SFF Anti-Spoofer: IIIT-H Submission for Automatic Speaker Verification Spoofing and Countermeasures Challenge 2017

by

Raju Alluri K.N.R.K, Sivanand a, sudarsanareddy.kadiri@research.iiit.ac.in kadiri, Suryakanth V Gangashetty, Anil Kumar Vuppala

in

Interspeech 2017
(*Interspeech 2017*)

Stockholm, Sweden

Report No: IIIT/TR/2017/-1

Centre for Language Technologies Research Centre
International Institute of Information Technology
Hyderabad - 500 032, INDIA
August 2017

SFF Anti-Spoofer: IIIT-H Submission for Automatic Speaker Verification Spoofing and Countermeasures Challenge 2017

K N R K Raju Alluri, Sivanand Achanta, Sudarsana Reddy Kadiri, Suryakanth V Gangashetty, and Anil Kumar Vuppala

Speech Processing Laboratory, KCIS
International Institute of Information Technology, Hyderabad, India
{raju.alluri, sivanand.a, sudarsanareddy.kadiri}@research.iiit.ac.in,
{svg, anil.vuppala}@iiit.ac.in

Abstract

The ASVspoof 2017 challenge is about the detection of replayed speech from human speech. The proposed system makes use of the fact that when the speech signals are replayed, they pass through multiple channels as opposed to original recordings. This channel information is typically embedded in low signal to noise ratio regions. A speech signal processing method with high spectro - temporal resolution is required to extract robust features from such regions. The single frequency filtering (SFF) is one such technique, which we propose to use for replay attack detection. While SFF based feature representation was used at front-end, Gaussian mixture model and bi-directional long short-term memory models are investigated at the back-end as classifiers. The experimental results on ASVspoof 2017 dataset reveal that, SFF based representation is very effective in detecting replay attacks. The score level fusion of back end classifiers further improved the performance of the system which indicates that both classifiers capture complimentary information.

Index Terms: Spoofing, countermeasures, replay attack, Gaussian mixture model, bi-directional long short-term memory, single frequency filtering.

1. Introduction

Recent advances in speech technology made automatic speaker verification (ASV) as a reliable biometric solution to many applications like e-commerce and telephone banking [1, 2]. A general assumption in ASV is that the authorized user produces speech signal to the verification system for access. However, this may not be true for all cases. For example, an unauthorized user may get the access of verification system by imitating the authorized speaker voice. This manipulation is known as spoofing attack. The current state-of-the-art ASV systems [3, 4] are robust to the session and channel variations. However, they are vulnerable to spoofing attacks [5]. In the literature, four spoofing attacks were registered [5]. They are, impersonation, voice conversion (VC), speech synthesis (SS), and replay. In the present study, the focus is on developing countermeasures for replay attacks.

A survey of studies on spoofing attacks for ASV is presented in-detail in [5]. According to the survey in [5], most of the countermeasures were developed with the prior knowledge of spoofing attacks. However, this may not be the scenario for practical cases, where the nature of the attacks can not be known prior. Also, most of the works were conducted on non standard databases and hence the results are not comparable.

With the aim to setup a standard datasets and common evaluation protocols, a special session [6] in spoofing countermeasures for ASV was conducted in INTERSPEECH 2013. Because spoofing can be affected by high quality SS and VC, collaborating with respective communities led to a rich and standard data set on which robust countermeasures could be evaluated. As a part of the series, the second special session was conducted in INTERSPEECH 2015 [7]. The organisers came up with a standard text independent data set and a common protocol to deal with VC and SS attacks. Several researchers have developed countermeasures for these attacks, the results are summarised in [8]. As replay attacks are not included in ASVspoof 2015 data set, the researchers from Idiap came up with a more generalised database AVspoof[1] [9] which contains VC, SS and replay attacks. In biometrics theory applications and systems (BTAS) 2016 [10], a special session was conducted on speaker anti spoofing challenge by providing a replay database collected from AVspoof. There are several studies reported on BTAS 2016 corpus [10, 11, 12, 13]. AVspoof corpus is collected with few recording devices in a controlled environment with varying acoustics. For practical scenario there is a need of more generalised replay corpus. In the current special session on ASV spoofing and countermeasures (ASVspoof 2017) challenge[2], the organisers provided a new text-dependent replay corpus [14] which is more diverse in nature than AVspoof for replay attack detection.

