based cepstral coefficients, modified group delay cepstral coefficients, weighted linear prediction group delay cepstral coefficients, linear prediction residual cepstral coefficients, cosine normalized phase-based cepstral features (CNPCC), and a combination of MFCC-CNPCC. The product spectrum-based features are influenced by both the amplitude and phase spectra. The Gaussian Mixture Model (GMM) classifier is used for the discrimination of the human and spoofed speech signals. Our primary submitted system is a linear fusion of the sub-systems based on the features mentioned above with fusion weights trained on the development dataset. Experimental results on the challenge evaluation data provided an average EER (equal error rate) of 0.041%, 5.347%, and 2.69% on the known, unknown and all (known + unknown) spoofing attacks, respectively. Among all the systems product spectrum-based cepstral coefficients- and conventional MFCC (without any feature normalization)-based systems performed the best in terms of EER measure. On the known, unknown and all conditions the EER obtained by the MFCC and product spectrum-based features are 0.78% & 0.65%, 5.39% & 5.37% and 3.09% & 3.01%, respectively.

Index Terms: spoofing countermeasures, modified group delay, cosine normalized cepstrum, product spectrum, LP residual cepstrum, GMM

1. Introduction

Speaker verification systems have become increasingly popular in the last few years. In a speaker verification system a binary decision is made for accepting to rejecting a claimed identity based on a speech recording. Speaker recognition systems have been incorporated into a number of forensic, civilian, and commercial applications. Some examples of its applications include computer or smart phone log in, telephone banking, calling cards. Given its widespread usage researchers have analyzed the vulnerability of speaker verification systems to various types of spoofing attacks such as impersonation [1], replay attacks [2], voice conversion [3, 4] and speaker-adapted speech synthesis [5].

Progress in the development of efficient spoofing countermeasures is less advanced in case of automatic speaker verification (ASV) than some other biometric modalities [6]. Some efforts [2, 8-18] have been made by different research groups to develop countermeasures for ASV, mostly for text-dependent ASV, by exploiting prior knowledge of particular spoofing attacks. The ASVspoof 2015 challenge (the first ASV spoofing and countermeasures challenge) [7] provides a common ground with standard corpus, protocols and metrics to facilitate the performance comparison of different spoofing countermeasures against known as well as unknown spoofing attacks.

Most of the successful spoofing countermeasures reported in the literature are based on phase [12-13, 20, 22]. In this ASVspoof 2015 challenge we use amplitude (e.g., MFCC)-, phase (e.g., cosine normalized phase-based cepstral coefficients (CNPCC))-, and combined amplitude - phase (e.g., product spectrum cepstral coefficients, MFCC-CNPCC)-based countermeasures for the detection of spoofing attacks. The standard Gaussian Mixture Model (GMM) classifier is used for this task. Our primary submitted system is based a linear fusion [19] of subsystems with fusion weights estimated from the development test data by logistic regression.

2. Spoofing Countermeasures

In this section we briefly describe the features used in this evaluation as spoofing countermeasures.

2.1. MFCC as Spoofing Countermeasures

In spoofed speech, specifically in synthesized or voice converted speech, the original phase information is lost. Therefore, against spoofing attacks phase-based features tend to outperform the amplitude-based feature, such as MFCC [12-13, 20]. Steps involved in the extraction of MFCC are shown in fig. 1. It has been observed by doing spoofing detection experiment on ASVspoof 2015 development dataset that MFCC feature (without applying any kind of feature normalization) can provide comparable performance to that of the phase (e.g., modified group delay)-based features.

2.2. Normalized Phase-based Countermeasures

A key problem with the phase spectrum is phase wrapping which results in an intractable, noise-like, and chaotic shape lacking any informative trend. This problem can be dealt with phase unwrapping methods [20, 23-25]. Phase unwrapping converts a wrapped phase signal to a continuous phase signal that is free from 2π jumps. After unwrapping, the range of the phase spectrum might vary which makes it difficult to model the phase information. Application of the cosine function on the unwrapped phase normalizes the range into ± 1 [20]. From this normalized phase spectra cepstral coefficients are obtained by applying a DCT (Discrete Cosine Transform). Here, we denote this feature as cosine normalized phase-based cepstral coefficients (CNPCC). Fig. 1 presents the CNPCC extraction block diagram.

