www.czasopisma.pan.pl PAN www.journals.pan.pl
Archives of Acoustics Vol. 47, No. 2, pp. 181–189 (2022), doi: 10.24425/aoa.2022.141648

Research Paper

Spoofed Speech Detection with Weighted Phase Features and Convolutional Networks

Gökay DİŞKEN

Department of Electrical-Electronics Engineering
Adana Alparslan Türkeş Science and Technology University
Adana, Turkey; e-mail: gdisken@atu.edu.tr

(*received February 1, 2021; accepted February 18, 2022*)

Detection of audio spoofing attacks has become vital for automatic speaker verification systems. Spoofing attacks can be obtained with several ways, such as speech synthesis, voice conversion, replay, and mimicry. Extracting discriminative features from speech data can improve the accuracy of detecting these attacks. In fact, a frame-wise weighted magnitude spectrum is found to be effective to detect replay attacks recently. In this work, discriminative features are obtained in a similar fashion (frame-wise weighting), however, a cosine normalized phase spectrum is used since phase-based features have shown decent performance for the given task. The extracted features are then fed to a convolutional neural network as input. In the experiments ASVspoof 2015 and 2017 databases are used to investigate the proposed system's spoof detection performance for both synthetic and replay attacks, respectively. The results showed that the proposed approach achieved 34.5% relative decrease in the average EER for ASVspoof 2015 evaluation set, compared to the ordinary cosine normalized phase features. Furthermore, the proposed system outperformed the others at detecting S10 attack type of ASVspoof 2015 database.

Keywords: spoofing detection; cosine normalized cepstrum; convolutional neural networks.

1. Introduction

Automatic speaker recognition (ASR) systems, where the aim is to verify or identify a person from his/her voice biometrics, have witnessed great performance improvements in the last decades with the development of methods such as Universal Background Model (UBM) (Reynolds *et al.*, 2000) and *i*-vectors (Dehak *et al.*, 2011). On the other hand, spoofing attacks are proven to be detrimental to ASR systems (De Leon *et al.*, 2012; González Hautamäki *et al.*, 2015; Wu *et al.*, 2012). The vulnerability of ASR systems is a serious threat, since high quality spoofing attacks can be obtained with the development of related technology in both hardware and software manners. Furthermore, the mass adoption of the ASR systems can attract attention of malicious individuals. Open source algorithms or recorded speech of a target speaker can be used for spoofing attacks with a little or even no expertise in the field.

The ASVspoof Challenges provided researchers large scale databases with several attack types and known/unknown attack conditions, which aid to the developments of counter-measures for spoof detection (Kinnunen, 2017; Todisco *et al.*, 2019; Wu *et al.*, 2017). Conventional methods such as Mel-Frequency Cepstral Coefficients (MFCCs) and Gaussian Mixture Models (GMMs) can be used for spoofed detection. However, many different methods have been developed with superior performances. For example, Constant-Q Cepstral Coefficients (CQCCs) (Todisco *et al.*, 2017) is found to be one of the best features for spoof detection, and given as a baseline method for the ASVspoof 2017 and 2019 challenges. CQCCs include constant-Q transform to obtain time-frequency representation of the speech signals. Further modifications to CQCCs to increase its performance are proposed in (Yang, Das, 2020) and (Yang *et al.*, 2018). On the other hand, it is shown that simply increasing the number of filters and cepstral coefficients for MFCC can generate a high performance (Chen *et al.*, 2018).

Besides the MFCC and CQCC features, many other magnitude-based features can be found in the literature for spoof detection (Font *et al.*, 2017;

Sahidullah *et al.*, 2015). The main reason for using the magnitude-based features may be the fact that the phase spectrum is usually neglected in speech and speaker recognition systems. Hence, researchers usually focus on the magnitude spectrum to extract useful information. However, phase features can also provide traits to detect spoof attacks, or at least they may provide complementary information to magnitude features (Jung *et al.*, 2019; Liu *et al.*, 2019; Tian *et al.*, 2016). Such combinations of phase-based features are used for spoof detection in (Patel, Patil, 2015; Singh, Pati, 2019), and (Srinivas *et al.*, 2018). Wu *et al.* (2012) showed that the cosine normalized phase spectrum (or cosine phase in short) and Modified Group Delay (MGD) features can outperform MFCCs. In (Liu *et al.*, 2015), local binary pattern, cosine phase, and MGD features are combined to achieve a better performance than each sub-system. A group delay based phase spectrum is found to be more effective to detect replay attacks than several other features (Cai *et al.*, 2019). A magnitude-phase spectrum, where both spectra are used to extract features, has performed better than MFCCs and CQCCs (Yang, Liu, 2018). Features obtained from instantaneous frequency, which is time derivative of phase, achieved a higher performance for replay attack detection task than MFCC, CQCC, and MGD features (Rafi, Murty, 2019). Various phase and magnitude based features are examined and fused in (Xiao *et al.*, 2015), where very high dimensional (13056) feature vectors are obtained for each frame by considering 51 consecutive frames. Fused system achieved almost a perfect score for the ASVspoof 2015 evaluation data, expect the S10 condition, where Equal Error Rate (EER) is 26.1. In (Tom *et al.*, 2018), group delay features and convolutional networks with attention are used to achieve 0.0% EER value for the ASVspoof 2017 database version 1.0.