The majority of the successful countermeasures to replay attack detection reported in the literature are based on non-conventional features like inverted mel frequency cepstral coefficients (IMFCC) [10], rectangular frequency cepstral coefficients (RFCC) [11] and constant Q cepstral coefficients (CQCC) [12, 13] coupled with Gaussian mixture model (GMM). In this study, we use recently proposed single frequency filtering coefficients (SFFCC) [15] for the task of replay attack detection. Gaussian mixture model (GMM) and bi-directional long short-term memory (BLSTM) model are used as classifiers. As these two classifiers are distinctive in nature i.e, GMM is a generative model whereas BLSTM is a discriminative model, we fuse the two classifier scores with a logistic regression to get benefit from complementary nature of these classifiers.

The organisation of the paper is as follows. In Section 2, proposed approach is described in detail which includes front-end features and back-end classifiers used. The experimental setup with detailed feature representation and classifier parameters are presented in Section 3. Results are discussed in Section 4. Finally, conclusion of study is presented in Section 5.

[1] https://www.idiap.ch/dataset/avspoof
[2] http://www.spoofingchallenge.org

2. Proposed Approach

In this section, the components used in the proposed system specifically front-end features, back-end classifiers and fusion are detailed.

2.1. Front-end Features: Single Frequency Filter Cepstral Coefficients (SFFCC)

In this study, the features are extracted from recently proposed single frequency filtering (SFF) method which provides high spectro-temporal resolution [16]. The block diagram of SFFCC extraction is shown in Figure 1.

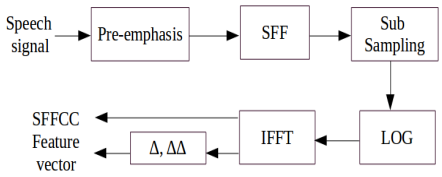

Figure 1: *Block diagram of SFFCC extraction.*

The blocks in Figure 1 can be grouped into following three steps. They are, SFF spectrum estimation [17], sub-sampling and cepstrum computation, respectively. The steps to extract SFFCC [15] are given below.

2.1.1. SFF Spectrum

The speech signal ($s[n]$) is differenced in order to remove the bias introduced by recording device during recording. The differenced speech signal (x[n]) is then multiplied with a complex sinusoidal $e^{j\bar{w}_k n}$. The resultant frequency shifted signal x[k,n] is then passed through a single-pole filter with a transfer function of $H(z) = \frac{1}{1+rz^{-1}}$. In this study, $r = 0.995$ is used. The output of the filter y[k,n] is given by

$$y[k,n] = -ry[k,n-1] + x[k,n] \tag{1}$$

The magnitude of the signal y[k,n] is given by

$$v[k,n] = \sqrt{Re(y[k,n])^2 + Im(y[k,n])^2} \tag{2}$$

Here Re, Im represents the real and imaginary parts respectively. The term v[k,n] corresponds to the SFF envelope of the signal at a desired frequency f_k. The magnitude spectrum can be obtained from SFF envelope for each time instant of n.

2.1.2. Sub-sampling

The instantaneous energy (E[n]) is computed from the magnitude spectrum v[k,n] by summing all the values across frequency. The equation for E[n] is given by

$$E[n] = \sum_{k=1}^{K} v[k,n] \tag{3}$$

For each 10 ms speech segment a lowest energy (low SNR) instant is selected from the instantaneous energy. The low SNR instant for the j^{th} segment is given by

$$l_j = \arg\min_i E_j[i] \tag{4}$$

where $E_j[i]$ represents the instantaneous energy in j^{th} segment. The resultant signal after sub-sampling is v[k,l], where l < n.