2.3. Joint MFCC-CNPCC Countermeasures

MFCC-CNPCC features, as shown in fig. 1, can be obtained by concatenating MFCC and CNPCC features. These features are influenced by both the amplitude and phase spectra.

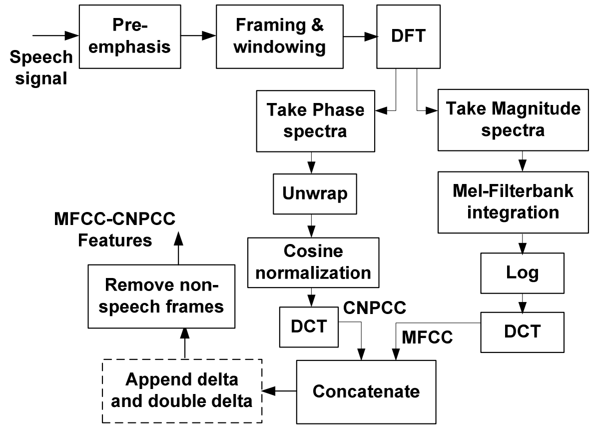

Figure 1: Conventional MFCC, phase-based feature CNPCC and joint Amplitude & phase-based countermeasures MFCC-CNPCC extraction steps.

2.4. Modified Group Delay-based Countermeasures

The negative derivative of the phase spectrum is known as group delay function and it is expressed as:

$$\tau_g(\omega) = -\frac{d}{d\omega}\big(\theta(\omega)\big) = \frac{X_R(\omega)Y_R(\omega) + X_I(\omega)Y_I(\omega)}{|X(\omega)|^2}, \qquad (1)$$

where $X(\omega)$ is the Fourier transform of $x(n)$, $Y(\omega)$ is the Fourier transform of $y(n) = nx(n)$, and the subscripts R and I denote the real and imaginary parts, respectively.

If the zeros of the system transfer function are not close to the unit circle the group delay function behaves well [26-29]. The modification of the group delay function, introduced in [26], is performed by suppressing the zeros of the transfer function. This is done by replacing the magnitude spectrum $X(\omega)$ by its cepstrally smoothed version $S(\omega)$ and by introducing two parameters α ($0 < \alpha \le 1$) and γ ($0 < \gamma \le 1$) to control the dynamic range. The modified function is given by

$$\tau_m(\omega) = \frac{\tau(\omega)}{|\tau(\omega)|}\left(|\tau(\omega)|^{\alpha}\right), \qquad (2)$$

where

$$\tau(\omega) = \frac{X_R(\omega)Y_R(\omega) + X_I(\omega)Y_I(\omega)}{|S(\omega)|^{2\gamma}}. \qquad (3)$$

$P(\omega) = X_R(\omega)Y_R(\omega) + X_I(\omega)Y_I(\omega)$ is known as the product spectrum. Cepstrally smoothed spectra $S(\omega)$ are obtained using the following the steps [13]:

- ✓ Take the log of $X(\omega)$ to obtain the log amplitude spectra and smooth it by applying median filter with a window of 5. Apply a DCT to the log spectra and take the first 30 cepstral coefficients.
- ✓ Apply the inverse DCT to the cepstral coefficients to obtain cepstrally smoothed spectra $S(\omega)$.

As shown in fig. 2, after computing $P(\omega)$ and $S(\omega)$ the modified group delay function (MGDF) is obtained using eqns. (2)-(3) by selecting optimal values for the parameters α and γ. Here, we found the optimal values for the tuning parameters are $\alpha = \gamma = 0.1$. The modified group delay cepstral coefficients (MGDCC) or modified group delay filterbank cepstral coefficients (MGDFCC, when the filterbank is integrated) by applying DCT to the MGDF and taking the first q=12 coefficients (excluding c0). Delta and double delta features are added. Finally, non-speech frames are removed using the VAD label files.