The aforementioned studies reveal the importance of phase related features. Recently, it is reported that the discriminative power of the features can be enhanced with weighting schemes (Yang *et al.*, 2019; 2020). The proposed approaches are combined with constant-Q transform and applied to magnitude spectrum. Improvements over the standard CQCCs are observed in both studies. Hence, regarding the importance of the phase spectrum, the performance of the frame-wise weighting for the cosine normalized phase features is investigated in this paper.

Although the performance of the extracted features is critical for detecting the spoofed speech, the other part of the system consists classifiers, where suitable models are built by using the features from the training data. Similar to the features, a large variety of methods can be found for classifiers. A comparison between GMMs, generalized linear discriminant sequence kernel, and *i*-vector can be found in (Hanilçi *et al.*, 2015) for ASVspoof 2015 database. On the other hand, a direct comparison between different classifiers found in the literature may not be meaningful due to the different databases, different features, and different configuration of parameters. However, a general trend can be seen from the results of the challenges. For example, ASVspoof 2015 results indicate that many participants preferred GMM or *i*-vector based classifiers (Wu *et al.*, 2017). Contrary to this, more and more studies include deep learning approaches in the recent ASVspoof 2019 challenge (Alzantot *et al.*, 2019; Białobrzeski *et al.*, 2019; Chang *et al.*, 2019; Gomez-Alanis *et al.*, 2019; Jung *et al.*, 2019; Zeinali *et al.*, 2019). Considering the superior performance of the deep learning architectures in diverse areas, researchers inherently examined them in spoof detection tasks. With the same concerns, convolutional neural networks (CNN) employed as classifiers in this paper.

The performance of the proposed system (weighted cosine phase features – CNN) is analyzed using ASVspoof 2015 and 2017 version 2.0 databases. The ASVspoof 2015 database includes speech synthesis and voice conversion attack types, while the ASVspoof 2017 focuses on the replay attack scenario. The remaining of the paper is organized as follows: in Sec. 2 details of the proposed approach are given from both feature and classifier perspectives, Sec. 3 shows the experimental setup and results, Sec. 4 includes a discussion of the results, and Sec. 5 concludes the paper.

2. Proposed approach

In this section, the proposed system is explained through two subsections. In the first subsection, frame-wise weighted cosine normalized phase features are introduced with the extraction steps. In the second subsection, the details of CNN classifier employed to detect spoofing attacks using the extracted features are given. The reasons behind choosing these methods are also explained in the respective subsections.

2.1. Weighted cosine phase cepstral features

A frame-wise weighting approach is proposed in (Yang *et al.*, 2019) for magnitude spectrum obtained via constant-Q transform, and achieved lower EER values than MFCCs and CQCCs, with a common Deep Neural Networks (DNN) classifier. The main idea behind the weighting is to increase the Fisher ratio between two classes, as given with Eq. (1):

$$F_{C_1C_2} = \frac{(m_{C_1} - m_{C_2})^2}{v_{C_1} + v_{C_2}}, \tag{1}$$

where C_1 and C_2 represent the classes, m_{C_1}, m_{C_2}, v_{C_1}, and v_{C_2} represent the means and variances of C_1 and C_2, respectively. $F_{C_1C_2}$ is the Fisher Ratio between the

classes. Assuming X and Y are the magnitude spectra of a genuine frame and spoofed speech frame as

$$X = x_1, x_2, ..., x_K, \qquad Y = y_1, y_2, ..., y_K, \tag{2}$$

where $x_1, x_2, ..., x_K$ and $y_1, y_2, ..., y_K$ are frequency coefficients of each spectrum, respectively, and K is the total bin number. The sums along frequency coefficients are represented by S_X and S_Y, and defined as