2.1.3. Cepstrum Computation

The cepstrum is computed from the sub-sampled SFF spectrum v[k,l] in the following manner.

$$c[k,l] = IFFT(log(v[k,l])) \tag{5}$$

From c[k,l], first few cepstral coefficients (p) are considered. Here p represents the number of static cepstral coefficients. Dynamic coefficients are computed from static coefficients.

2.2. Back-end Classifiers

2.2.1. Gaussian Mixture Model (GMM)

GMM [18] is a weighted sum of M component densities given by the equation

$$p(x/\lambda) = \sum_{t=1}^{M} w_i p_i(x) \tag{6}$$

where x is D-dimensional feature vector, w_i, i=1,2,...,M represents mixture weights and $p_i(x)$, i=1,2....,M, are the component densities with mean vector $\overrightarrow{\mu}$ and covarinace matrix $\sum_i$. Here mixture weights will satisfy the constraint that $\sum_{i=1}^{M} w_i = 1$. GMM parameters are represented by $\lambda = \{w_i, \overrightarrow{\mu}, \sum_i\}_{i=1}^{M}$.

The model parameters are estimated using expectation maximization (EM) algorithm [19] for each class individually with maximum likelihood (ML) criteria. In EM algorithm a new model $\bar{\lambda}$ parameters are estimated from the previous parameters λ such that $P(x/\bar{\lambda}) \geq P(x/\lambda)$, this process is continued till the convergence threshold is reached.

In this study, from the training data, two GMMs for each genuine ($\lambda_{genuine}$) and spoof (λ_{replay}) are build. For the test utterance (X), the score is computed with the following equation,

$$Score(X) = llk(X|\lambda_{genuine}) - llk(X|\lambda_{replay}) \tag{7}$$

where X = $\{x_1, x_2, ..., x_T\}$ is the feature vector of test utterance, T represents number of frames. Here $llk(X|\lambda)$ represents the average likelihood of X given model λ.

$$llk(X|\lambda) = (1/T)\sum_{t=1}^{T} logp(x_t/\lambda) \tag{8}$$

2.2.2. Bi-directional Long Short-Term Memory (BLSTM)

BLSTM is a recurrent neural network architecture shown in Figure 2 [20]. It can be seen from the Figure 2 that, The forward hidden state $\overrightarrow{\mathbf{h_t}}$ [21] is computed at every time step using the current input x_t and past hidden state $\overrightarrow{\mathbf{h}}_{t-1}$. Similarly the backward hidden states $\overleftarrow{\mathbf{h_t}}$ are computed.

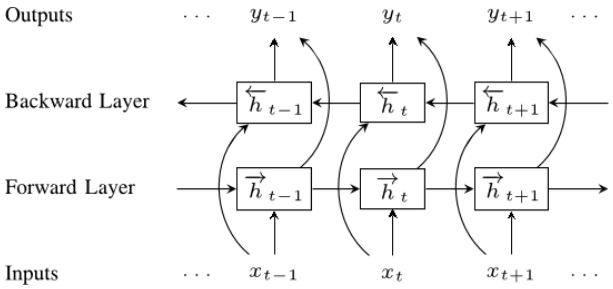

Figure 2: *Bi-directional Long-Short Term Memory architecture (figure adapted from [20]).*

Hence the state at the end of the sequence $\overrightarrow{\mathbf{h_T}}$ is a summarization of the entire signal in the forward direction. Analogously $\overleftarrow{\mathbf{h_T}}$ in the backward direction. From this utterance representation, we estimate posteriors of the classes using a softmax function. The equation of output is given as

$$\mathbf{y_T} = Softmax(\mathbf{U_f}\overrightarrow{\mathbf{h_T}} + \mathbf{U_b}\overleftarrow{\mathbf{h_T}} + \mathbf{b_u}). \quad (9)$$

Where $\mathbf{U_f}, \mathbf{U_b}$ and $\mathbf{b_u}$ represent output layer forward, backward weights and the bias.

