SMALL-FOOTPRINT CONVOLUTIONAL NEURAL NETWORK FOR SPOOFING DETECTION

Heinrich Dinkel, *Student Member, IEEE,* Yanmin Qian, *Member, IEEE* and Kai Yu, *Senior Member, IEEE*
Key Laboratory of Shanghai Education Commission for Intelligent Interaction and Cognitive Engineering
SpeechLab, Department of Computer Science and Engineering Brain Science and Technology Research Center
Shanghai Jiao Tong University, Shanghai, China

***Abstract*—Albeit recent progress in speaker verification engendered powerful models, malicious attacks in the form of spoofed speech, are generally not coped with. In previous attempts, deep neural networks were used to extract high dimensional features which were later classified using an independent classifier. Even though the results of this approach are promising, this architecture's disadvantage is it's complexity of optimizing both, neural network and back-end classifier. In this paper we present a simplified neural network approach to address this problem based on the convolutional neural network architecture. Our model concatenates the output of all abstract convolutional representations within the network into a single high-dimensional vector. By preserving all the information within the network, the networks generalization capabilities are greatly enhanced, resulting in an favorable error rate of 5.4 % on the S10 condition. Scores are frame wise obtained by directly extracting the posteriors from the output neurons and further reduced to an utterance score by the use of variance reduction. We show that by using variance posterior score reduction, large performance gains can be achieved. This model outperforms standard feature extracting neural network approaches, in addition on being more versatile, robust and faster to train. Our best model achieves an error rate of 0.7% on the ASVspoof corpus, utilizing common PLP features. It significantly outperforms conventional feature extraction neural networks, while only having 100k parameters.**

I. INTRODUCTION

BIOMETRIC identification describes any method for uniquely distinguishing people by evaluating their biological traits. The oldest and most popular form is the fingerprint, which is known to be unique for each human. Another approach is to use "voice fingerprint", unique features of our own inherent speech, which is the most natural approach to identify any person as an acquaintance or a stranger. Using voice fingerprints can be seen as the next generation method of securing systems, which either replaces or supplements the common concept of arduous memorization of passwords. The performance of automatic speaker verification (ASV) technology has been advanced significantly by channel compensation techniques e.g. i-vector based approaches [1] and deep learning techniques [2], [3]. Even though current research is promising in creating real world applications for speaker identification, the threat of imposing a speaker's speech has not been coped with. In order to address this problem, speaker anti-spoof is a discipline supporting the creation of a well versed ASV system. ASV systems which haven't been prepared for potential malicious spoofing attacks performance degrades heavily under spoofed conditions [4], [5], [6]. Thus the anti-spoof task aims to discriminate between spoofed (artificially or naturally created) and genuine (human) speech. Until this date, [4] four common spoofing types (impersonation, replay, speech synthesis, voice conversion) have been identified. With the advance in available text to speech (TTS) systems, synthesis and voice conversion attacks become increasingly more potent (e.g. google's recent wavenet [7]), thus novel approaches need to be investigated to cancel out these threats.

Spoof detection commonly acts as a binary classification problem: from a speech utterance u, the task is to decide whether u belongs to the genuine speech class hypothesis H_{gen}, or to the spoofed speech class hypothesis H_{spoof}. The decision is based upon the ratio score ω.

$$\omega(u) = \frac{P(u|H_{\text{gen}})}{P(u|H_{\text{spoof}})} \tag{1}$$

In order to protect ASV systems from spoofing attacks, spoofing countermeasures were designed to decide whether a given speech utterance is provided by a genuine speaker or by a spoofed one. Spoof detection can be easily integrated with preexisting ASV systems. One of the key problems, which need to be faced is to find appropriate features which can both, identify spoofed speech but at the same time verify speakers for an ASV task. In this paper we address this problem by introducing a model, capable of detecting spoofed speech, while making use of traditional ASV features, thus making it possible to integrate this model into existing ASV systems.

