# Audio-Replay Attack Detection Countermeasures

Galina Lavrentyeva[1(✉)], Sergey Novoselov[1,2], Egor Malykh[1],
Alexander Kozlov[2], Oleg Kudashev[1,2], and Vadim Shchemelinin[1]

[1] ITMO University, St. Petersburg, Russia
{lavrentyeva,novoselov,malykh,kudashev,shchemelinin}@speechpro.com
[2] STC-innovations Ltd., St. Petersburg, Russia
kozlov-a@speechpro.com

**Abstract.** This paper presents the Speech Technology Center (STC) replay attack detection systems proposed for Automatic Speaker Verification Spoofing and Countermeasures Challenge 2017. In this study we focused on comparison of different spoofing detection approaches. These were GMM based methods, high level features extraction with simple classifier and deep learning frameworks. Experiments performed on the development and evaluation parts of the challenge dataset demonstrated stable efficiency of deep learning approaches in case of changing acoustic conditions. At the same time SVM classifier with high level features provided a substantial input in the efficiency of the resulting STC systems according to the fusion systems results.

**Keywords:** Spoofing · Anti-spoofing · Speaker recognition · Replay attack detection · ASVspoof

## 1 Introduction

In recent years, due to increasing security concerns in all aspects of our daily lives, the need for convenient and non-intrusive authentication methods has grown. Automatic speaker verification (ASV) offers a low-cost and reliable solution for identification problem when voice services are provided. It is already used in social security entitlement, immigration control and election management. Speaker recognition systems are widely used in customer identification during call to a call center, Internet-banking systems and other fields of e-commerce. However, despite the fact that it has reached the point of mass market adoption this technology is acknowledged to be vulnerable to spoofing attacks [1,2].

According to the [3] ASV spoofing are classified into direct and indirect attacks according to the stage they are applied to. Indirect ones attack inner modules (feature extraction module, voice models, classification results). In opposite, direct attacks use the recording stage or transmission level and is focused on the substitution of the input data. Since speaker verification is mostly used in automatic systems without face-to-face contact, direct attacks are more

likely to be used by criminals due to implementation simplicity. The most well-known spoofing attacks are impersonation, replay attack (RA), voice conversion (VC) attack and text-to-speech (TTS) attack [4,5].

Unlike other spoofing types, impersonation does not leave any traces of the recording, playback devices or signal processing, as it is genuine speech. It can be detected by reliable speaker verification system [5]. TTS and VC detection methods were the topic of Automatic Speaker Verification Spoofing (ASVspoof) 2015 Challenge [6]. Results of ASVspoof 2015 confirmed the great potential in detection of VC and TTS. Compared to these spoofing types RA is much more simpler as it does not require specific audio signal processing knowledges. In RA fraudster uses pre-recorded speech samples of the target speaker that can be easily prepared via low-cost recording devices or smartphones. Due to this RA is the most available and therefore critical spoofing technique.

For today there is a small number of studies addressed to the RA detection. The most part of solutions presented for text-dependent ASV are based on the comparison of the test utterance with the stored utterance recorded during the registration. Vulnerability of text-independent ASV to RA was considered in [4,7]. These papers show the serious increase in false acceptance rate of ASV system in case of RA presence. RA detection methods for text-independent case are mostly based on additional noise detection, specific for acoustic conditions.

The ASVspoof Challenge conducted in 2017 was aimed to promote the development of RA countermeasures reliable to both known and unknown conditions [8] which can vary greatly. ASVspoof 2017 was focused on a standalone RA detection task for text-dependent case considered without ASV system and any pre-recorded enroll data.

In this paper we described Speech Technology Center (STC) RA detection systems proposed for ASVspoof 2017. Here we investigated and compared different approaches for spoofing detection. These were Gaussian Mixture Model (GMM) based systems, systems based on high level features with simple classifier and deep learning approaches.