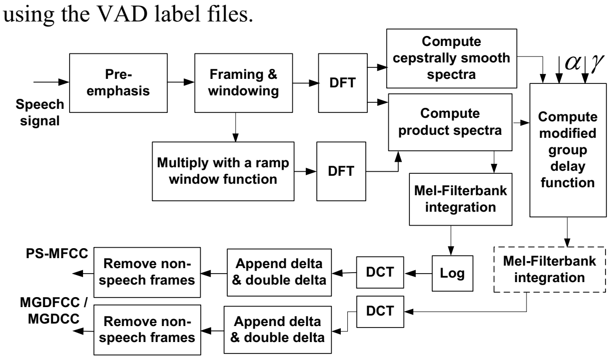

Figure 2: Schematic diagram showing various steps to extract spoofing countermeasures based on modified group delay and product spectrum.

2.5. Product Spectrum based Countermeasures

The product spectrum $P(\omega)$ was first introduced in [27] for a speech recognition task. It helps to mitigate the effect of zeros in the group delay function. It is defined as the product of power spectrum $|X(\omega)|^2$ and the group delay function $\tau_g(\omega)$ and expressed as:

$$P(\omega) = |X(\omega)|^2 \tau_g(\omega) = X_R(\omega)Y_R(\omega) + X_I(\omega)Y_I(\omega). \qquad (4)$$

Eqn. (4) indicates that the product spectrum incorporates information from both the amplitude and phase spectra and therefore, this feature may be a good candidate for spoofing detection and speaker verification. Figure 2 provides an overview of the MFCC feature extraction procedure from the product spectrum.

2.6. All-pole Group Delay-based Countermeasures

Because of the excitation source and also due to an artifact of short-time processing [24, 25] some zeros can occur in the vicinity of the unit circle. Calculation of the group delay function using eqn. (1) at frequency bins near these zeros thus results in high amplitude spurious peaks. These peaks mask out the formant structure [27, 30]. Modified group delay [26], product spectrum [27], and chirp group delay [30], all-pole group delay [24-25] functions have been proposed to alleviate the problem associated with group delay. In all-pole modeling the idea is to keep only the vocal tract (filter) component of the speech signal and discard the contribution due to the excitation source. This can be approximated by extracting the spectral envelope of the speech signal via all-pole modeling. Fig. 4 presents different steps for the extraction of cepstral features from the all-pole group delay function.

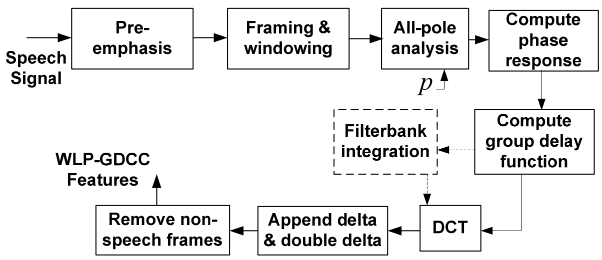

Figure 3: All-pole group delay-based countermeasures extraction stages.

Once pre-processing (i.e., pre-emphasizing, framing and windowing) is done perform all-pole modeling (e.g., linear prediction (LP) with a model order p to obtain the autoregressive (AR) coefficients $\mathbf{a}=\{a(k,m)\}$, $k=1,2,...,p$; $m=1,2,...,M$; where k is the index for AR coefficients and m is the frame index. Compute the phase response from the AR coefficients $\mathbf{a}$. Here, we use weighted LP (WLP) [33] for all-pole analysis with prediction order $p=80$. The group delay is calculated by taking the negative derivative of the phase response. Cepstral coefficients are obtained by applying a DCT on the group delay function. We keep the first q coefficients (here, q = 12) excluding the 0-*th* cepstrum and we append delta and double delta features to form $3q$ dimensional features.

Note that, no compression (logarithmic or power-law nonlinearity) is needed to compute cepstral features from the phase spectra or group delay function. This is because multiplication of Fourier transform of two signals (e.g., source and filter) corresponds to addition of their phase spectra [24].