The remainder of this paper is organized as follows. At first Section II reviews previous work in the context of spoofing detection. Continuing with Section III, which describes the model architecture. Section IV describes our experimental setup, model parameters, used dataset and demonstrates the results while comparing our approach with other neural network anti-spoof techniques. A conclusion is also given in Section V.

This work was supported by the Shanghai Sailing Program No. 16YF1405300, the China NSFC projects (No. 61573241 and No. 61603252) and the Interdisciplinary Program (14JCZ03) of Shanghai Jiao Tong University in China.

978-1-5090-6182-2/17/$31.00 ©2017 IEEE

II. Previous work

In previous works, the main trend for designing efficient countermeasures in this field aims towards creating powerful, spoof aware features. Phase spectra such as cosine phase and modified group delay phase (MGDP) features [8], traditional MFCC [9], [10], [4], inverted MFCC [11], [9] higher order Mel-cepstral coefficients [12] and the recently published q-cepstral coefficients [13], seem to be effective in detecting artificially created speech. Moreover, the results of the recent BTAS2016 [9] challenge focused on replay attacks and has shown that most features, that are capable of detecting synthesized attacks, can also assist in detecting replay attacks. Another interesting investigation was done in [14], which identified three different subbands which seem to contain useful information for spoof discrimination. Work focused on finding suitable models, including i-vector [10], [15], GMM-UBM [9], [13] have been reported to be efficient against artificial spoofing attacks.

In recent work from [16], the first ASV-anti-spoof system was designed and evaluated, where common anti-spoof features were employed to identify if anti-spoof features generalize well on the speaker verification task.

Model	Feature	EER
DNN-LDA [2]	FBANK	2.56
SVM [15]	i-vector, MFCC	1.96
GMM [17]	CFCC+MFCC+IF	1.21
GMM	**Q-cepstral [13]**	**0.2**
DNN-BLSTM Fusion [3]	**FBANK**	**1.1**

TABLE I: Previous results for ASVSpoof [4], bold indicates results published after challenge

Previous results (as seen in Table I) clearly indicate that powerful features commensurate with the final EER performance. The best model of the originally published ASVSpoof2015 challenge used a GMM with a combination of cochlear cepstral, mel frequency and instantaneous frequency (CFCC+MFCC+IF) features. Later, this system was surpassed by q-cepstral features [13], utilizing a similar GMM classifier.

III. Proposed CNN Architecture

We perceive that even though deep neural networks (DNNs) are capable of learning feature representations, they are trained to discriminate between classes, not to enhance the discriminability of it's hidden layers. Feature extracting neural networks have no notion of improving the hidden layers feature accuracy. In addition, the layer at which features are extracted is empirically determined, meaning that a time-consuming trial and error phase needs to be conducted in order to ascertain an appropriate layer. Consequently, features extracted from different layers might result in greatly varying results [3]. Moreover, there is no explicit need to extract features for this task, thus we propose a model which can be used to estimate scores directly. Recently it has been observed that convolutional neural networks (CNNs) have the capability to outperform sequence based LSTM classifiers in speech recognition [3]. In addition, we also make use of padded convolutions, similar to the introduced residual neural networks (ResNet [18]), but avoid a deep structure explicitly. Our main idea is to carry on all information provided by the abstract convolutional layers directly to the final output. Only by focusing on the creation of a network which transforms the input into a high-dimensional space using a non-linearity, which further maps this high-dimensional representation onto the output layer, leads us to our contention that a powerful, yet compact classifier can be created. This classifier keeps all information from the input layers, therefore creating a vector which represents the whole model. We refer to this approach *direct*-CNN (DCNN).

A. Features

In order to showcase our potent model and be able to compare it with other deep neural network approaches[18], traditional Filterbank (FBANK) [19] and perceptual linear predictor (PLP) [20] features were used in this work.