## 2    Automatic Speaker Verification Spoofing Detection Challenge 2017

ASVspoof Challenge was organized in order to assess the potential to detect RA "in the wild", specifically in varying acoustic conditions. For this purpose the spoofing database was collected using text-dependent RedDots data [9]. RedDots corpus was replayed and recaptured in heterogeneous acoustic environment (open lab space, balcony, etc.). For spoofing trials 15 different playback and 16 recording devices were used, including smartphones and high-quality speakers. The original RedDots records were used as genuine speech trials. This dataset was divided into training, development and evaluation parts. The evaluation part contained no information about spoofing trials, devices and recording conditions. Spoofing trials from evaluation part were prepared with the use of devices that were not used in the recording process of the training or development data. In this way they presented previously unforeseen spoofing attacks.

## 3   GMM Approach

Authors of [10] insist that spoofing detection methods should be more focused on the feature engineering rather than on models complication. They proposed the system based on constant Q transform cepstral coefficients (CQCC) and simple GMM. This approach showed impressive results on the ASVspoof 2015 dataset and achieved the 72% improvement over the best system participated in the challenge. That is why the reference implementation of this system was provided as a Baseline system by organisers of ASVspoof 2017.

### 3.1   Baseline

Constant Q transform (CQT) is widely used in music signal processing. By using geometrically spaced frequency bins it overcomes the lack of frequency resolution at lower frequencies and time resolution at higher frequencies that can be produced by Fourier transform with regular space frequency bins. Figure 1 demonstrates the example of CQT spectrum. CQCC estimated according to the scheme in Fig. 2 were used as input features in the Baseline system.

In the Back-End the Baseline system used standard 2-class GMM classifier. 512-component models were trained with an expectation-maximisation (EM) algorithm with random initialisation for genuine and spoofed speech, respectively. For each utterance the log-likelihood score was obtained from GMM models and the final score was computed as the log-likelihood ratio: $\Lambda(X) = \log L(X|\theta_g) - \log L(X|\theta_s)$, where $X$ is a sequence of test utterance feature vectors, $L$ denotes the likelihood function, and $\theta_g$ and $\theta_s$ represent the GMM for genuine and spoofed speech.
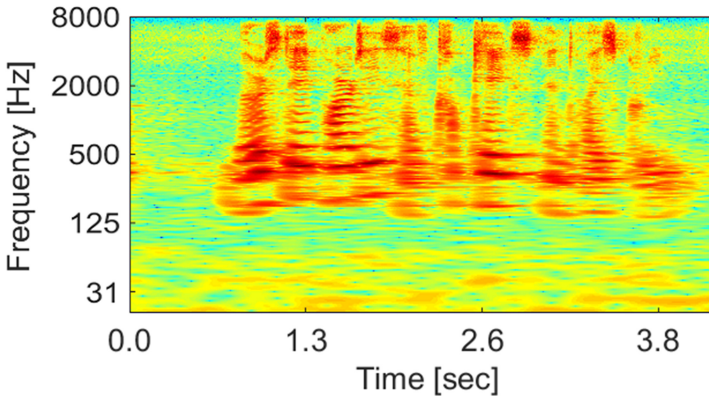


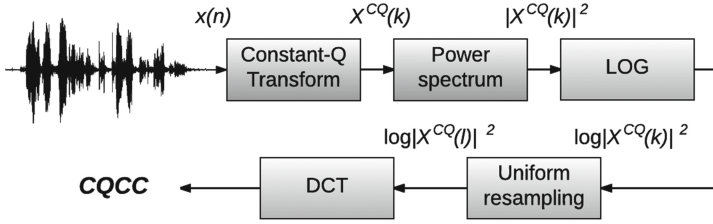**Fig. 1.** Log power magnitude CQT spectrum

**Fig. 2.** CQCC features extraction

### 3.2    Baseline Modifications

The first modifications we tried were normalization techniques for acoustic features. We considered mean and variance normalization on the log power spectrum (mvn) and cepstral coefficients (cmvn) in different combinations. We have also tried acoustic features that were effective for VC and TTS detection. These were Mel Wavelet Packet Coefficients (MWPC), based on applying the multiresolution wavelet transform and phase-based features CosPhasePC [11].

