УДК 510.9

Speech Replay Spoofing Attack Detection System Based on Fusion of Classification Algorithms*

A.A. Lependin, Y.A. Filin, P.V. Malinin

Altai State University (Barnaul, Russia)

Система обнаружения атак воспроизведением речи, основанная на смеси алгоритмов классификации

А.А. Лепендин, Я.А. Филин, П.В. Малинин

Алтайский государственный университет (Барнаул, Россия)

Fast development of modern technologies of digital processing and speech recording leads to the fact that it is necessary to take into account the potential threats from the speech replay attacks. We propose our ensemble fusion replay attack detection system. It uses constant Q cepstral coefficients as speech features and short-time mean normalization for their preprocessing. The set of binary classifiers includes multiple Gaussian mixture models based Bayesian classifier, i-vector based Gaussian Probabilistic Linear Discriminant Analysis and XGBoost tree boosting algorithm. Fusion of scores was made by modified logistic regression algorithm from BOSARIS toolbox. ASV Spoof 2017 corpus is utilized in the experiments as the main database for anti-spoofing systems evaluation. Obtained results demonstrate that the proposed system can provide substantially better performance than the baseline Gaussian mixture model classifier. The pre-processing of cepstral features is crucial for the better performance of the system. High evaluation performance can be obtained using only few algorithms in a set. The attained value of equal error rate EER=12.44% for our fusion classifier is competitive with the best results obtained during last two years.

Keywords: automatic speaker verification; voice spoofing; replay attacks; universal background model; i-vector; probabilistic linear discriminant analysis; tree boosting; model fusion.

Быстрое развитие современных технологий цифровой обработки и записи речевых сигналов привело к тому, что стал актуальным учет потенциальных угроз, связанных с атаками на биометрические системы аутентификации, которые основаны на воспроизведении речи. В работе предложен подход к детектированию подобных атак при помощи ансамбля из нескольких классификаторов. В качестве информативных признаков речевого сигнала применялись Q-константные кепстральные коэффициенты. Проводилась их нормализация путем вычитания кепстрального среднего, оцениваемого на коротком временном интервале. Множество использованных бинарных классификаторов состояло из алгоритма на гауссовых смесях, гауссового вероятностного линейного дискриминантного анализа в сочетании с извлечением i-векторов речевых сигналов и алгоритма XGBoost. Смешивание осуществлялось при помощи модифицированного алгоритма логистической регрессии. Качество работы предложенного подхода оказалось существенно выше базового метода, основанного на применении смесей гауссовых распределений. Дополнительное существенное улучшение качества было связано с предобработкой кепстральных коэффициентов. Было показано, что качество работы, близкое к наилучшему, может быть достигнуто при смешивании небольшого числа классификаторов. Достигнутое значение частоты ошибок EER = 12.44% для смеси классификаторов близко к лучшим из достигнутых к нынешнему моменту.

Ключевые слова: верификация дикторов, подделка голоса, атаки воспроизведением, универсальная базовая модель, i-вектор, вероятностный линейный дискриминантный анализ, бустинг на деревьях принятия решений, смешивание моделей.

DOI 10.14258/izvasu(2018)1-19

*The work done in this paper is financially supported by RFBR Grant 17-47-220739 r_a.

I. Introduction

Automatic speaker verification (ASV) is one of the key components in modern security systems. It has to meet requirements of robustness to changes in acoustic environment or changes in voice of target speakers. However, fast development of modern technologies of voice generation, digital processing and recording leads to the fact that it is necessary to take into account the potential threats from the so-called voice spoofing attacks [1, 2]. The first and most probable of the attack vectors is a replay attack. It is easy to be performed, and the potential threat it poses to ASV reliability has been confirmed in independent studies. The essence of this kind of attack is simple. The target speaker's voice is recorded by some smart device. Then, this recording is replayed to an ASV system in the place of the genuine speech to unlock smartphone or take access to an application.