$$S_X = \sum_{k=1}^{K} x_k, \qquad S_Y = \sum_{k=1}^{K} y_k. \tag{3}$$

Yang *et al.* (2019) found out that multiplying each magnitude spectrum frame with its sum along frequency coefficients, leads to increased discriminative power, i.e. a bigger Fisher ratio value. That means, instead of using X and Y, the authors used X' and Y' which are defined as

$$\begin{aligned} X' &= S_x x_1, S_x x_2, ..., S_x x_K, \\ Y' &= S_y y_1, S_y y_2, ..., S_y y_K, \end{aligned} \tag{4}$$

to enlarge discriminative information between genuine and replayed speech. The detailed mathematical proof is given in the reference for interested readers. A similar approach is derived in (Yang *et al.*, 2020), using mean-based and variance-based log magnitude spectra. The modified log magnitude improved the results, compared to the sum along frequency coefficients method.

When the given equations are examined, it can be seen that the frame-wise weighting does not rely on magnitude spectrum. In fact, it can be included at different steps of feature extraction process. Therefore, phase features can also benefit from a weighting method. In order to verify this, the cosine normalized phase features (Wu *et al.*, 2012) are chosen in this work. To extract these features, a cosine function is applied to the unwrapped phase spectrum. Then, a discrete cosine transform is applied to obtain cepstral coefficients. Although there are many other phase based features, the cosine phase features are chosen due to their straightforward implementation. It is assumed that the benefits of weighting approach can be easily observed, since there are only a few steps in the extraction process, any improvement in the results can be mainly attributed to the frame-wise weighting. The cosine function maps its inputs into [−1, 1] interval, hence, the modified log weighting of Yang *et al.* (2020) cannot be used due to the negative values of the phase spectrum, before or after cosine normalization. The proposed feature extraction process is summarized as a block diagram in Fig. 1. The frame-wise weighting is applied after the cosine function. Although it may be applied directly to the phase spectrum, mapping the results into [−1, 1] interval with the cosine function may hinder the potential improvements. Therefore, once the cosine normalized phase spectrum is obtained, the frame-wise weighting is applied by summing each frequency coefficient of the frame as explained previously. Then, the discrete cosine transform is applied to extract cepstral features.

In the previous studies, various dimensional cosine phase-based features have been used with different classifiers. In (Alam *et al.*, 2015), 12-dimensional cosine phase features are combined with 12-dimensional MFCC and log energy, and GMM is used as a classifier. Voice activity detection is also applied to remove non-speech frames. In (Wu *et al.*, 2012), 12-dimensional cosine phase features are used with a GMM classifier. In (Novoselov *et al.*, 2016), first and second order derivatives are included along 12-dimensional cosine phase features and a support vector machine is chosen as the classifier. In (Hanilçi *et al.*, 2016), GMM and *i*-vector are considered as classifiers for 32-dimensional cosine phase features. 57-dimensional features were used in (Hanilçi, 2018a), and 60-dimensional features were used in (Sahidullah *et al.*, 2015).

Since those number of features and their variations (such as delta, acceleration, log energy, voice activity detection, cepstral mean normalization, etc.) make it hard to determine a fixed number of coefficients for optimal performance, using high dimensional features seems to be an effective way. In (Tian *et al.*, 2016) and (Xiao *et al.*, 2015), very high dimensional vectors are extracted from both magnitude and phase spectrums, and neural networks are used to handle those high dimensional inputs. In (Tom *et al.*, 2018), phase spectrum-based group delay grams are constructed and fed to a CNN to detect replay attacks. Both of these high dimensional systems resulted in very high performing systems (2.62% EER on ASVspoof 2015 and 0% EER for ASVspoof version 1.0, respectively). Inspired by the success of high dimensional features and modelling capacity of deep learning models, 128-dimensional cepstral features and CNNs are used in this work. The next subsection gives the details of the CNN models.

2.2. Convolutional neural networks

CNN is one of the most popular deep learning architectures, and used for spoofed speech detection in many studies with various combinations (Cai *et al.*, 2019; Chettri *et al.*, 2020; Dinkel *et al.*, 2017;

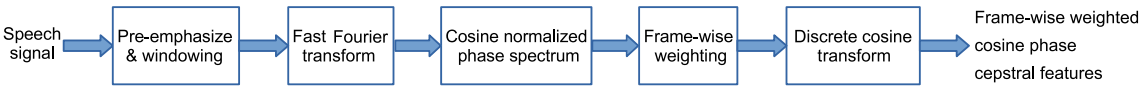

Fig. 1. Extraction process of the proposed frame-wise weighted cosine phase features.