Feature	Window size	Window shift	Static dimension	Dynamic dimension
PLP	25ms	10ms	13	39 $(\Delta + \Delta\Delta)$
FBANK	25ms	10ms	24	48 (Δ)

TABLE II: Feature extraction parameters

Normalization, such as cmvn (cepstral mean variance)[21], are not sought out, since our initial experiments could not achieve a significant gain. Instead, batchnormalization is employed as our main normalization method, which is sufficient to stabilize the mean and variance of the input features. In this task, it is likely that artificially generated spoofed speech (e.g. TTS or VC) contains usually long segments of silence. This silence is kept, since it provides additional information to the model, enhancing it's learning capability. Therefore, the usage of voice activity detection (VAD) is avoided.

B. Model Specification

Our model is based on an idea similar to scatter networks ([22], [23]), which aim to build an invariant representation of an input signal. Suppose that a given feature $x \in R^d$ should be linearly separated. To do so, a transformation ϕ is used, which transforms the feature space into a higher dimensional space $\phi(x) \in R^D$, where $D >> d$. If the space is larger than the number of samples, linear separation can be easily applied [22]. In order to achieve this feat we firstly use three convolutions to expand the existing feature space into a larger dimension. Standard CNN approaches (as in Table III and fig. 2) use non-padded convolutions, which reduced the output feature size, whereas in our approach the input features are padded with zeros during the convolution stage, thus producing an output feature equally sized to the input. The first two convolutions increases the number of feature maps, while keeping the feature size constant. In order to constrain the model to avoid producing an unfeasibly large vector, we use a stride of two during the third convolution, effectively halving the feature size (height and width). After each convolution we apply a non-linearity in form of a ReLU [24] activation

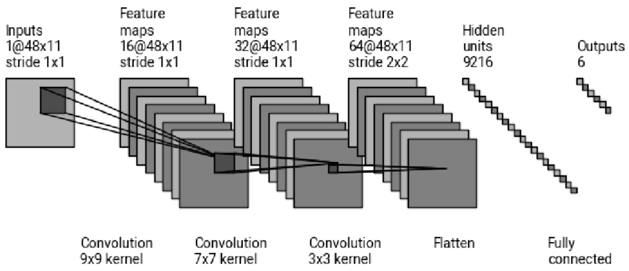


Fig. 1: The proposed DCNN architecture. Note that the input consist a 5 left-right context extended, 48-dim FBANK feature. The context enables learning of a spatial relationship between adjacent frames.

function. This enables the model to learn non-linear behavior and generally be able to generalize on unseen data. We choose to have 16, 32, 64 convolution-filters in each stage respectively (as seen in Figure 1). The three dimensional output (filters × height × width) of the last convolution is flattened into a single high dimensional vector, merging the feature with its filters into a single vector representation. The final layer then transforms the high dimensional vector to the six class output layer. This is the only fully connected layer that is used within the entire framework. The described model only contains 100000 (Table III) parameters. Batch-normalization is employed after each convolution to stabilize the features and increase the overall performance. As optimization method we use adam with a start learning rate of 0.001.

Model	Structure	Activation	# Parameters
DNN	7*1024 FC	ReLU	8M
CNN	64+128 Filters, 3*1024 FC	ReLU	1.8M
DCNN	16+32+64 Filters	ReLU	0.1M

TABLE III: Comparison between baseline models and the proposed DCNN. Filters are the number of feature maps, while FC stands for fully connected layer. Number of parameters are given in million.

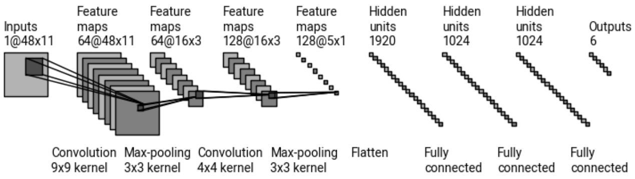


Fig. 2: Baseline CNN

In order to compare the proposed model with other deep learning approaches, standard deep neural and convolutional networks are run. The models can be seen in Table III, whereas the CNN is specified in Figure 2. Throughout all our experiments we decided to use Rectangular Units (ReLU)[24] activations.