2.7. LP Residual-based Countermeasures

In LP analysis each sample is predicted as a linear weighted sum of the past p samples as:

$$\hat{x}(n)=\sum_{k=1}^{p}a_k x(n-k)\,, \tag{5}$$

where p is prediction order, $x(n)$ is current sample, and $\{a_k\}$ are LP coefficients. The residual $r(n)$ is the prediction error obtained as the difference between the predicted speech sample $\hat{x}(n)$ and the actual speech sample as:

$$r(n)=x(n)-\hat{x}(n)=x(n)-\sum_{k=1}^{p}a_k x(n-k)\,. \tag{6}$$

If the proper prediction order (in the range pf 8-20 for a 8kHz sampled signal) is used the LP residual mostly contains the excitation source information [31, 32]. In this work we use p = 24 as the sampling frequency of challenge data is 16kHz. It is evident that $r(n)$ might contain information which has not been captured by the LP coefficients of the actual signal and which can be used for speaker recognition [32] and spoofing detection tasks. In this work we propose LP residual cepstral coefficients (LPRC) as a countermeasure for spoofing attacks. Fig. 4 depicts the various steps involved in the extraction of LPCC (linear prediction cepstral coefficients) and LPRC countermeasures by performing LP analysis of actual and the residual signals, respectively, and then converting the LP coefficients directly to cepstral coefficients. Energy computed (without applying any pre-processing) from the raw signal is appended before computing derivative features. It is observed from fig. 5 that LPRC is more discriminative than LPCC for distinguishing human speech from spoofed speech.

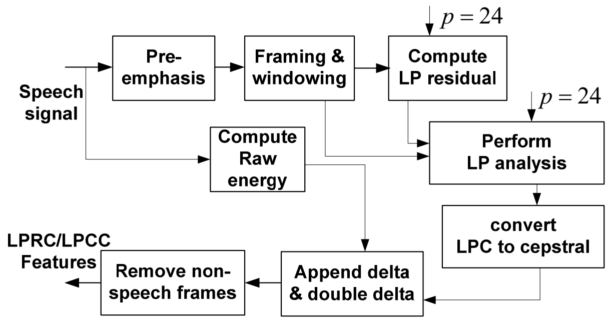

Figure 4: Block diagram showing various steps to extract linear prediction cepstral coefficients (LPCC) and linear prediction residual cepstral coefficients (LPRCC) by performing LP analysis of actual speech signal and residual signal, respectively.

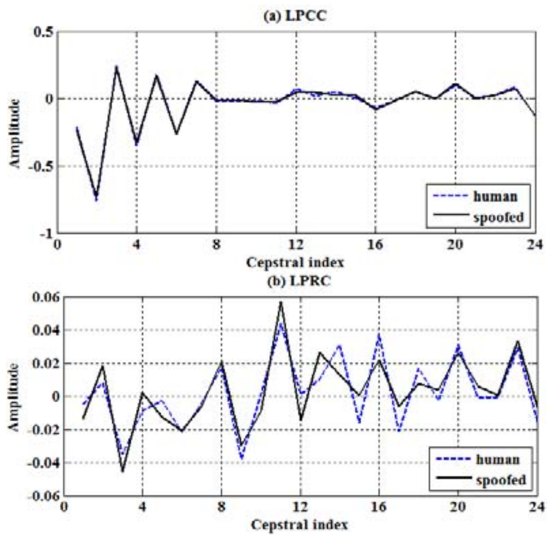

Figure 5: Comparison of human and spoofed speech (using voice conversion) for a frame when LPCC and LPRC countermeasures are used.

3. Experiments

3.1. Features and Training of Models

In this work we use features described in section 2 as spoofing countermeasures. The Feature dimension is 39 (including log energy, delta and double deltas) for all systems except the MFCC-CNPCC. The MFCC-CNPCC feature dimension is 25 (12-dimensional MFCC + 12-dimensional CNPCC + log energy, no derivatives are used). Without the energy feature dimension is 36. No Feature normalization is applied. Non-speech frames were removed using VAD (voice activity detector) segmentations generated by a GMM-based VAD [34, 35].

A 512-component Gaussian Mixture Model (GMM) is used for training the human speech and synthetic speech models, λ_h and λ_s. Then, given the feature vector sequence O of a test speech signal, the human or synthetic speech is decided based on the following log-likelihood ratio:

$$\ell(O) = \log \frac{\left(p(O|\lambda_s)\right)}{\left(p(O|\lambda_h)\right)}. \tag{7}$$

For each countermeasure we train a human speech model using the provided human (i.e., natural) speech signals and six synthetic speech models using the provided synthetic speech signals generated by the five different spoofing techniques (denoted as S1, S2...,S5) [7]. The sixth model is trained by pooling the all the synthetic speech signals of the five spoofing techniques mentioned in the evaluation plan.