Gomez-Alanis *et al.*, 2019; Jung *et al.*, 2019; Tom *et al.*, 2018; Zhang *et al.*, 2017). It should be noted that more complex models such as ResNet (Tom *et al.*, 2018), combination of CNNs with recurrent neural networks, etc. may boost the system's performance. However, in this work, our main goal is to investigate the effectiveness of frame-wise weighting approach on phase features. Therefore, a conventional CNN model is preferred following the setup given in (Zhang *et al.*, 2017).

Similar CNNs are used for both ASVspoof 2015 and 2017 databases, except the output layer where six outputs are used for ASVspoof 2015 database, and two outputs are used for ASVspoof 2017 database to represent genuine and spoof attack types. Four convolutional layers with different filter sizes and filter numbers are used. Pooling layers are added to each convolutional layer for down sampling. After the last convolutional layer, a fully-connected layer is used which is connected to the softmax output layer. The number of neurons in the fully-connected layer is 1024 for ASVspoof 2015 database, and 64 for ASVspoof 2017 database. Since the ASVspoof 2017 is relatively smaller than the other database, it is found that a smaller number of neurons fit better to the data. Batch normalization and ReLU activation function are used. 50% dropout is added to the fully-connected layer. A summary of the CNN parameters is given in Table 1.

Table 1. CNN architecture for the proposed system.

	Type	Size/# of filters
Layer 1	Convolutional pooling	$7\times7/16$ 3×3
Layer 2	Convolutional pooling	$5\times5/32$ 3×3
Layer 3	Convolutional pooling	$3\times3/32$ 3×3
Layer 4	Convolutional pooling	$3\times3/32$ 3×3
Layer 5	Fully-connected	1024 (64 for ASVspoof 2017)
Layer 6	Sofmax output layer	6 (2 for ASVspoof 2017)

3. Experiments

In this section, details of the databases are given first, then experimental setup is described. Finally, the results are provided for each database separately, a comparison between the proposed systems and similar previous works on the same database is also included.

3.1. Datasets

As mentioned previously, ASVspoof 2015 and ASVspoof 2017 version 2.0 were used in the experiments. The ASVspoof 2015 data includes synthetic speech attacks, and the ASVspoof 2017 data includes replayed speech attacks. Sampling rate of the utterances is 16 kHz for both databases. The statistics of each database are given in Tables 2 and 3. It should be noted that ASVspoof 2019 database is also publicly available, and it includes both the synthetic and the replayed speech attacks. However, to the best of the author's knowledge, any study on the ASVspoof 2019 data including the cosine phase features is not present yet. Therefore, only 2015 and 2017 data were considered in this work to make a comparison between the proposed system and other cosine phase feature based systems found in the literature.

Table 2. Statistics of ASVspoof 2015 data; number of available utterances *per class* and their portions in the database.

Class	Segments		
	Train	Development	Evaluation
Genuine	3750	3497	9404
S1 to S5	2525	9975	18400
S6 to S10	0	0	18400

Table 3. Statistics of ASVspoof 2017 data; number of available utterances *per class* and their portions in the database.

Class	Segments		
	Train	Development	Evaluation
Genuine	1507	760	3565
Spoof	1507	950	14465

In the ASVspoof 2015 data, 10 different spoofing attacks are present. Five of them (S1–S10) are included in each part of the database (train., dev., eval.) and called as known attack types. The rest is only available in the evaluation data and is called as unknown attack types. All attack types use the same STRAIGHT vocoder, except S5 and S10 attacks. S5 uses mel-log spectrum approximation vocoder, and S10 uses a more complex diphone concatenation method. Among all of the available attacks, S10 is considered as the hardest one to detect, as it overlaps with the genuine speech data (based on *i*-vector representations), and it is not present in the training partition. Therefore, the classifiers can misinterpret it as genuine speech. Interested readers can refer to Fig. 2 of Wu *et al.* (2017) for a visual representation, and also for the further details of the speech synthesis algorithms.

The ASVspoof 2017 version 2.0 data, consists of replayed speech attacks, obtained via replaying and recording genuine utterances using various devices and in various acoustics environments. The version 1.0 data includes data anomalies such as zero-valued samples and silence periods, which affects the detection performance (Delgado *et al.*, 2018). Those files were fixed in the version 2.0 data.