C. Scoring

Most approaches in this field which use neural networks classify in classical dvector [25], [26] manner, that is, high-dimensional vector representations are extracted from the neural network and then fed into an independent classifier, such as a SVM or LDA. The downside of this approach is that it extends the overall training time of a system, since two classifiers need to be trained. Moreover by using this approach, it is unclear which classifier (e.g. DNN or LDA) contributed to the final result and how to optimize the overall classification accuracy. In similar work [3], it was observed that resulting EER's can greatly vary depending on the underlying feature extracting neural network. In order to avoid the problems described, the neural network is used to directly produce output probabilities, which will be used as scores for the final classification. More specifically, during our experiments we assume having one genuine class H_{gen} and five spoofed classes, each representing one of the ASVSpoof2015 known spoofing types $H_{spoof} = \{H_{S1}, \ldots, H_{S5}\}$ (see Section IV-A).

$$\text{score}_s = P(H_{gen}|s) \tag{2}$$

$$\text{score}_u = \frac{1}{u_s}\sum_{i=0}^{u_s} \text{score}_{u_i} \tag{3}$$

Given an utterance u with u_s frames and frame s, the neural network to calculates the posterior that this frame belongs to the H_{gen} class (Equation (2)). Then, a final score for each utterance u is calculated by computing the mean over all frame-level output posteriors (Equation (3)). Thus, we assume that the scores for genuine speakers are larger than the ones for spoofed utterances.

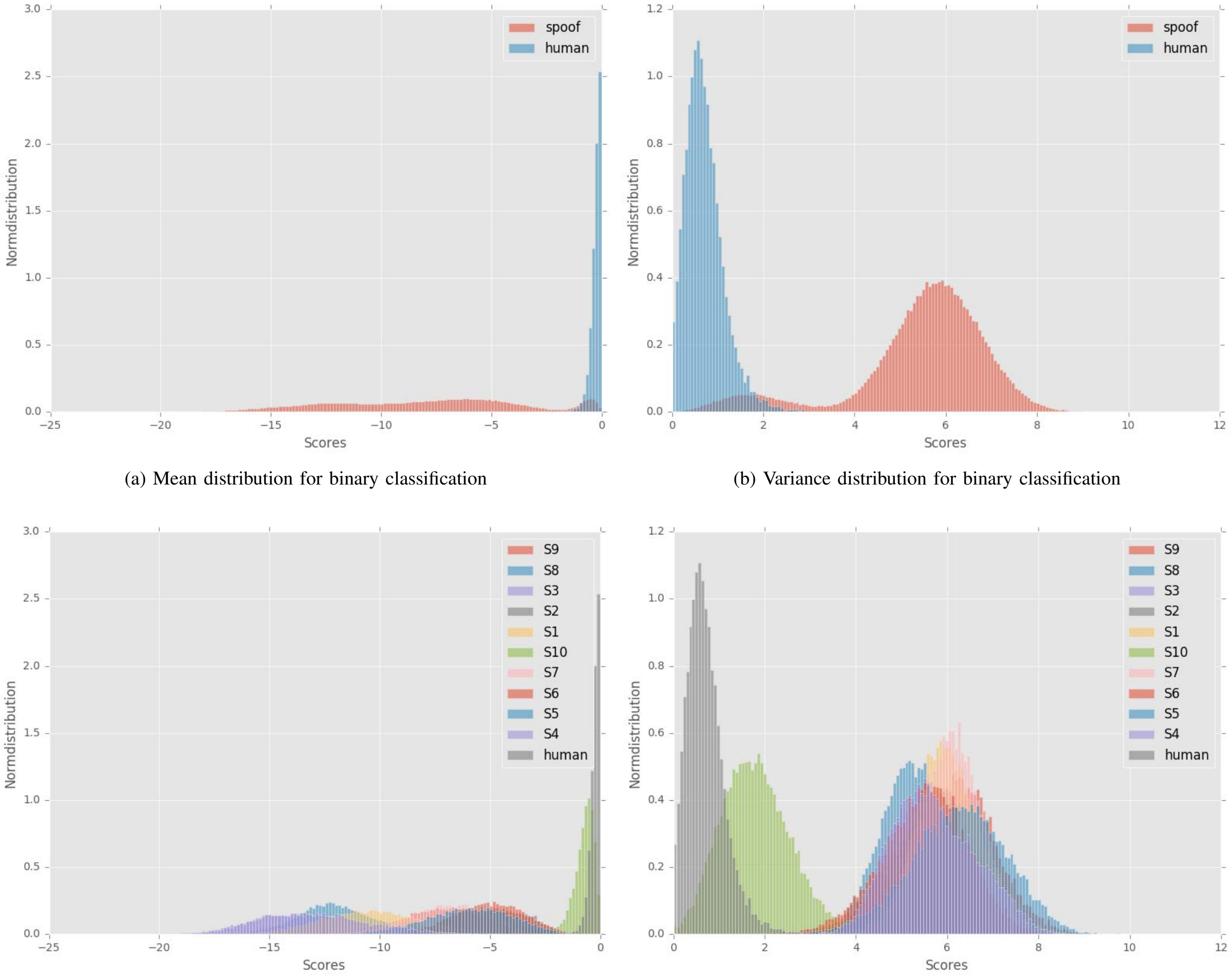
In our research, we also analyzed the score distributions (Figure 3) of our models. We compared the given mean scores with scores calculated by using the standard deviation (variance).