Experiments results obtained on the development part (Table 1) confirm the efficiency of normalization for CQCC based systems. Systems with other Front-End features did not outperform these results.

**Table 1.** Experimental results for GMM-based systems obtained on the train and development parts of ASVspoof 2017 database (EER %)

| Features | CQCC | | | | MWPC | MWPC | CosPhasePC |
|---|---|---|---|---|---|---|---|
| Normalization | - | mvn | cmvn | mvn+cmvn | - | cmvn | - |
| train EER (%) | 0.03 | 1.81 | 1.25 | 1.1 | 0.4 | 0.23 | 0.43 |
| dev EER (%) | 10.35 | 8.74 | 11.86 | 9.85 | 8.81 | 19.17 | 24.64 |

## 4    I-Vector Based System

### 4.1    SVM$_{\text{i-vector}}$

The most efficient spoofing detection systems on ASVspoof 2015 were based on high level features modelling by standard i-vector approach [12]. Inspired by its success for VC ans TTS spoofing detection we proposed similar approach for RA detection. We experimented with variety of acoustic features from ASVspoof 2015, such as MFCC, CosPhasePC and MWPC features. According to our observations the best system was the system based on the Linear Prediction Cepstral Coefficients (LPCC).

78 LPCC coefficients were obtained using the Hanning window function with a 0.128 s window size and a 0.016 s step for FFT power spectrum estimation [13]. I-vectors of the Total Variability space were extracted from the

whole speaker utterances. Here we used the 128-component Gaussian mixture model of the described features for diagonal covariance UBM (Universal Background Model), and the dimension of the T-matrix was 200. After centering and length-normalization i-vectors were used as an input for a linear kernel SVM classifier. For SVM training the efficient LIBLINEAR [14] library was used.

### 4.2   Phrase-Dependent System

During our investigations we suggested that phrase-dependent system can perform a higher accuracy than text-independent system. We compared i-vector based system with the similar systems trained for several different phrases independently. Our experiments presented in Table 2 showed reduction in spoofing detection in comparison with the common systems trained on the whole training dataset. This effect can be explained by the insufficient size of the training data in a phrase-dependent case which leads to fast overfitting.

**Table 2.** Experimental results for the i-vector based system obtained on the train and development parts of ASVspoof 2017 database (EER %)

| UBM | Phrase-dependent | Common | Common | Common |
|---|---|---|---|---|
| T-matrix | Phrase-dependent | Phrase-dependent | Common | Common |
| SVM | Phrase-dependent | Phrase-dependent | Phrase-dependent | Common |
| train EER (%) | 0.398 | 0.133 | 0.401 | 1.459 |
| dev EER (%) | 11.702 | 12.368 | 11.447 | 9.95 |

## 5   Deep Learning Approaches

Deep learning approaches have already achieved remarkable performance in many classification and recognition tasks. The success of CNN in video classification [15], image classification [16,17], face recognition [18] prompts to apply such approaches for ASV anti-spoofing tasks. The idea of employing CNN for spoofing detection is not new and was used in face spoofing detection [19,20].

Several experimental results confirm the efficiency of CNN based approaches for synthetic speech detection. For example in [21] authors demonstrated high efficiency of deep neural network (DNN), CNN and recurrent neural network (RNN) architectures for VC and TTS spoofing detection on the base of ASVspoof 2015 dataset. They also proposed a stacked CNN+RNN architecture and demonstrated its state-of-the-art performance. It is particularly important to note that CNN architecture showed similar to CNN+RNN results and their fusion outperformed the best individual system twice in terms of detection quality.

In [22] temporal CNN architectures were proposed for VC and TTS detection. This approach also achieved notable results on the ASVspoof 2015 corpora.

In this section we describe several systems based on CNN frameworks for RA spoofing detection. Such problem can be reduced to the detection of local spectral artifacts presented in the reproduced replay attack that distinguish it from the genuine speech. For this purpose the CNN was used as a robust feature extractor from the input signal representation in a time-frequency domain.