Detection of replay attacks using only acoustic characteristics of a given speech utterance is one of the most prominent strategies to spoofing countermeasures. This solution, in prospect, can easily be integrated in modern ASV systems. It is more convenient for users, because verification of their identity and verification of spoofing attack can be done for them in a single step. Although, this strategy is potentially problematic. The difficulty relates to unpredictable variation in quality of a replay attack. Recordings may contain significant additive or convolutional noise, or, in contrast, they can be made with high definition recording hardware. Effective spoofing detection system must consider both possibilities. It has to distinguish acoustic conditions in genuine and recorded voice, noise properties of the acoustic channel, and possible changes in voice spectral characteristics. The promotion of the development of effective replay attack countermeasures was the main motivation for organizing the ASV Spoof Challenge 2017 [1]. This challenge focuses on a standalone replay attack detection. The speech samples corpus for this task consists of nonreplayed (or "genuine") utterances and their replayed ("spoofed") versions. The former subset originates from the RedDots speech corpus [3], whereas the latter was collected by volunteers using smartphones and high definition portable recorders.

In this paper, we describe our replay attack detector. The main idea behind our system is the fusion of scores [4] given by a set of binary classification algorithms to obtain better classification results. We opted for some common speaker modelling techniques such as Gaussian mixture models (GMM) [5], GMM based universal background models (UBM) and i-vectors [6, 7]. The set of classifiers consists of multiple GMMs based classifier [1, 5], i-vector-Gaussian Probabilistic Linear Discriminant Analysis (GPLDA) [7] and highly popular in machine learning community tree boosting method XGBoost [8]. The most common choice of an acoustic feature for voice processing system is Mel frequency cepstral coefficients (MFCCs) [9]. Although, these features are not equally good for the tasks of speech recognition and for spoofing attack detection. In [10], constant-Q cepstral coefficients were proposed for spoofing detection task. We used them instead of MFCCs in our spoofing detection system.

The remainder of this paper is organized as follows. Section 2 introduces our feature extraction studies. The proposed set of classifiers and ensemble learning method are described in section 3. Experimental setup, performance measures and results are discussed in section 4. Section 5 concludes the paper.

II. Features

In modern ASV systems, speech features extraction is based on Fourier transform estimation and some succeeding transformation of obtained spectrum. This traditional approach is not necessarily ideal. Fourier transform is a powerful and widely used tool, but it imposes equally spaced bins in the time-frequency domain. This transformation lacks temporal resolution at higher frequencies. In other words, the selectivity increases when moving from low to high frequencies. In contrast, the constant Q transform (CQT), initially proposed in the field of music processing [11], employs geometrically spaced frequency bins. This ensures the constant selectivity factor across the entire spectrum.

In this study, we used the coupling of the constant Q transform with traditional cepstral analysis. These features are referred to as constant Q cepstral coefficients (CQCCs). Brown introduced them in [11] for the identification of musical instruments with a discrete success. Our version of feature extraction algorithm based on [10] performs a linearization of the frequency scale of the CQT, so that the orthogonality of the DCT basis is preserved. Experimental results in [10] showed that it has much better performance than traditional acoustic features for identification of synthesized and voice conversion spoofing speeches.

The extraction of CQCC features is done as follows [10]. First, the CQT of a time sequence $x(n)$ is obtained. It is defined by:

$$X^{CQ}(k,n)=\sum\nolimits_{j=n-\lceil N_k/2\rceil}^{n+\lceil N_k/2\rceil} x(j)a_k^{*}(j-n+N_k/2), \quad (1)$$

where k=1,2,..., K is the frequency bin index, $a_k^{*}(n)$ is the complex conjugate of basis function $a_k(n)$ and N_k are variable window lengths. The basis functions $a_k(n)$ are complex-valued time-frequency atoms, defined according to:

$$a_k(n)=\frac{1}{C}(\frac{n}{N_k})exp[i(2\pi n\frac{f_k}{f_s}+\Phi_k)]. \quad (2)$$

f_k is the center frequency of the bin k, f_s is the sampling rate, and Φ_k is a phase offset. The scaling factor C is given by:

$$C=\sum\nolimits_{\lceil l=-N_k/2\rceil}^{\lceil N_k/2\rceil} w\left(\frac{l+N_k/2}{N_k}\right), \quad (3)$$

where $w(t)$ is a window function (e.g. Hann window). The center frequencies f_k are defined by:

$$f_k = f_1 2^{\frac{k-1}{B}}, \tag{4}$$

where f_1 is the center frequency of the lowest-frequency bin and B determines the number of bins per octave. Cepstral analysis cannot be applied directly to $X^{CQ}(k)$ since the k bins in this sequence are geometrically spaced. This problem is solved by converting geometric frequency space to linear space. This procedure of signal reconstruction can be viewed as a downsampling operation over the first k bins in low frequencies and as an upsampling operation for the remaining $K-k$ bins in high frequencies. The result is a resampled sequence $X^{CQ}(k)$ at the uniform sample rate F_l given by:

$$F_l = \left[f_1 \left(2^{\frac{k_l-1}{B}} - 1 \right) \right]^{-1} \tag{5}$$

Then, constant Q cepstral coefficients (CQCCs) are extracted the same way as the MFCCs. They obtained from the inverse transformation of the logarithm of the spectrum according to:

$$CQCC(p) = \sum_{l=1}^{L} log \left| X^{CQ}(l) \right|^2 cos \left[\frac{p(l-\frac{1}{2})\pi}{L} \right], \tag{6}$$

where $l = 1,2,\ldots,L-1$ and where l are the newly resampled frequency bins.

Post-processing of CQCC features

The experimental results from [12] demonstrate that performing some feature normalization techniques can effectively improve performance of ASV system. Most of these techniques are applied in the cepstral domain. We used short-time mean normalization (STMN) (without variance normalization) for obtained constant Q cepstral coefficients. The normalized version of our features was given by:

$$C_{STMN}(i,p) = C(i,p) - \mu_{st}(i,p) \tag{7}$$

where i and p represent the frame index and cepstral coefficient index. Short-time mean $\mu_{st}(i,p)$ in the sliding window with length L measured in frames was defined by:

$$\mu_{st}(i,p) = \frac{1}{L} \sum_{k=i-L/2}^{i+L/2} C(i,p) \tag{8}$$

III. Classifiers

Gaussian Mixture Models (Baseline Classifier)

GMM is a weighted sum of N component probabilistic densities defined by:

$$p(x|\lambda) = \sum_{i=1}^{N} p_i b_i(\mathrm{x}), \tag{9}$$

where x is a C-dimensional feature vector, $p_i, i = 1,2,\ldots,M$ represents mixture weights. Each component density with mean vector μ_i and covariance matrix $\sum_i$ is given by:

$$b_i(x) = \frac{1}{(2\pi)^{C/2} \left| \sum_i^{-1} \right|^{1/2}} exp\left\{ -\frac{1}{2}(x-\mu_i)^T \sum_i^{-1} (x-\mu_i) \right\}. \tag{10}$$

Here mixture weights satisfy the constraint $\sum_{i=1}^{N} p_i = 1$.

GMM parameters are represented by $\lambda = \{p_i, \mu_i, \sum_i\}_{i=1}^{N}$.

The model parameters were estimated using expectation maximization (EM) algorithm [5] for each class individually with maximum likelihood (ML) criteria. Two models λ_{replay} and $\lambda_{genuine}$ were built. For the speech feature vector x the score was computed as follows:

$$score(x) = LLK(x|\lambda_{genuine}) - LLK(x|\lambda_{replay}). \tag{11}$$

Here $LLK(x|\lambda)$ is the average likelihood of x given model λ.

$$LLK(x|\lambda) = \frac{1}{T} \sum_{i=1}^{T} \log p(x_i | \lambda), \tag{12}$$

where T represents a number of frames for a given speech signal.

i-Vectors and Gaussian PLDA

All "replayed" and "genuine" labelled training speech utterances were used to train a GMM-based universal background model (UBM) with parameters λ_{UBM}. This model was supposed to be a common label-unrelated speech acoustic space [5]. Given a UBM and training feature vectors of two models for "replayed" and "genuine" speech, acoustic spaces are derived. For this purpose, maximum a posteriori (MAP) adaptation method [5] was used.