As the present study is a two class problem, $\mathbf{y_T}$ is a two-dimensional vector with $\mathbf{y_{genuine}}$ and $\mathbf{y_{replay}}$ as elements. $\mathbf{y_T} = [y_{genuine}, y_{replay}]$

The score for the test utterance is computed as

$$Score(X) = y_{genuine} - y_{replay}. \quad (10)$$

2.3. Score Fusion

The two classifiers used in this study are of distinctive in nature i.e., GMM is a generative model whereas BLSTM is a discriminative model. In order to get benefit from both the models, we fused our systems at score level using a logistic regression classifier whose weights are trained on training data.

3. Experimental Setup

3.1. Database

In this study, the experiments are carried out on ASVspoof 2017 dataset which is provided as a part of spoofing challenge [14]. The genuine recordings in this dataset are taken from RedDots corpus [22] and their replayed recordings are collected [23]. The dataset is divided into three non-overlapping subsets. The details of these recordings are given in Table 1.

Table 1: *Details of number of speakers and number of utterances in ASVspoof 2017 data-set.*

		# utterances		
Subset	# Speakers	Genuine	Spoofed	Total
Triaining	10	1508	1508	3016
Development	8	760	950	1710
Test	24	1298	12922	14220

In this data set, more diverse spoof recordings are incorporated in development set and evaluation set than that of training set. Variety of new configurations in development and evaluation sets are considered to provide more generalized countermeasures for practical scenarios where we do not know the type of replayed configuration beforehand. Further details regarding replay configurations and sessions can be found in [24].

3.2. Parameters used for Feature Extraction

In SFF, amplitude envelopes can be computed at each frequency (f_k). In this study, we computed amplitude envelope at every 15.6 Hz frequencies within the range of 0 to Nyquist frequency ($f_s/2$) which results 513 envelopes. In principle, the SFFCC can be obtained at each time instant. Here, instead of computing at each instant, we computed the SFFCC from the sub-sampled SFF spectrum.

Figure 2(a) shows a segment of speech signal, Figure 2 (b) shows SFF spectrogram, and Figure 2 (c) shows the instantaneous energy computed from SFF spectrum. Sub-sampling is done as explained in Section 2, by selecting the low SNR instant in each 10 ms segment. The low SNR instant in each segment are marked with (□) in Figure 2(c) whereas vertical lines represent 10 ms instants.

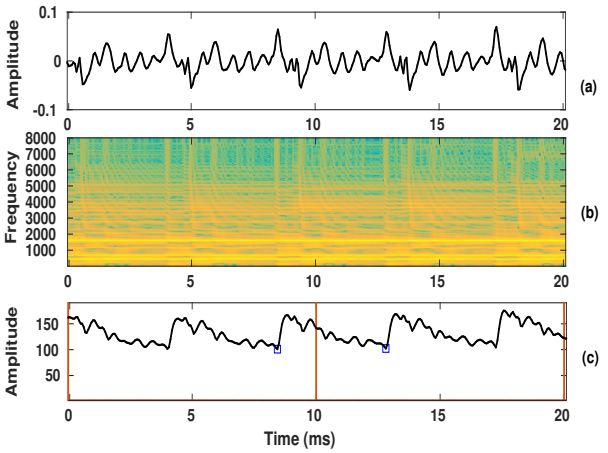

Figure 3: *(a) A segment of speech signal (b) SFF spectrogram of (a). (c) Instantaneous energy of (a) with marked 10 ms instant(|) and low SNR instant (□) within each 10 ms segment.*

In this study, various dimensions (p= 13, 20 and 30) of cepstral coefficients were considered. For experimentation, different combination of static (S), delta (D) and double-delta (A) are considered.

3.3. Classifiers

In this study, GMM and BLSTM are used as classifiers. The parameters considered for training are as follows.

3.3.1. GMM

512 mixture GMMs are built with 10 iterations of EM algorithm for each class of genuine and replay.