## 5.1   Unified Shape Time-Frequency Features

In our research we chose 2 types of features based on spectrum estimation of the utterance. To prepare CNN input acoustic features we used the normalized log power magnitude spectrum obtained via:

– constant Q transform (CQT) [10]
– Fast Fourier Transform (FFT)
– Discrete wavelet Transform (DWT), obtained by Daubechies wavelets db4

Special attention should be paid here to the fact that CNN input data should have a unified form. We considered two techniques for obtaining a unified time-frequency (T-F) shape of features. First one truncates the spectrum along the time axis with a fixed size. During this procedure short files are extended by repeating their contents if necessary to match the required length. The other technique uses a sliding window approach with a fixed window size.

## 5.2   Deep Learning Architectures

It is known from image processing that the choice of convolutional neural network architecture is a critical task and greatly affects learning result. In this research we investigated several deep neural network architectures, demonstrated the best results in RA detection.

**Inception CNN Based System $ICNN_{CQT}^{SW}$:** The first proposed neural network architecture was CNN with inception modules (ICNN) ([23]). The proposed architecture was the reduced version of GoogLeNet and contained 3 inception modules. Inception module acts as multiple convolution filter inputs, that are processed on the same input in parallel. It also does pooling at the same time. All resulting feature maps are then concatenated before going to the next layer. This allows the model to pick the best convolutions and take advantages of multi-level feature extraction by recovering both local feature via smaller convolutions and high abstracted features with larger convolutions.

The ICNN was applied for high level feature extraction from the log power magnitude CQT spectrum. The last fully-connected (FC) layer with softmax activation function was used to discriminate between genuine and spoofing classes during the training process. And a low-dimensional high-level audio representation was extracted from the penultimate FC layer.

To obtain unified time-frequency representation of the audio-signal we used sliding window technique with $864 \times 200 \times 1$ window size and 90% overlapping. For ICNN input we applied mean variance normalization.

Since in this space of high level features genuine and replay spoofing classes are well separated, it was enough to use the simplest one-component models for each class distribution modelling. We used the standard 2-class GMM classifier (1 GMM for genuine speech and 1 for spoofed speech) trained on the training part of the ASVspoof 2017 database with EM algorithm. The score for an input signal was computed as the loglikelihood ratio. In this scenario we extracted high-level features independently for each sliding window and all high-level features corresponding to one utterance were used to estimate GMM likelihood. However, it should be mentioned that such deep neural approach can be used for End-To-End solution without additional classifier.

**Light CNN.** The second CNN we explored was the reduced version of the LCNN proposed in [24] with a smaller number of filters in each layer. LCNN consisted of 5 convolution layers, 4 Network in Network (NIN) layers [25], 10 Max-Feature-Map layers, 4 max-pooling layers and 2 fully connected layers [26]. The proposed LCNN used Max-Feature-Map activation function that allows to reduce CNN architecture. In contrast to commonly used Rectified Linear Unit function that suppresses a neuron by a threshold (or bias), MFM suppresses a neuron by a competitive relationship. In this way being applied in particular MFM selects more informative features. Each convolution layer was a combination of two independent convolutional parts calculated for layer's input. MFM activation function was used then to calculate element-wise maximum of these parts. Max-Pooling layers with kernel of size $2 \times 2$ and stride of size $2 \times 2$ were used for time and frequency dimensions reduction. Described CNN was used to obtain high-level audio features similar to the ICNN based system and simple GMM classifier was used at the evaluation stage.

We proposed a number of systems based on LCNN high-level features extractor with different acoustic features. $\text{LCNN}_{\text{CQT}}$ used truncated features obtained from the normalized CQT spectrograms with $864 \times 400 \times 1$ size. Additionally we explored FFT based features instead of CQT: $\text{LCNN}_{\text{FFT}}$ system used truncated features of size $864 \times 400 \times 1$ and $\text{LCNN}_{\text{FFT}}^{\text{SW}}$ was based on the sliding window features extraction with $864 \times 200 \times 1$ window and 90% overlapping along time axis. Alternative system $\text{LCNN}_{\text{DWT}}^{\text{SW}}$ was based on DWT implementation with sliding window of $256 \times 200 \times 1$ size and 83.4% overlapping.