3.2. Results and discussion

The countermeasures discussed in section 2 are evaluated on the ASVspoof 2015 challenge evaluation and development test data. Equal Error rate (EER) is used as evaluation metric.

Results, in terms of percentage EER, obtained with different countermeasures and with the primary submitted system (FUSED) on the challenge evaluation data are reported in table 1. Among the individual systems PS-MFCC and MFCC outperformed other systems in ***unknown*** and ***All*** conditions. In ***known*** condition MFCC-CNPCC is the 2nd best system after FUSED system. Overall, PS-MFCC provided reduced EER compare to other individual systems. The advantage of PS-MFCC is that it is database independent. Unlike MGFCC/MGDCC and WLP-GDCC it dies not have any tuning parameters.

Table 1. Spoofing performance on the challenge evaluation data using a standard GMM classifier with various features as countermeasures and with fused system (CRIM's primary system). Lower EERs are highlighted with orange color.

	EER (%)		
	Known	Unknown	All
MFCC	0.785	5.398	3.091
MFCC-CNPCC	0.450	6.599	3.525
PS-MFCC	0.652	5.372	3.011
MGDFCC	1.003	5.704	3.3539
MGDFCC (w/o E)	2.053	7.061	4.557
MGDCC	1.924	7.124	4.524
WLP-GDCC	1.436	8.941	5.188
FUSED	0.041	5.347	2.694

Spoofing detection performance of the FUSED system against different spoofing techniques present in the evaluation set is given in table 2. It is observed from the results that the average EER obtained by the FUSED system over all spoofing techniques but S10 is 0.0604%. The reason behind this very good result on ***known*** (S1-S5) and ***unknown*** (S6-S9) attacks is the use of similar vocoding technique (STRAIGHT) in the training and evaluation data. It is mentioned in [36] that no vocoder was used in spoofing technique S10 synthesis. Vocoder mismatch between the training and evaluation data resulted an EER of 26.39% on the S10 attack. These results depict that though a countermeasure can provide a good result on known and vocoder matched attacks, it may not perform well on mismatched vocoded and unknown spoofing attacks.

Since the scores with LPCC and LPRC systems were not ready before the submission of primary system's results the FUSED system did not include these two features. As the keys for the evaluation data have not been released yet, in Table 3, we reported results of LPRC and LPCC systems on all development (**All-dev**) test data. In this case, spoofed models were trained on only S1 spoofed data. **All-dev** contains one known attack S1 and four unknown attacks S2-S5. The LPRC countermeasure system performed very well as the EER obtained by it is 0.735% on **All-dev** test set and this is the lowest EER compared to the results obtained with other systems. This indicates that LPRC is able to give good performance both on known and unknown attacks.

Table 2. Spoofing detection performance against various known (S1-S5) and unknown (S6-S10) spoofing attacks on the challenge evaluation data with CRIM's primary system. We use a standard GMM classifier for spoofing detection task. The lowest EERs are highlighted with orange color.

EER (%)					
Known					
S1	S2	S3	S4	S5	Average
0.0242	0.1046	0.0252	0.0167	0.0325	0.041
Unknown					
S6	S7	S8	S9	S10	Average
0.0932	0.0108	0.2362	0.0000	26.3926	5.347
Average					2.694

Table 3. Comparison of performance of LPRC and LPCC with other countermeasures when the spoofed model is trained on the S1 [7] spoofing technique data and tested on all five development test spoofing techniques (S1-S5) [7] data. The lowest EER is highlighted with orange color.

EER (%)			
	All-dev		**All-dev**
MFCC	6.24	WLPGDCC	4.7
MFCC-CNPCC	4.01	LPCC	6.40
PS-MFCC	5.51	LPRC	0.735
MGDFCC	22.23		

4. Conclusions

In this paper we used some existing (e.g., MGDCC, CNPCC) countermeasure and introduced two new ones, MFCC-CNPCC and LPRC, for the first ASVspoof 2015 challenge tasks. Our primary system performed very well on known attacks with an EER of 0.041% and but resulted in considerably higher EER on unknown attacks. The LPRC countermeasure showed excellent performance on both known and unknown spoofing attacks.

5. Acknowledgements

We would like to thank Zhizheng Wu and the organizer of ASVspoof 2015 challenge for evaluating some additional scores for us.