Then GMM mean super-vectors [13] were used as input feature vectors for modified joint-factor analysis (JFA) technique. Mean super-vector m is defined as a column vector of dimension CM containing the concatenation of adapted GMM-UBM model mean vectors M_i. It can be decomposed by JFA as follows [13]:

$$m = m_0 + Tw, \tag{13}$$

where m_0 is a mean super-vector for all speakers, T is a total variability matrix, which is a projection matrix to the low-dimensional total variability space simultaneously captures the genuine, spoof and channel variability. Low-dimensional i-vector w is a dense representation of all relevant discrimination information.

The backend of this classifier was a simplified GPLDA algorithm [7]. We used a log likelihood ratio of probabilities of two hypotheses. The first H_1 denotes that two i-vectors belong to the same class and the second H_0 is vice versa:

$$LLR(x_{target}, x_{test}) = \log \frac{P(x_{target}, x_{test} \mid H_1)}{P(x_{target} \mid H_0) P(x_{test} \mid H_0)}. \quad (14)$$

Then, the score was a difference of mean *LLR* for *x* and all "genuine" labelled training utterances and mean *LLR* for x and all "replayed" labelled training utterances, which was defined by:

$$score(x) = \frac{1}{\#G} \sum\nolimits_{x_g \in G} LLR(x_g, x) - \frac{1}{\#R} \sum\nolimits_{x_r \in R} LLR(x_r, x), \quad (15)$$

where *G* and *R* are the genuine and replayed speech utterance sets, *#G* and *#R* are their sizes.

Gradient Boosting

XGBoost is the state-of-the-art tree boosting algorithm. For input labelled examples, tree ensemble model uses K additive functions and predict the output [8]:

$$\hat{y}_i = \sum\nolimits_{k=1}^{K} f_k(x_i), f_k \,\epsilon\, \mathcal{F}, \quad (16)$$

where $\mathcal{F} = \left\{ f(x) = \omega_{q(x)} \right\}$ is the space of all possible CARTs (classification and regression trees), $q : \mathcal{R}^m \to T, \omega \,\epsilon\, \mathcal{R}^T$ is a structure of each tree, ω — weights, *T* — number of leaves in the tree.

This algorithm can return the results in a form of probabilities of given classes. The score for the feature vector *x* with probability to be a genuine $p_{genuine}(x)$ was computed as follows:

$$score(x) = \log p_{genuine}(x) \quad (17)$$

Fusion and Calibration

Model fusion is a mechanism to combine the advantages of different models to further improve the system performance. We use the state-of-the-art approach from Matlab Bosaris toolkit [4]. It provides a logistic regression solution, which can train combination weights to fuse multiple subsystems into a single subsystem, which outputs well-calibrated log-likelihood-ratios. This functionality is provided by optimization of parameters of the following mapping:

$$l(x) = a + \sum\nolimits_{i=1}^{F} b_i s_i(x), \quad (18)$$

where *l*(*x*) is the fused and calibrated output log-likelihood ratio for *x*, *F* is a set of subsystems to be fused, $s_i(x)$ is the score of subsystem *i* [4]. The parameters to be optimized are the scalar offset *a* the scalar combination weights b_i.

IV. Experiments

Dataset description and performance metrics

In this study, we carried out our experiments using ASV Spoof 2017 corpus [1], which is provided as a part of spoofing challenge. It originates from the RedDots corpus [3], which was collected by volunteers from across the globe. Replayed speech utterances were played through one of the 15 transducers of varying quality and recorded by 16 different devices in various combinations.

The database is partitioned into three subsets: training, development, and evaluation. Details of numbers of utterances in each of them are presented in Table I. The first two subsets were provided for the design of replay countermeasures. Besides the primary labels (genuine/replayed), each audio file in the training and development data sets was also provided with information of the text context, speaker, recording environment, playback device, and recording device. Only some of the replay conditions would be the same as those in the training and/or development parts. Majority of the replay attacks would originate from other unseen configurations than those in the training and development parts.

Table I

Number of Samples in ASV Spoof 2017 Database

Type of sample	Subset		
	Train	Development	Evaluation
"genuine" speech	1508	760	1298
"replayed"speech	1508	950	12000

For the training, we used training and development subsets of ASV Spoof Challenge 2017 dataset. Since this competition was over, we had an opportunity to use known labels of the evaluation subset for performance evaluation of our system. We used only the primary labels ("genuine"-"replayed") for training and testing phases.