**Stacking CNN and RNN.** We also probed the combined CNN+RNN architecture from [21] for RA spoofing detection. In this stacked architecture CNN was used as a feature extractor and RNN modeled the long-term dependencies of a speech sequence. Both CNN and RNN were optimized jointly through the back-propagation algorithm. In this implementation CNN+RNN was used as End-to-End solution.

CNN was a reduced version of LCNN, but unlike the previous systems max-pooling was applied with the stride 2 along the frequency axis to compress frequency information and stride 1 along the time axis to save time dimensionality.

The RNN part consists of two gated recurrent units [27] forming the bidirectional gated recurrent unit (BGRU). The first GRU was responsible for the forward pass, while the second GRU performed the backward pass. The last output vectors of both forward and backward passes were taken further to obtain two 16-dimensional vectors. Such BGRU unit was applied to each channel of CNN's output resulting in $16 \times 2 \times 8$ tensor. Weights were shared between each channel's unit to prevent overfitting. The flattened output of RNN was used as an input to the fully-connected layers with MFM activations resulting in probability of the utterance being spoofed.

System based on this architecture, $\text{CNN}_{\text{FFT}}+\text{RNN}$, used truncated features extracted from log magnitude power FFT spectrum. But due to the limited computational resources we reduced the input data dimension to $256 \times 400 \times 1$.

Alternatively we used CNN+RNN architecture for $\text{CNN}_{\Delta\text{EEMD}}+\text{RNN}$ system based on the ensemble empirical mode decomposition (EEMD) features. These features were obtained with the use of libEEMD library [28] as follows:

1. Let $S_o$ be the FFT spectrogram of the original signal $x(t)$
2. Get the first empirical mode $c_1(t)$ of the signal $x(t)$ using the EEMD with ensemble size of 50 and noise strength equals to $0.1\sqrt{\text{Var}(x(t))}$
3. Compute $S_r$ as the FFT spectrogram of the signal $c_1(t)$
4. $S_\Delta = |S_o - S_r|$

## 6   Evaluation Results and Discussion

The experimental results of all described individual systems on development and evaluation parts of ASVspoof 2017 corpus are presented in Table 3.

The best result for development and evaluation sets was demonstrated by LCNN system with FFT truncated features. Similar system with CQT features showed poor stability on the evaluation set. This is explained by the poor robustness of CQT features, which is also approved by results of the baseline system.

The sliding window technique demonstrated worse results compared to the truncated approach on the evaluation set. A possible reason for this is that using spectrograms of the whole utterances (in most cases) as CNN input leads to more accurate text-dependent deep model.

Our CNN+RNN combination also performed worse RA detection quality than single LCNN. We explain performance degradation by the reduced frequency resolution in the spectrum estimation.

Summarizing all results of our individual systems we prepared two solutions that use fusion of individual systems on the score level with Bosaris toolkit [29]. Our primary system used fusion of $\text{LCNN}_{\text{FFT}}$, $\text{SVM}_{\text{i-vect}}$ and $\text{CNN}_{\text{FFT}}+\text{RNN}$ systems scores. And our contrastive system combined $\text{SVM}_{\text{i-vec}}$, $\text{ICNN}_{\text{CQT}}^{\text{SW}}$, $\text{LCNN}_{\text{FFT}}$, $\text{LCNN}_{\text{FFT}}^{\text{SW}}$, $\text{LCNN}_{\text{DWT}}^{\text{SW}}$, $\text{CNN}_{\text{FFT}} + \text{RNN}$ and $\text{CNN}_{\Delta\text{EEMD}} + \text{RNN}$ systems.