3.2. Experimental setup

For each database, the CNN model given in Subsec. 2.2. is used with the indicated parameters. For the feature extraction, the audio signals are divided into 25 ms length frames with 10 ms overlap. Hamming window is applied before taking 512-points discrete Fourier Transform. The cosine function is then applied to the unwrapped phase spectrum. Frame-wise weighting is introduced at this step as explained in Subsec. 2.1. Finally, discrete cosine transform is used to extract cepstral features. First 128-dimensions (excluding the zeroth coefficient) are selected from each frame.

As the CNN model requires a uniform input, the length of the audio signals is adjusted to be 4 seconds as in (Zhang *et al.*, 2017). Note that this length is chosen based on the average length of ASVspoof 2015 data (3.5 s), but in this work it is also applied to the ASVspoof 2017 data (which has 2.6 s average training data (Chettri *et al.*, 2018)). The utterances longer than 4 s are truncated, and the utterances less than 4 s are padded by repeating the data to match the length. So, a 2-D input per utterance is provided to the inputs of the CNN. For the outputs, it is observed that using six outputs (one for genuine type, five for different spoof types represented in the training data) is more effective than two outputs (one for genuine, the other for all spoof types) for ASVspoof 2015 data (Zhang *et al.*, 2017). Therefore, six outputs are also used in this work as stated previously. In the ASVspoof 2017 data, only two classes are available, hence two outputs. The CNN models are trained separately for each database, using only the training data available in each case. MATLAB is used for all of the aforementioned process. The CNN network is trained on a single GPU (Nvidia GTX 1070 TI).

3.3. Results on ASVspoof 2015

Table 4 shows the performance of the proposed system for the development set of the ASVspoof 2015 database, and some other studies that also implemented the cosine phase-based cepstral features. Although the features are based on the cosine normalized phase spectrum, various configuration led to a high range of EER values. The proposed system achieved a moderate performance. Since EER values based on each attack type are not available for the most studies, a comparison is not possible in that sense. However, the proposed method mostly failed at capturing S2 and S5 attacks in the development data.

Table 5 shows the EER values for the evaluation data. In this case, most of the previous studies reported the results with fused systems, or simply igno-

Table 4. Comparison of the performances between the proposed Cos-CNN system and other cosine phase spectrum based studies in terms of EER [%] values for the development set of ASVspoof 2015 data. Note that most studies just reported the average EER.

System	Spoofing type					Average
	S1	S2	S3	S4	S5	
Proposed weighted Cos-CNN	0.8897	3.2001	1.4097	1.2799	5.6838	2.4926
CosPhasePC-SVM (Novoselov *et al.*, 2016)	0.13	0.20	0.04	0.05	0.23	0.15
Cos-GMM (Hanilçi, 2018b)	0.170	0.985	0.237	0.219	2.7	0.862
Cos-GMM (Liu *et al.*, 2015)	–	–	–	–	–	4.487
Cos-SVM (Liu *et al.*, 2015)	–	–	–	–	–	4.403
Cos-GMM (Hanilçi *et al.*, 2016)	–	–	–	–	–	1.09
Cos-*i*-vector (cosine scoring) (Hanilçi *et al.*, 2016)	–	–	–	–	–	11.8
Cos-*i*-vector (PLDA scoring) (Hanilçi *et al.*, 2016)	–	–	–	–	–	4.54
Cos-GMM (Sahidullah *et al.*, 2015)	–	–	–	–	–	1.11
Cos-SVM (Sahidullah *et al.*, 2015)	–	–	–	–	–	10.83

Table 5. EER [%] values for evaluation part of ASVspoof 2015.

System	Known					Unknown					
	S1	S2	S3	S4	S5	S6	S7	S8	S9	S10	Average
Proposed weighted Cos-CNN	0.63	2.64	0.89	0.73	5.06	9.83	1.59	2.78	1.01	16.72	4.193
Cos-GMM	1.20	5.03	0.12	0.15	5.49	4.53	1.77	8.13	3.79	33.90	6.41
Cos-GMM (Hanilçi, 2018b)	0.08	0.68	0.06	0.06	2.04	2.83	0.13	0.32	0.33	34.74	4.13
Fusion (Liu *et al.*, 2015)	0.17	0.61	0.31	0.28	0.39	0.90	0.24	0.41	0.24	28.58	3.21
Fusion (Xiao *et al.*, 2015)	0	0	0	0	0.01	0.01	0	0	0	26.1	2.62
Fusion (Novoselov *et al.*, 2016)	0	0.02	0	0	0.01	0.01	0	0.01	0	19.57	1.96

red the cosine normalized phase features and used several other feature types that performed better on the development data. Therefore, to emphasize the effect of weighted feature parameters, ordinary cosine phase cepstral features are extracted without the weighting step for comparison purposes. 20 cepstral coefficients (including zeroth coefficient) saved after the discrete cosine transform, and two 512 mixtures GMM are trained for genuine and spoof classes. The proposed weighted Cos-CNN system achieved ~34.5% relative improvement over the ordinary Cos-GMM system. On the other hand, a similar system of (excluding zeroth coefficient) (Hanilçi, 2018b) yielded similar average EER value as the propose weighted system, which proves that feature dimensions and types affect the performance. Further discussions are left to the next section.