3.3.2. BLSTM

In this study, we use a neural network with one bi-directional LSTM layer [25]. The number of nodes in the input layer is equal to the dimensionality of the feature vector. The output layer contains 2 nodes as there are two classes (genuine and replay). The forward and backward hidden layers contain 5 units each with $tanh$ non-linearity. The weights of the network are randomly initialised from a Gaussian distribution with variance scaled to 0.01, and biases are initialized to zero and the forget-gate bias which is initialised to 1 [26]. While training the model, all the data from both genuine and spoof are pooled and randomized so that the model is not biased to any particular class during the initial phase of training. The model was trained using pure stochastic gradient descent with ADAM optimiser [27]. The learning rate was set to $\alpha = 0.003$ and decay rate was set to $\beta_1 = 0.9$. While testing the posterior probabilities at output nodes are differenced to get a similar type of score mentioned in Section 2.2.

4. Results and Discussion

As per the ASVspoof 2017 challenge protocol, the results are reported in terms of equal error rate (EER). The EER values are computed using BOSARIS toolkit [28]. The experimental results on development and evaluation datasets are discussed in detail in the following section.

4.1. Results on Development Data

In order to select proper dimension and feature combination for replay attack detection, we conducted several experiments by considering three different dimensions (13, 20 and 30) and several combinations of static and dynamic coefficients in each

dimension on the development set with GMM classifier. The results are reported in Table 2.

Table 2: *Performance (in % of EER) for SFFCC with different configurations on ASV Spoof 2017 development dataset using GMM.*

	Dimensions		
	13	20	30
S	6.21	6.48	**6.16**
D	8.19	5.06	**2.35**
A	10.22	12.27	**8.68**
SD	**5.03**	6.13	**5.03**
SA	7.25	7.56	**6.50**
DA	5.20	**4.40**	5.78
SDA	6.13	6.30	**4.44**

From the results in Table 2, it can be observed that the best EER for each of 13, 20 and 30 dimensions is obtained with SD, DA and D combinations, respectively. However there is no consistency in the feature combination across dimensions. For majority of the feature combinations 30-dimensional coefficients were performed better than the 13 and 20 dimensional coefficients. This suggest us that higher order coefficients are relatively better performing than the lower order coefficients for replay attack detection. The best performing feature combination across dimensions is 30-dimensional delta coefficients. So, we consider this as our primary countermeasure for ASVspoof 2017 challenge. The second best performing in 30-dimensional combination (SDA) is considered for further experiments using BLSTM classifier and the results are reported in Table 3.

Table 3: *Performance (in % of EER) for SFFCC with different classifiers on ASV Spoof 2017 Development dataset.*

	GMM	BLSTM
SFFCC-D	2.35	3.66
SFFCC-SDA	4.44	4.10

From the results in Table 3, it can observe that the channel variations which are useful for replay attack detection with delta coefficients are better captured by GMM, whereas for static appended with dynamic coefficients BLSTM is more suitable. This result suggests us that lower dimensional features are better modeled by GMM and higher order features by BLSTM.

4.2. Results on Evaluation Data

The same setup as mentioned in above section is used to produce results on evaluation data and the results are reported in Table 4. These results were provided by the challenge organisers. From the results, it can be observed that for SFFCC-D,

Table 4: *Performance (in % of EER) for SFFCC with different classifers on ASV Spoof 2017 Evaluation dataset.*

	GMM	BLSTM
SFFCC-D	20.2	22.4
SFFCC-SDA	30.73	20.86

the performance of GMM system is better than that of BLSTM and for SFFCC-SDA BLSTM is performing better. The same trend is also observed on development data. Surprisingly the performance of BLSTM on SFFCC-SDA is better than SFFCC-D which is not the case with development data. This ensures that BLSTM is better generalized for SFFCC-SDA features than the SFFCC-D for diverse attacks.

4.3. Results on Fusion of Classifier Scores

As the two classifiers used in this study are of distinctive in nature based on their training modalities, we fused these two classifier scores using a multi-class linear regression. The results are reported in the last column of Table 5. From the results, it is evident that the fused systems performance is better than their individual counterparts. This indicates that, the two classifiers are capturing the complementary information for replay attacks.