**Table 3.** Evaluation results

| Individual system | EER (%) | |
|---|---|---|
| | Dev set | Eval set |
| Baseline | 10.35 | 30.6 |
| $\text{Baseline}_{\text{MVN+CMVN}}$ | 9.85 | 17.31 |
| $\text{SVM}_{\text{i-vec}}$ | 9.80 | 12.54 |
| $\text{ICNN}_{\text{CQT}}^{\text{SW}}$ | 10.74 | 15.11 |
| $\text{LCNN}_{\text{FFT}}$ | **4.53** | **7.37** |
| $\text{LCNN}_{\text{FFT}}^{\text{SW}}$ | 5.25 | 11.81 |
| $\text{LCNN}_{\text{CQT}}$ | 4.80 | 16.54 |
| $\text{LCNN}_{\text{DWT}}$ | 8.71 | 16.08 |
| $\text{CNN}_{\text{FFT}}$ + RNN | 7.51 | 10.69 |
| $\text{CNN}_{\Delta\text{EEMD}}$ + RNN | 9.94 | 18.90 |
| *Fusion system* | | |
| Primary: $\text{LCNN}_{\text{FFT}}$, $\text{SVM}_{\text{i-vec}}$, $\text{CNN}_{\text{FFT}}$+RNN | **3.95** | **6.73** |
| Contrastive: $\text{SVM}_{\text{i-vec}}$, $\text{ICNN}_{\text{CQT}}^{\text{SW}}$, $\text{LCNN}_{\text{FFT}}$, $\text{LCNN}_{\text{FFT}}^{\text{SW}}$, $\text{LCNN}_{\text{DWT}}^{\text{SW}}$, $\text{CNN}_{\text{FFT}}$+RNN, $\text{CNN}_{\Delta\text{EEMD}}$+RNN | 2.77 | 7.56 |

Comparing two fusion systems on the development and evaluation parts we can see that complicated fusion of 7 systems have less performance than simpler fusion of three systems which are completely different in architectures, features and presumably detect different artifacts. Despite the noticeable quality reduction for some individual systems on the eval part, the difference in 1% of EER for fusion systems is explained by the impressive result of $\text{LCNN}_{\text{FFT}}$. According to Bosaris fusion model, this system has the biggest weight in both fusion solutions. However, we suppose that complex system will better detect RA spoofing with more various conditions.

## 7    Conlusions

In this paper we explored the applicability of several different approaches for replay attack spoofing detection. We investigated state-of-the-art methods from VC and TTS spoofing detection and deep learning approaches. Our experiments conducted on the ASVspoof 2017 dataset confirmed high efficiency of deep learning frameworks for spoofing detection "in the wild". EER of the best individual CNN system was 7.34%. At the same time SVM classifier with high level features provides a substantial input into the efficiency of the resulting STC systems according to the fusion systems results. Our primary system based on systems score fusion provided 6.73% EER on the evaluation set.