The results of the evaluation data indicate that the proposed frame-wise weighted phase spectrum is much more efficient to capture S10 attacks. Even the best performing systems on average EER, which have combined several features or subsystems, are struggling to detect S10 attacks. Further optimizations on the features may help to improve the proposed system's accuracy on detection of the other attack types. As an alternative, the proposed system can be combined with the other subsystems to decrease the average EER.

3.4. Results on ASVspoof 2017

Results for both development and evaluation data are presented in Table 6. Compared to the previous database, studies that include the cosine phase features are limited in this case. Nevertheless, the experiments are conducted to observe the performance of the proposed system for replay attack detection. 57-dimensional feature vectors (including static, and its first and second order derivatives) are used in (Hanilçi, 2018a), and several different features and classifiers are investigated. For comparison, only the cosine features with GMM and *i*-vector classifiers are included in Table 6. It can be observed that the performance of the proposed system lies between the others for the development set. For the evaluation set, performance of the proposed system got closer to the *i*-vector classifier. All of the systems performed very poor on the evaluation data, which may indicate that cosine phase based features are not suitable for replay attack detection tasks.

Table 6. EER [%] values observed with the ASVspoof 2017 database.

System	Development	Evaluation
Proposed weighted Cos-CNN	17.44	39.28
Cos-GMM (Hanilçi, 2018a)	25.95	46.9
Cos-*i*-vector (Hanilçi, 2018a)	11.9	36

To have a visual examination of the network outputs, weighted cosine phase features for a selected sentence are given in Fig. 2, where the top row shows the genuine speech, and the bottom row shows the replayed speech. Left column indicates the correctly detected utterances, while right column indicates the missed utterances. Although it may be hard to see a clear correlation between the figures, the detected replayed speech (bottom left) and the missed genuine speech (top right) have a similar content at the beginning, and also around 300th frame.

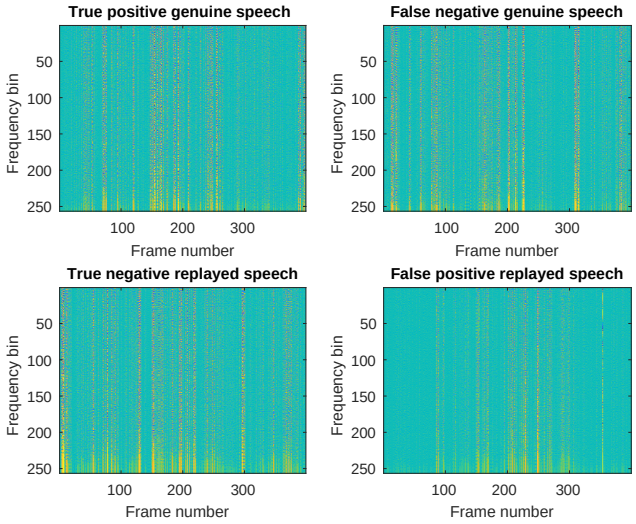

Fig. 2. Weighted cosine phase features for the correctly classified utterances (left column) and the missed utterances (right columns).

To further investigate the network outputs, Fisher ratio between the classes was calculated following Eq. (1). The ratio between the true positive genuine speech and the false negative genuine speech was calculated as 0.0115, whereas the ratio between the true positive genuine speech and the false positive replayed speech was 0.0035. Similarly, the ratio between the true negative replayed speech and the false positive replayed speech was 0.002, and the ratio between the true negative replayed speech and the false negative genuine speech was 0.0006. As the ratios between the class members are expected to be relatively small compared to the inter-class ratios, these results indicate that the proposed network's predictions were meaningful.

4. Discussion

The proposed system was tested under two different database conditions, and the results are presented in the previous section. In general, the proposed approach achieved compatible results compared to several different studies found in the literature.