Table 5: *Performance (in % of EER) for our primary submission for ASVspoof 2017 challenge with fusion of classifiers on development and evaluation sets.*

		GMM	BLSTM	GMM+BLSTM
SFFCC-D	Dev	2.35	3.66	**2.21**
	Eval	20.2	22.4	**17.82**

4.4. Comparision with Baseline

The proposed system results are compared with that of ASVspoof 2017 challenge baseline which is based on CQCC-SDA. The results are reported in Table 6. For meaningful comparison, we have considered GMM classifier. From th eresults

Table 6: *Performance (in % of EER) for baseline and proposed systems using GMM classifier.*

	Dev	Eval
CQCC-SDA	10.69	30.17
SFFCC-D	2.35	20.2

in Table 6. it can be observed that proposed system has performed better than the baseline system. From the results, it is evident that SFFCC features extracted at low SNR time instants capturing the channel variations effectively than the CQCC.

5. Summary and Conclusions

This study presents IIIT-Hyderabad submission for ASVspoof 2017 challenge. In this study, single frequency filtering cepstral coefficients are used as front-end features. Both generative and discriminative models are investigated at the back-end as classifiers. From the experimental results, it is observed that higher dimensional coefficients (30) have an additional cues to detect replay attacks than lower order coefficients (13 and 20). Further analysis on 30 dimensional features revealed that lower order features (D) are better modeled by GMM whereas higher order feature (SDA) by BLSTM. The complimentary information captured by GMM and BLSTM further improved the system performance when they are fused at score level. The performance of the proposed system is compared with baseline (CQCC based system) and it has been found that the performance of the proposed system is better than the baseline in both development and evaluation data set. As the results on development set are relatively convincing than the results on evaluation set, further investigations are needed for generalization of replay attacks.

6. Acknowledgements

The first author would like to thank the Department of Electronics and Information Technology, Ministry of Communication & IT, Govt of India for granting PhD Fellowship under Visvesvaraya PhD Scheme. The second and third authors would like to thank Tata Consultancy Services (TCS), India for supporting their PhD program.

7. References

[1] J. P. Campbell, "Speaker recognition: A tutorial," *Proceedings of the IEEE*, vol. 85, no. 9, pp. 1437–1462, 1997.

[2] T. Kinnunen and H. Li, "An overview of text-independent speaker recognition: From features to supervectors," *Speech ommunication*, vol. 52, no. 1, pp. 12–40, 2010.

[3] N. Dehak, P. J. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, "Front-end factor analysis for speaker verification,," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 4, pp. 788–798, 2011.

[4] P. Kenny, P. Ouellet, N. Dehak, V. Gupta, and P. Dumouchel, "A study of interspeaker variability in speaker verification,," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 16, no. 5, pp. 980–988, 2008.

[5] 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, pp. 130–153, 2015.

[6] N. Evans, J. Yamagishi, and T. Kinnunen, "Spoofing and countermeasures for speaker verification: a need for standard corpora, protocols and metrics,," *IEEE Signal Processing Society Speech and Language Technical Committee Newsletter*, 2013.

[7] Z. Wu, T. Kinnunen, N. Evans, J. Yamagishi, C. Hanilçi, M. Sahidullah, and A. Sizov, "Asvspoof 2015: the first automatic speaker verification spoofing and countermeasures challenge,," in *Proc. INTERSPEECH*, 2015, pp. 2037–2041.

[8] Z. Wu, J. Yamagishi, T. Kinnunen, C. Hanilci, M. Sahidullah, A. Sizov, N. Evans, M. Todisco, and H. Delgado, "Asvspoof: the automatic speaker verification spoofing and countermeasures challenge," *IEEE Journal of Selected Topics in Signal Processing*, 2017.

[9] S. K. Ergünay, E. Khoury, A. Lazaridis, and S. Marcel, "On the vulnerability of speaker verification to realistic voice spoofing,," in *Proc. BTAS*, 2015, pp. 1–6.

[10] P. Korshunov, S. Marcel, H. Muckenhirn, A. Gonçalves, A. S. Mello, R. V. Violato, F. Simoes, M. Neto, M. de Assis Angeloni, J. Stuchi *et al.*, "Overview of BTAS 2016 speaker anti-spoofing competition,," in *Proc. BTAS*, 2016, pp. 1–6.