6. References

[1] Y. Lau, D. Tran, and M. Wagner, "Testing voice mimicry with the yoho speaker verification corpus," in Knowledge-Based Intelligent Information and Engineering Systems. Springer, 2005, pp. 907-907.

[2] J. Villalba and E. Lleida, “Preventing replay attacks on speaker verification systems,” in IEEE Int. Carnahan Conf. on Security Technology (ICCST) , 2011.

[3] D. Matrouf, J.-F. Bonastre, and C. Fredouille, "Effect of speech transformation on impostor acceptance," in proceed. of ICASSP, vol. 1. pp. I-I, 2006.

[4] T. Kinnunen, Z. Wu, K. A. Lee, F. Sedlak, E. S. Chng, and H. Li, "Vulnerability of speaker verification systems against voice conversion spoofing attacks: The case of telephone speech," in International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 4401-4404, 2012.

[5] P. L. De Leon, M. Pucher, and J. Yamagishi, "Evaluation of the vulnerability of speaker verification to synthetic speech," in Proc. IEEE Speaker and Language Recognition Workshop (Odyssey), pp. 151-158, 2010.

[6] N. Evans, T. Kinnunen, Junichi Yamagishi, "Spoofing and countermeasures for automatic speaker verification," in Proc. of INTERSPEECH, Lyon, France, 2013.

[7] Z. Wu, T. Kinnunen, N. Evans, J. Yamagishi, ASVspoof 2015: Automatic Speaker Verification Spoofing and Countermeasures Challenge Evaluation plan, 2015. http://www.spoofingchallenge.org/asvSpoof.pdf

[8] J. Villalba and E. Lleida,, “Detecting replay attacks from far-field recordings on speaker verification systems,” in Biometrics and ID Management , ser. Lecture Notes in Computer Science, C. Vielhauer, . Dittmann, A. Drygajlo, N. Juul, and M. Fairhurst, Eds. Springer, pp. 274–285, 2011.

[9] F. Alegre, A. Janicki, and N. Evans, “Re-assessing the threat of replay spoofing attacks against automatic speaker verification,” in Proc. Int. Conf. of the Biometrics Special Interest Group (BIOSIG), 2014.

[10] Z. Wu, S. Gao, E. S. Chng, and H. Li, “A study on replay attack and anti-spoofing for text-dependent speaker verification,” in Proc. Asia-Pacific Signal Information Processing Association Annual Summit and Conference (APSIPA ASC), 2014.

[11] P. L. De Leon, I. Hernaez, I. Saratxaga, M. Pucher, and J. Yamagishi, “Detection of synthetic speech for the problem of imposture,” in Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing (ICASSP) , 2011.

[12] P. L. De Leon, M. Pucher, J. Yamagishi, I. Hernaez, and I. Saratxaga, “Evaluation of speaker verification security and detection of HMM-based synthetic speech,” IEEE Trans. Audio, Speech and Language Processing , vol. 20, no. 8, pp. 2280–2290, 2012.

[13] Z. Wu, X. Xiao, E. S. Chng, and H. Li, “Synthetic speech detection using temporal modulation feature,” in Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing (ICASSP) , 2013.

[14] J. Sanchez, I. Saratxaga, I. Hernaez, E. Navas, and D. Erro, “A cross-vocoder study of speaker independent synthetic speech detection using phase information,” in Proc. Interspeech , 2014.

[15] Z. Wu, E. S. Chng, and H. Li, “Detecting converted speech and natural speech for anti-spoofing attack in speaker recognition,” in Proc. Interspeech 2012 , 2012.

[16] Z. Wu, T. Kinnunen, E. S. Chng, H. Li, and E. Ambikairajah, “A study on spoofing attack in state-of-the-art speaker verification: the telephone speech case,” in Proc. Asia-Pacific Signal Information Processing Association Annual Summit and Conference (APSIPA ASC), 2012.

[17] F. Alegre, R. Vipperla, A. Amehraye, and N. Evans, “A new speaker verification spoofing countermeasure based on local binary patterns,” in Proc. Interspeech, 2013.

[18] F. Alegre, A. Amehraye, and N. Evans, “Spoofing countermeasures to protect automatic speaker verification from voice conversion,” in Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing (ICASSP) , 2013.