Performance evaluation

The primary performance measure is the equal error rate (EER) [1, 4]. We obtain the scores from our classification algorithms. Higher scores are assumed to correspond to genuine speech. Let $P_{fa}(\theta)$ and $P_{miss}(\theta)$ denote the false alarm and miss rates at score threshold *θ*:

$$P_{fa}(\theta) = \frac{\#\{replay\ trials\ with\ score > \theta\}}{\#\{total\ replay\ trials\}} \quad (19)$$

$$P_{miss}(\theta) = \frac{\#\{genuine\ trials\ with\ score < \theta\}}{\#\{total\ genuine\ trials\}} \quad (20)$$

These values $P_{fa}(\theta)$ and $P_{miss}(\theta)$ are monotonically decreasing and increasing functions of θ. EER corresponds to the value of θ at which $P_{fa}(\theta) = P_{miss}(\theta)$. EER was estimated using the convex hull method available in Bosaris toolkit [4].

Experimental Setup

As stated above, CQCC features were used in our experiments. For the feature extraction, we followed the typical settings of CQCC extraction function introduced in [10]. The speech signal was analyzed using an overlapping 25-ms Hamming window every 10-ms. We use 30 cepstral coefficients with additional delta and delta-delta coefficients. For the half of our models we use short-time mean normalization. Another half was trained on CQCCs and deltas without post-processing.

GMMs were trained with 512 components and the diagonal covariance matrixes. The dimensionality of i-vectors and the dimensionality of eigenvoice subspace for GPLDA method was 100 and 90 respectively. XGBoost algorithm was used with default settings (number of trees was 1000, maximum depth was 6).

All feature extraction and classification except XGBoost were carried out in MATLAB environment. For GMM-based classification and i-vector extraction we used MSR Identity Toolkit [14], fusion and performance measures were done by Bosaris toolkit. For boosting we used the XGBoost python wrapper.

Experimental Results

The detection error trade-off (DET) curves for all classifiers and their fusions are shown in Figure 1. The comparison of performances is demonstrated in Table II.

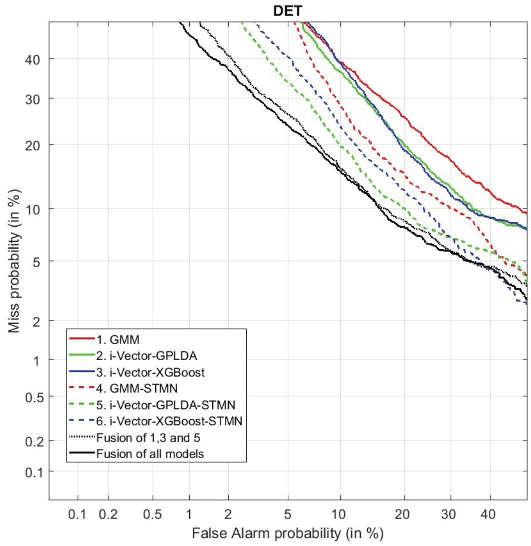

Fig. 1. Detection error trade-off profiles for all models.

Table II

Comparison of models

Classifier	EER, %
GMM (Baseline)	22.47
i-vector-GPLDA	20.00
i-vector-XGBoost	19.42
GMM-STMN	16.70
i-vector-GPLDA-STMN	13.81
i-vector-XGBoost-STMN	15.56
Fusion of GMM + i-vector-XGBoost + i-vector-GPLDA-STMN	12.69
Fusion of all models	12.44
1st place in ASV Spoof Challenge 2017 [1]	6.73
2nd place in ASV Spoof Challenge 2017 [1]	12.34
3rd place in ASV Spoof Challenge 2017 [1]	14.03

We had two versions of used classification algorithms. First three were trained with nonnormalized CQCC features, another three were trained with cepstral mean subtracted features. Results show that all models with STMN CQCCs get much better performance in comparison with their nonnormalized counterparts. The greatest increase in performance was shown by i-Vector-GPLDA classification algorithm. XGBoost was not the best performer in our set.