As it is depicted in in Figure 3, the mean and variance genuine classes both look alike, but the spoof classes differ in their shape greatly. Variance based scores seem to be governed by a Gaussian distribution, while mean scores seem to be governed by a uniform distribution. In addition, by comparing Figure 3c with Figure 3d, it can be seen that variance discrimination should outperform mean discrimination in this task, since the majority of the spoof classes (S1-S9) are tightly packed, simplifying the two class classification task. Since the basic assumption for calculating the EER threshold is that the scores in $H_{gen} > H_{spoof}$, we modify the variance scores to be conform with the given constraint by multiplying all scores with -1.

IV. Experiments

A. Dataset

The ASVSpoof2015 dataset is used to run experiments. This dataset focuses mainly on spoofing attacks generated from synthesized and voice converted speech [4], [16]. The spoofed speech is generated by 10 different voice conversion and speech synthesis approaches $(S1, \ldots, S10)$.



(a) Mean distribution for binary classification

(b) Variance distribution for binary classification

(c) Mean distribution for all classes

(d) Variance distribution for all classes

Fig. 3: Score distribution on the example of our baseline dnn. The variance distributions clusters of classes S1-S9 are tightly packed, without any overlap with the human class. The mean distribution clusters for classes S1-S9 are all distinct to identify from each other.

- Voice conversion (VC): S1, S2, S5, S6, S7, S8, S9
- Speech synthesis (SS): S3, S4, S10

The training dataset only contains spoofed speech from the five conditions S1 to S5, thus a potent model needs to generalize well on unseen conditions (S6,. . .,S10) to successfully classify all spoofed utterances. In particular, the condition S10 can be seen as the most difficult in this task, where a speech synthesis system was used with twice the amount of enroll utterances compared to the other conditions.

Two different approaches can be used for classification on this dataset. One either merges all available spoofing classes into one, thus making the task a binary classification problem or uses all classes making it a six-class classification problem. We adapt the six-class classification method, since it has been seen to be beneficial for the final performance [2].

TABLE IV: ASVSpoof2015 dataset structure. Values represent number of utterances. The known and unknown attacks are a subsets of all-attacks.