[19] Niko Brummer, Edward de Villiers, "The Bosaris Toolkit," 2013: https://sites.google.com/site/bosaristoolkit/

[20] Z. Wu, E.S. Chng, and H. Li, “Detecting converted speech and natural speech for anti-spoofing attack in speaker recognition,” in Interspeech , 2012.

[21] Zhizheng Wu, Tuomas Virtanen, Eng Siong Chng, Haizhou Li, "Exemplar-based sparse representation with residual compensation for voice conversion", IEEE/ACM Transactions on Audio, Speech and Language Processing, Vol. 22, Issue 10, pp. 1506-1521, 2014.

[22] Z. Wu, N. Evans, T. Kinnunen, J. Yamagishi, F. Alegre, and H. Li, “Spoofing and countermeasures for speaker verification: A survey,” Speech Communication, vol. 66, no. 0, pp. 130 – 153, 2015.

[23] S. Nakagawa, L. Wang, S Ohtsuka, "Speaker Identification and Verification by combining MFCC and Phase Information," IEEE TASLP, vol. 20, no. 4, pp. 1085-1095, 2011.

[24] E.Loweimi, S.M.Ahadi, T.Drugman, A New Phase-based Feature Representation for Robust Speech Recognition, IEEE International Conference on Audio Speech and Signal Processing (ICASSP), Vancouver, Canada, 2013.

[25] P. Rajan, T. Kinnunen, C. Hanilci, J. Pohjalainen, P. Alku, ”Using group delay functions from all-pole models for speaker recognition”, *Proc. Interspeech 2013*, pp. 2489--2493, Lyon, France, August 2013.

[26] H. Murthy and V. Gadde. The modified group delay function and its application to phoneme recognition. In Proc. of ICASSP, vol. 1, p. 68–71, 2003.

[27] D. Zhu and K. Paliwal, “Product of power spectrum and group delay function for speech recognition,” in Proc. Int. Conf. Acoust., Speech, Signal Process. , pp. 125–128, 2004.

[28] H. Banno, J. Lu, S. Nakamura, K. Shikano, and H. Kawahara, "Efficient representation of short-time phase based on group delay," In Proc. Int. Conf. Acoust. Speech Signal Process., vol. 2, p. 861–864, 1998.

[29] R. Hegde, H. Murthy, and V. Gadde, "Significance of the modified group delay feature in speech recognition," IEEE Transactions on Audio, Speech, and Language Processing, , vol. 15, no. 1, p.190–202, 2007.

[30] B. Bozkurt, L. Couvreur, and T. Dutoit, "Chirp group delay analysis of speech signals," Speech Commun., vol. 49, p. 159–176, 2007.

[31] S. R. M. Prasanna, C. S. Gupta and B. Yegnanarayana, ”Extracting speaker-specific information from linear prediction residual”, Speech Communication, vol. 48, no. 10, pp. 1243 - 1261, 2006.

[32] K. P. Markov and S. Nakagawa, “Text-independent speaker recognition using multiple information sources,” *International Conference on Spoken Language Processing ICSLP-1998*, pp. 0744, 1998.

[33] Jouni Pohjalainen and Paavo Alku, "Robust speech analysis by lag-weighted linear prediction", in Proc. 37th International Conference on Acoustics, Speech, and Signal Processing (ICASSP), Kyoto, Japan, March 25-30, 2012.

[34] T. Kinnunen, P. Rajan, ”A practical, self-adaptive voice activity detector for speaker verification with noisy telephone and microphone data”, *Proc. of* ICASSP, pp. 7229-7233, Vancouver, Canada, May 2013.

[35] Alam, J., Kenny, P., Ouellet, P., Stafylakis, T. and Dumouchel, P., "Supervised/Unsupervised Voice Activity Detectors for Text-Dependent Speaker Recognition on the RSR2015 Corpus," Proc. Odyssey Speaker and Langauge Recognition Workshop, Joensuu, Finland June 2014.

[36] Z. Wu, T. Kinnunen, N. Evans, J. Yamagishi, C. Hanilçi, M. Sahidullah, A. Sizov, "ASVspoof 2015: the First ASV Spoofing and Countermeasures Challenge," INTERSPEECH 2015. http://www.spoofingchallenge.org/is2015_asvspoof.pdf