# References

1. Marcel, S., Nixon, M.S., Li, S.Z. (eds.): Handbook of Biometric Anti-Spoofing. ACVPR. Springer, London (2014)
2. Shchemelinin, V., Simonchik, K.: Study of voice verification system tolerance to spoofing attacks using a text-to-speech system. Instrum. Eng. **57**(2), 84–88 (2014). in Russian
3. Faundez-Zanuy, M., Hagmller, M., Kubin, G.: Speaker verification security improvement by means of speech watermarking. Speech Commun. **48**(12), 1608–1619 (2006)
4. Villalba, J., Lleida, E.: Speaker verification performance degradation against spoofing and tampering attacks. In: FALA 2010, pp. 131–134 (2010)
5. Wu, Z., Evans, N., Kinnunen, T., Yamagishi, J., Alegre, F., Li, H.: Spoofing and countermeasures for speaker verification: a survey. Speech Commun. **66**, 130–153 (2015)
6. Wu, Z., Kinnunen, T., Evans, N., Yamagishi, J., Hanilçi, C., Sahidullah, M., Sizov, A.: ASVspoof 2015: the first automatic speaker verification spoofing and countermeasures challenge. Training **10**(15), 3750 (2015)
7. Villalba, J., Lleida, E.: Preventing replay attacks on speaker verification systems. In: IEEE International Carnahan Conference on Security Technology, ICCST 2011, pp. 284–291. IEEE (2011)
8. 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. In: Interspeech 2017 (2017, submitted)
9. Lee, K.A., Larcher, A., Wang, G., Kenny, P., Brmmer, N., van Leeuwen, D.A., Aronowitz, H., Kockmann, M., Vaquero, C., Ma, B., Li, H., Stafylakis, T., Alam, M.J., Swart, A., Perez, J.: The reddots data collection for speaker recognition. In: INTERSPEECH, ISCA, pp. 2996–3000 (2015)
10. Todisco, M., Delgado, H., Evans, N.: A new feature for automatic speaker verification antispoofing: constant Q cepstral coefficients. In: Processings of Odyssey (2016)
11. Novoselov, S., Kozlov, A., Lavrentyeva, G., Simonchik, K., Shchemelinin, V.: STC anti-spoofing systems for the ASVspoof 2015 challenge. In: 2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (2016)
12. Li, M., Zhang, X., Yan, Y., Narayanan, S.: Speaker verification using sparse representations on total variability i-vectors. In: Processings of INTERSPEECH (2011)
13. P., D., Ellis, W.: PLP and RASTA (and MFCC, and inversion) in Matlab (2005)
14. Wu, T., Wang, C.C., Yu, H.F.: Liblinear library. http://www.csie.ntu.edu.tw/cjlin/liblinear
15. Karpathy, A., Toderici, G., Shetty, S., Leung, T., Sukthankar, R., Fei-Fei, L.: Largescale video classification with convolutional neural networks. In: Proceedings of the IEEE Conference on CVPR 2014, pp. 1725–1732 (2014)
16. Bengio, Y., Courville, A., Vincent, P.: Representation learning: a review and new perspectives. IEEE Trans. Pattern Anal. Mach. Intell. **35**(8), 1798–1828 (2013)

17. Krizhevsky, A., Sutskever, I., Hinton, G.E.: ImageNet classification with deep convolutional neural networks. In: Advances in Neural Information Processing Systems, pp. 1097–1105 (2012)
18. Taigman, Y., Yang, M., Ranzato, M., Wolf, L.: Deepface: closing the gap to human level performance in face verification. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1701–1708 (2014)
19. Galbally, J., Marcel, S., Fierrez, J.: Biometric antispoofing methods: a survey in face recognition. IEEE Access **2**, 1530–1552 (2014)
20. Yang, J., Lei, Z., Li, S.Z.: Learn convolutional neural network for face anti-spoofing. arXiv preprint arXiv:1408.5601 (2014)
21. Zhang, C., Yu, C., Hansen, J.H.L.: An investigation of deep learning frameworks for speaker verification anti-spoofing. IEEE J. Sel. Top. Sign. Process. **4**(99), 684–694 (2011)
22. Tian, X., Xiao, X., Siong, C.E., Li, H.: Spoofing speech detection using temporal convolutional neural network. In: 2016 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA) (2016)
23. Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A.: Going deeper with convolutions. In: Computer Vision and Pattern Recognition (2015)
24. X., X.W., He, R., Sun, Z., Tan, T.: A light CNN for deep face representation with noisy labels. IEE J. Sel. Top. Sign. Process. (2015)
25. Lin, M., Chen, Q., Yan, S.: Network in network. CoRR abs/1312.4400 (2013)
26. Lavrentyeva, G., Novoselov, S., Malykh, E., Kozlov, A., Kudashev, O., Shchemelinin, V.: Audio replay attack detection with deep learning frameworks. In: Interspeech-2017 (2017, submitted)
27. Chung, J., Gülçehre, Ç., Cho, K., Bengio, Y.: Empirical evaluation of gated recurrent neural networks on sequence modeling. CoRR abs/1412.3555 (2014)
28. Luukko, P., Helske, J., Rsnen, E.: libeemd library. https://bitbucket.org/luukko/libeemd
29. Brmmer, N., de Villiers, E.: Bosaris toolkit. https://sites.google.com/site/bosaristoolkit