For the ASVspoof 2015 database, the importance of the feature extraction parameters are observed from Tables 4 and 5. Although similar phase features and similar classifiers are used in several studies, the results vary. That indicates more improvements may be achieved with the proposed weighted phase spectrum and CNN system. For instance, first and second order derivatives can be extracted. Also, the effects of voice activity detector, cepstral mean/variance normalization, different number of features, using a specific frequency range, etc. can be investigated. Similar findings are also relevant for the results obtained via the ASVspoof 2017 database.

An interesting observation from the Table 5 is that the proposed system performed much more better than the others for the S10 portion of the evaluation set of ASVspoof 2015. The S10 attack, which is a speech synthesis algorithm, is considered as the most harmful spoof attack of that database. As can be seen in Table 5, even the systems that have achieved nearly perfect scores for the other types are highly susceptible to this specific attack. One of the most important outcomes of this paper is that the weighted cosine phase features can be used in the fused systems to further increase the system's performance, especially for the S10 attack.

A possible drawback of the proposed system could be the feature dimensions. The feature dimensions are adjusted to match the CNN inputs of Zhang *et al.* (2017), since the main focus of the proposed work is on the effects of frame-wise weighting, the CNN architecture is not modified (except the last fully connected and output layers for the ASVspoof 2017). Hence, the CNN model may be modified for optimal performance by adding/removing layers, changing the filter sizes and numbers, etc. More importantly, the 128-dimensional vectors obtained after the discrete cosine transform may include redundant information. Usually, a few coefficients are stored and the remaining are neglected, as the most useful information are presented in the first coefficients. Although high dimensional feature vectors have proven to be effective in the previous studies, high dimensional static cepstral coefficients seem to be exceptional for this case due to the aforementioned reason. As an alternative, 30 static coefficients and its derivatives may be used, instead of using the 128 static coefficients. This way, temporal information will be also served to the classifier, which may help to boost the system performance.

Other than the possible improvements for the cosine phase features, different types of phase features can also be used with the frame-wise weighting approach. As mentioned in the Sec. 2, the cosine normalized phase spectrum is chosen for its straightforward implementation, which allowed us to understand the effects of weighting. In fact, many other phase-based features such as group delay outperforms the cosine phase features. They can also benefit from the frame-wise weighting within their respective extraction process. So, a future work subject will be the comparison between several different weighted phase-based features.

5. Conclusions

In this paper, a frame-wise weighting strategy, which was reported to be effective with the magnitude spectrum, was applied to the phase spectrum. 128-dimensional cepstral coefficients were extracted from the weighted cosine phase features, and a CNN model was used as the classifier. The performance of the proposed system was analyzed through the experiments conducted with ASVspoof 2015 and 2017 databases separately, to examine the system under various spoofing attacks. Comparable results were obtained for each database. Further, the proposed system outperformed many other systems, which include fusion of features/classifiers, for the spoofing attack S10 of ASVspoof 2015 database.

Although the results indicate that the frame-wise weighting approach could be beneficial for phase spectrum features, a more detailed analysis is required for further improvements. Therefore, future works will be conducted to find a more optimal feature set, and deep learning model.

References

1. Alam M.J., Kenny P., Bhattacharya G., Stafylakis T. (2015), Development of CRIM system for the automatic speaker verification spoofing and countermeasures challenge 2015, [in:] *Interspeech 2015*, pp. 2072–2076, Dresden, Germany.
2. Alzantot M., Wang Z., Srivastava M.B. (2019), Deep residual neural networks for audio spoofing detection, [in:] *Interspeech 2019*, pp. 1078–1082, doi: 10.21437/Interspeech.2019-3174.
3. Białobrzeski R., Kośmider M., Matuszewski M., Plata M., Rakowski A. (2019), Robust Bayesian and light neural networks for voice spoofing detection, [in:] *Interspeech 2019*, pp. 1028–1032, doi: 10.21437/Interspeech.2019-2676.
4. Cai W., Wu H., Cai D., Li M. (2019), The DKU replay detection system for the ASVspoof 2019 challenge: on data augmentation, feature representation, classification, and fusion, [in:] *Interspeech 2019*, pp. 1023–1027, doi: 10.21437/Interspeech.2019-1230.
5. Chang S.-Y., Wu K.-C., Chen C.-P. (2019), Transfer-representation learning for detecting spoofing attacks with converted and synthesized speech in automatic speaker verification system, [in:] *Interspeech 2019*, pp. 1063–1067, doi: 10.21437/Interspeech.2019-2014.
6. Chen Z., Zhang W., Xie Z., Xu X., Chen D. (2018), Recurrent neural networks for automatic replay spoof-

ing attack detection, [in:] *IEEE International Conference on Acoustics, Speech and Signal Processing – Proceedings (ICASSP)*, pp. 2052–2056, doi: 10.1109/ICASSP.2018.8462644.