Type of data	Train	Evaluation
genuine	3750	9404
all-attacks	12625	184000
known-attacks	12625	92000
unknown-attacks	0	92000

B. Evaluation Protocol

Firstly, a model is trained on the training data (Table IV), where we typically run between 5 and 20 epochs of neural network training. The trained model is then tuned on the development dataset. After tuning, posteriors are estimated over the evaluation data. Each utterance is frame wise fed into the network, which outputs posteriors for each respective

Model	Classifier	Known	Unknown	Average
LDA [3]	GDF	2.3	9.8	6.0
DNN	Softmax	0.2	5.4	2.8
CNN	Softmax	0.1	3.9	2.0

TABLE V: The baseline performance EERs (%)

frame. The final scores are obtained by merging the frame scores into a single utterance score. Finally, the produced scores are scored using the common EER metric. In this work, we made use of the well known bob [27] toolkit to calculate the EER. Let ω_{sp} be the threshold between spoofed speech and genuine for the spoof type sp. The subsequent false alarm rate $P_{fa}(\omega_{sp})$ and false rejection rate $P_{rej}(\omega_{sp})$ are defined as:

$$P_{fa}(\omega_{sp}) = \frac{|\text{spoof trials with score} > \omega_{sp}|}{|\text{ all spoofed trials}|} \tag{4}$$

$$P_{rej}(\omega_{sp}) = \frac{|\text{genuine trials with score} < \omega_{sp}|}{|\text{all genuine trials}|} \tag{5}$$

From the Equation (5), we can obtain the final equal error rate (EER) for the spoof type sp, which is the point where $P_{fa}(\omega_{sp}) = P_{rej}(\omega_{sp}) = \text{EER}_{sp}$. In the ASVSpoof challenge, every spoofed class will be calculated independently, which means that 10 different thresholds are obtained. The overall EER is then calculated as the average of all the class specific EERs.

C. Baseline

The baseline is taken from [3], where FBANK features were fed into a LDA classifier, which produced posteriors for each class. These posteriors are used as scores for each class, respectively. Moreover, basic CNN and DNN (Table III) are compared with. The neural network models use a frame extension of 5 to the left and right. Since our proposed direct classification framework produces frame-wise posteriors for each utterance, a reduction is necessary to obtain utterance scores.

$$\text{mean}(\boldsymbol{X}) = \frac{1}{N}\sum_{i=0}^{N} \boldsymbol{x}_i \tag{6}$$

$$\text{variance}(\boldsymbol{X}) = \frac{1}{N}\sum_{i=0}^{N} (\boldsymbol{x}_i - \text{mean}(\boldsymbol{X}))^2 \tag{7}$$

We propose two reduction methods, mean (Equation (6)) and variance (Equation (7)). By comparing the results table V we want to verify the most appropriate reduction method. Given that the frame-wise posteriors from the neural networks are concatenated to a single matrix $\mathbf{X}$ of size $N \times D$, where N represents the number of frames for the current utterance and D the dimension (in this work we always have six outputs), a single score vector of size D for each utterance is estimated by collapsing N frames into a single value using the formulations in Equations (6)(7). A comparison between the two reduction methods can be seen in Table VI.

As we can see from the results (Table VI), variance reduction greatly improves the performance. Therefore for further experiments, variance reduction is utilized.

Model	Meanreduction	Variancereduction
DNN	2.8	2.0
CNN	2.0	1.4
DCNN	1.6	1.0

TABLE VI: Comparison of mean and std features (using FBANK) in EER (%)

Model	Feature	S1	S2	S3	S4	S5	Known
BLSTM-DNN	FBANK	0.0	0.0	0.0	0.0	0.0	0.0
DCNN	FBANK	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**
DCNN	PLP	0.0	0.5	0.1	0.1	0.2	0.2

Model	Feature	S6	S7	S8	S9	S10	Unknown
BLSTM-DNN	FBANK	0.1	0.0	**0.0**	0.0	10.7	2.2
DCNN	FBANK	**0.1**	**0.0**	0.1	**0.0**	9.4	1.9
DCNN	PLP	0.2	0.2	0.1	0.3	**5.4**	**1.2**

Model	Feature	Overall EER
BLSTM-DNN	FBANK	1.1
DCNN	FBANK	1.0
DCNN	PLP	**0.7**

TABLE VII: Comparison between final results for feature extracting BLSTM-DNN Fusion and DCNN

D. Results

Final results can be seen in table table VII. A comparison between the currently best deep feature results [3] and the proposed DCNN are shown in Table VII. The referenced BLSTM-DNN fusion model uses a large context window of 30 frames and obtain scores by using an independent SVM classifier [3].