[11] P. Korshunov and S. Marcel, "Cross-database evaluation of audio-based spoofing detection systems," in *Proc. INTERSPEECH*, 2016, pp. 1705–1709.

[12] D. Paul, M. Sahidullah and G. Saha, "Generalization of spoofing countermeasures: A case study with ASVspoof 2015 and BTAS 2016 corpora," in *Proc. ICASSP*, 2017, pp. 2047–2051.

[13] M. Todisco, H. Delgado, and N. Evans, "Constant Q cepstral coefficients: A Spoofing Countermeasure for Automatic Speaker Verification," *Computer Speech and Language*, 2017.

[14] T. Kinnunen, N. Evans, J. Yamagishi, K. A. Lee, M. Sahidullah, M. Todisco, and H. Delgado, "Asvspoof 2017: Automatic speaker verification spoofing and countermeasures challenge evaluation plan."

[15] K N R K Raju Alluri, Sivanand Achanta, Sudarsana Reddy Kadiri, Suryakanth V Gangashetty and Anil Kumar Vuppala, "Detection of replay attacks using single frequency filtering cepstral coefficients," *manuscript, submitted to INTERSPEECH 2017*.

[16] G. Aneeja and B. Yegnanarayana, "Single frequency filtering approach for discriminating speech and nonspeech,," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 23, no. 4, pp. 705–717, 2015.

[17] S. R. Kadiri and B. Yegnanarayana, "Epoch extraction from emotional speech using single frequency filtering approach,," *Speech Communication*, vol. 86, pp. 52–63, 2017.

[18] D. A. Reynolds and R. C. Rose, "Robust text-independent speaker identification using gaussian mixture speaker models," *IEEE Transactions on Speech and Audio Processing*, vol. 3, no. 1, pp. 72–83, 1995.

[19] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the em algorithm," *Journal of the Royal Statistical Society. Series B (methodological)*, pp. 1–38, 1977.

[20] A. Graves, N. Jaitly, and A.-r. Mohamed, "Hybrid speech recognition with deep bidirectional LSTM," in *Proc. ASRU*, 2013, pp. 273–278.

[21] K. Greff, R. K. Srivastava, J. Koutník, B. R. Steunebrink, and J. Schmidhuber, "LSTM: A search space odyssey," *arXiv preprint arXiv:1503.04069*, 2015.

[22] K.-A. Lee, A. Larcher, G. Wang, P. Kenny, N. Brümmer, D. A. van Leeuwen, H. Aronowitz, M. Kockmann, C. Vaquero, B. Ma *et al.*, "The reddots data collection for speaker recognition." in *Proc. INTERSPEECH*, 2015, pp. 2996–3000.

[23] T. Kinnunen, M. Sahidullah, M. Falcone, L. Costantini, R. G. Hautamaki, D. A. L. Thomsen, A. K. Sarkar, Z.-H. Tan, H. Delgado, M. Todisco *et al.*, "RedDots replayed: A new replay spoofing attack corpus for text-dependent speaker verification research," in *Proc. ICASSP*, 2017, pp. 5395–5399.

[24] Tomi Kinnunen, Md Sahidullah, Hector Delgado, Massimiliano Todisco, Nicholas Evans, Junichi Yamagishi, Kong Aik Lee, "The ASVspoof 2017 Challenge: Assessing the Limits of Replay Spoofing Attack Detection," *manuscript, submitted to INTERSPEECH 2017*.

[25] M. Schuster and K. K. Paliwal, "Bidirectional recurrent neural networks," *IEEE Transactions on Signal Processing*, vol. 45, no. 11, pp. 2673–2681, 1997.

[26] R. Jozefowicz, W. Zaremba, and I. Sutskever, "An empirical exploration of recurrent network architectures," in *Proc. ICML*, 2015, pp. 2342–2350.

[27] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. ICLR*, 2014.

[28] N. Brümmer and E. de Villiers, "The BOSARIS Toolkit: Theory, Algorithms and Code for Surviving the New DCF,," *arXiv preprint arXiv:1304.2865*, 2013.