The best performance was obtained by combination of all classifiers. It is surprising that close EER value could be achieved only by fusing three classifiers (GMM + i-vector-XGBoost + i-vector-GPLDA-STMN). For comparison, top three performance results of ASV Spoof Challenge 2017 [1] are given in Table 2.

V. Conclusion

This work shows that the proposed ensemble fusion system can provide substantially better performance than the GMM baseline for detection the audio replay attacks. The normalization of cepstral features is crucial for better performance of replay attack detecting algorithms. High evaluation performance could be obtained using only few algorithms in a set. The achieved value of EER=12.44% for our fusion classifier is competitive with the best results obtained during ASV Spoof Challenge 2017.

References

1. Kinnunen T., Sahidullah M., Delgado H., Todisco M., Evans N., Yamagishi J., Lee K.A. The ASVspoof 2017 challenge: Assessing the limits of replay spoofing attack detection // Proc. INTERSPEECH 2017. 2017. DOI:10.21437/Interspeech.2017-1111.

2. Wu Z., Yamagishi J., Kinnunen T., Hanilçi C., Sahidullah M., Sizov A., Evans N., Todisco M., Delgado H. ASVspoof: The Automatic Speaker Verification Spoofing and Countermeasures Challenge // IEEE Journal of Selected Topics in Signal Processing. — 2017. — Vol. 11, No. 4. DOI:10.1109/JSTSP.2017.2671435.

3. K. Lee, A. Larcher, G. Wang, P. Kenny, N. Brummer, D. A. van Leeuwen, H. Aronowitz, et al. The RedDots data collection for speaker recognition // Proc. Interspeech, Annual Conf. of the Int. Speech Comm. Assoc., 2015.

4. Morrison G.S. Tutorial on logistic-regression calibration and fusion:converting a score to a likelihood ratio // Australian Journal of Forensic Sciences. — 2013. — Vol. 45, No. 2. DOI: 10.1080/00450618.2012.733025.

5. Reynolds D.A., Quatieri T.F., Dunn R.B. Speaker verification using adapted Gaussian mixture models // Digital Signal Processing. — 2000. — Vol. 10, No. 1. DOI: 10.1006/dspr.1999.0361.

6. Senoussaoui M., Kenny P., Dehak N., Dumouchel P. An i-vector extractor suitable for speaker recognition with both micro-phone and telephone speech // Proc. Odyssey Speaker and Language Recogntion Workshop, 2010.

7. Verma P., Das P.K. I-vectors in speech processing applications: a survey // International Journal of Speech Technolng. — 2015. — Vol. 18, No. 4. DOI: 10.1007/978-981-10-6626-9_18.

8. Chen T., Guestrin C. XGBoost: A Scalable Tree Boosting System // KDD'16 Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. 2016.

9. Shikha G., Jaafar J., Fatimah W., Ahmad W., Bansal A. Feature Extraction using MFCC // International Journal of signal and image processing (SIPIJ). — 2013. — Vol. 4, No. 4. DOI: 10.5121/sipij.2013.4408.

10. Todisco M., Delgado H., Evans N. A new feature for automatic speaker verification anti-spoofing: Constant Q cepstral coefficients // Speaker Odyssey Workshop, Bilbao, Spain. 2016.

11. Brown J. C. Calculation of a constant Q spectral transform // Journal of Acoustic Society America. — 1991. — Vol. 89, No. 1.

12. Alam M., Ouellet P., Kenny P., O'Shaughnessy D. Comparative evaluation of feature normalization techniques for speaker verification // Advances in Nonlinear Speech Processing: 5th International Conference on Nonlinear Speech Processing, NOLISP 2011. DOI: 10.1007/978-3-642-25020-0_32.

13. Dehak N., Kenny P., Dehak R., Dumouchel P., Ouellet P. Front-End Factor Analysis For Speaker Verification // IEEE Transactions on Audio, Speech and Language Processing. — 2010. — Vol. 19, No. 4. DOI: 10.1109/TASL.2010.2064307.

14. Sadjadi S. O., Slaney M., Heck L. MSR identity toolbox v1.0: A MATLAB toolbox for speaker recognition research // Proc. IEEE Signal Process. Soc. Speech Lang. Tech. Committee Newsl. 2013.