We achieve our best result using the DCNN model by inserting common PLP features. This result shows that the models capabilities are vast, yet limited by the input features.

E. Discussion

As we can see from the results in Table VII, the proposed DCNN model's performance outperforms the larger BLSTM-DNN fusion model. Using FBANK features leads to an impressive result in all categories, having an average of 0.0% EER on known attacks. We conducted additional experiments using PLP features in order to showcase the model's capabilities. We perceive that the resulting PLP feature gain stems from the correlated nature of PLP features, compared to uncorrelated FBANK ones. We believe that this model could be one of the most compact, yet powerful classifiers for the spoofing task.

V. Conclusion

This paper presents a new classifier approach to the automatic speaker verification spoofing challenge (ASVspoof 2015). The proposed model uses network posteriors for the classification scores. We observed that by using the standard deviation of the scores rather than the mean, a significant gain can be obtained. Our final result outperforms the first ranked model within the original ASVspoof challenge, while being the only system which does employ any score or model fusion attempts. In future work, we like to also incorporate spoof-aware features into our model, which most likely will improve the overall performance significantly.

References

[1] E. Khoury, T. Kinnunen, A. Sizov, Z. Wu, and S. Marcel, "Introducing I-vectors for joint anti-spoofing and speaker verification," *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, no. Cm, pp. 61–65, 2014.

[2] N. Chen, Y. Qian, H. Dinkel, B. Chen, and K. Yu, "Robust deep feature for spoofing detection - the sjtu system for asvspoof 2015 challenge," in *Proc. InterSpeech*, 2015, pp. 2097–2101.

[3] Y. Qian, N. Chen, and K. Yu, "Deep Features for Automatic Spoong Detection," *Speech Communication*, vol. 85, pp. 43–52, 2016. [Online]. Available: http://dx.doi.org/10.1016/j.specom.2016.10.007

[4] 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.

[5] S. Kucur Ergünay, E. Khoury, A. Lazaridis, and S. Marcel, "On the Vulnerability of Speaker Verification to Realistic Voice Spoofing," *IEEE International Conference on Biometrics: Theory, Applications and Systems*, 2015.

[6] N. Evans, T. Kinnunen, J. Yamagishi, Z. Wu, F. Alegre, and P. D. Leon, "Speaker Recognition Anti-spoofing," *Handbook of Biometric Anti-Spoofing*, no. Springer, pp. 1–25, 2014.

[7] A. van den Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. Senior, and K. Kavukcuoglu, "WaveNet: A Generative Model for Raw Audio," pp. 1–15, 2016. [Online]. Available: http://arxiv.org/abs/1609.03499

[8] Z.-z. Wu, E. S. Chng, and H. Li, "Detecting Converted Speech and Natural Speech for anti-Spoofing Attack in Speaker Recognition," in *Proc. 13th Annual Conference of the International Speech Communication Association, (ISCA)*, pp. 2–5.

[9] P. Korshunov, S. Marcel, H. Muckenhirn, A. R. Goncalves, A. G. S. Mello, R. P. V. Violato, F. O. Simoes, M. U. Neto, M. de Assis Angeloni, J. A. Stuchi, H. Dinkel, N. Chen, Y. Qian, D. Paul, G. Saha, and M. Sahidullah, "Overview of BTAS 2016 Speaker Anti-spoofing Competition," in *IEEE International Conference on Biometrics: Theory, Applications and Systems (BTAS)*, Niagara Falls, NY, USA, sep 2016.

[10] S. Weng, S. Chen, L. Yu, X. Wu, W. Cai, Z. Liu, and M. Li, "The SYSU System for the Interspeech 2015 Automatic Speaker Verification Spoofing and Countermeasures Challenge," *Sixteenth Annual Conference of the International Speech Communication Association*, 2015.

[11] S. Chakroborty, A. Roy, and G. Saha, "Improved Closed Set Text-Independent Speaker Identification by combining MFCC with Evidence from Flipped Filter Banks," *International Journal of Signal Processing*, vol. 4, no. 2, pp. 114–121, 2007.

[12] X. Xiao, X. Tian, S. Du, H. Xu, E. S. Chng, and H. Li, "Spoofing speech detection using high dimensional magnitude and phase features: The NTU approach for ASVspoof 2015 challenge," *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, vol. 2015-Janua, pp. 2052–2056, 2015.

[13] M. Todisco, H. Delgado, and N. Evans, "A New Feature for Automatic Speaker Verification Anti-Spoofing: Constant Q Cepstral Coefficients," no. July, 2016.

[14] K. Sriskandaraja, V. Sethu, P. N. Le, and E. Ambikairajah, "Investigation of Sub - Band Discriminative Information between Spoofed and Genuine Speech," pp. 1710–1714, 2016.

[15] S. Novoselov, A. Kozlov, G. Lavrentyeva, K. Simonchik, and V. Shchemelinin, "STC ANTI-SPOOFING SYSTEMS FOR THE ASVSPOOF 2015 CHALLENGE ITMO University , St . Petersburg , Russia," *Icassp 2016*, pp. 5475–5479, 2016.

[16] M. Todisco, H. Yu, T. Kinnunen, N. Evans, and Z.-h. Tan, "Integrated Spoofing Countermeasures and Automatic Speaker Verification : an Evaluation on ASVspoof 2015," pp. 1700–1704, 2016.

[17] T. B. Patel and H. A. Patil, "Combining evidences from mel cepstral, cochlear filter cepstral and instantaneous frequency features for detection of natural vs. spoofed speech," in *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, 2015, pp. 2062–2066.

[18] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," dec 2015. [Online]. Available: http://arxiv.org/abs/1512.03385

[19] Y. Liu, Y. Qian, N. Chen, T. Fu, Y. Zhang, and K. Yu, "Deep feature for text-dependent speaker verification," *Speech Communication*, vol. 73, pp. 1–13, 2015. [Online]. Available: http://dx.doi.org/10.1016/j.specom.2015.07.003

[20] H. Hermansky, "Perceptual linear predictive (PLP) analysis of speech." *The Journal of the Acoustical Society of America*, vol. 87, no. 4, pp. 1738–52, 1990. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/2341679

[21] N. V. Prasad and S. Umesh, "Improved cepstral mean and variance normalization using Bayesian framework," in *2013 IEEE Workshop on Automatic Speech Recognition and Understanding, ASRU 2013 - Proceedings*, 2013, pp. 156–161.

[22] J. Bruna and S. Mallat, "Invariant scattering convolution networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 8, pp. 1872–1886, 2013.

[23] X. Chen and X. Cheng, "Unsupervised Deep Haar Scattering on Graphs," *Neural Information Processing System*, pp. 1–9, 2014.

[24] K. He, X. Zhang, S. Ren, and J. Sun, "Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification," 2014.

[25] N. Chen, Y. Qian, and K. Yu, "Multi-Task Learning for Text-dependent Speaker Verication," in *Proc. InterSpeech*, 2015, pp. 185–189.

[26] E. Variani, X. Lei, E. Mcdermott, I. L. Moreno, and J. Gonzalez-dominguez, "DEEP NEURAL NETWORKS FOR SMALL FOOTPRINT TEXT-DEPENDENT SPEAKER VERIFICATION Google Inc ., USA ATVS-Biometric Recognition Group , Universidad Autonoma de Madrid , Spain," pp. 4080–4084, 2014.

[27] A. Anjos, L. E. Shafey, R. Wallace, M. Günther, C. McCool, and S. Marcel, "Bob: a free signal processing and machine learning toolbox for researchers," in *20th ACM Conference on Multimedia Systems (ACMMM), Nara, Japan*. ACM Press, Oct. 2012. [Online]. Available: https://publications.idiap.ch/downloads/papers/2012/Anjos_Bob_ACMMM12.pdf