7. Chettri B., Benetos E., Sturm B.L.T. (2020), Dataset Artefacts in anti-spoofing systems: a case study on the ASVspoof 2017 benchmark, *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, **28**: 3018–3028, doi: 10.1109/TASLP.2020.3036777.
8. Chettri B., Sturm B.L., Benetos E. (2018), Analysing replay spoofing countermeasure performance under varied conditions, *2018 IEEE 28th International Workshop on Machine Learning for Signal Processing (MLSP)*, pp. 1–6, doi: 10.1109/MLSP.2018.8516968.
9. De Leon P.L., Pucher M., Yamagishi J., Hernaez I., Saratxaga I. (2012), Evaluation of speaker verification security and detection of HMM-Based synthetic speech, *IEEE Transactions on Audio, Speech, and Language Processing*, **20**(8): 2280–2290, doi: 10.1109/TASL.2012.2201472.
10. Dehak N., Kenny P.J., Dehak R., Dumouchel P., Ouellet P. (2011), Front-end factor analysis for speaker verification, *IEEE Transactions on Audio, Speech, and Language Processing*, **19**(4): 788–798, doi: 10.1109/TASL.2010.2064307.
11. Delgado H. *et al.* (2018), ASVspoof 2017 Version 2.0: Meta-data analysis and baseline enhancements, [in:] *The Speaker and Language Recognition Workshop*, pp. 296–303, doi: 10.21437/Odyssey.2018-42.
12. Dinkel H., Qian Y., Yu K. (2017), Small-footprint convolutional neural network for spoofing detection, [in:] *2017 International Joint Conference on Neural Networks (IJCNN)*, pp. 3086–3091, doi: 10.1109/IJCNN.2017.7966240.
13. Font R., Espín J.M., Cano M.J. (2017), Experimental analysis of features for replay attack detection-Results on the ASVspoof 2017 Challenge, [in:] *Proceedings of the Annual Conference of the International Speech Communication Association, Interspeech*, pp. 7–11, doi: 10.21437/Interspeech.2017-450.
14. Gomez-Alanis A., Peinado A.M., Gonzalez J.A., Gomez A.M. (2019), A light convolutional GRU-RNN deep feature extractor for ASV spoofing detection, *Interspeech 2019*, pp. 1068–1072, doi: 10.21437/Interspeech.2019-2212.
15. González Hautamäki R., Kinnunen T., Hautamäki V., Laukkanen A.-M. (2015), Automatic versus human speaker verification: the case of voice mimicry, *Speech Communication*, **72**: 13–31, doi: 10.1016/j.specom.2015.05.002.
16. Hanilçi C. (2018a), Features and classifiers for replay spoofing attack detection, [in:] *2017 10th International Conference on Electrical and Electronics Engineering, ELECO 2017*, pp. 1187–1191, Bursa, Turkey.
17. Hanilçi C. (2018b), Linear prediction residual features for automatic speaker verification anti-spoofing, *Multimedia Tools and Applications*, **77**(13): 16099–16111, doi: 10.1007/s11042-017-5181-0.
18. Hanilçi C., Kinnunen T., Sahidullah M., Sizov A. (2015), Classifiers for synthetic speech detection: a comparison, [in:] *Interspeech 2015*, pp. 2057–2061, Dresden, Germany.
19. Hanilçi C., Kinnunen T., Sahidullah M., Sizov A. (2016), Spoofing detection goes noisy: An analysis of synthetic speech detection in the presence of additive noise, *Speech Communication*, **85**: 83–97, doi: 10.1016/j.specom.2016.10.002.
20. Jung J., Shim H., Heo H.-S., Yu H.-J. (2019), Replay attack detection with complementary high-resolution information using end-to-end DNN for the ASVspoof 2019 challenge, [in:] *Interspeech 2019*, pp. 1083–1087, doi: 10.21437/Interspeech.2019-1991.
21. Kinnunen T. (2017), The ASVspoof 2017 Challenge: Assessing the limits of replay spoofing attack detection, [in:] *Interspeech 2017*, pp. 1–5, Stockholm, Sweden.
22. Liu M., Wang L., Dang J., Nakagawa S., Guan H., Li X. (2019), Replay attack detection using magnitude and phase information with attention-based adaptive filters, [in:] *ICASSP 2019 – 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 6201–6205, doi: 10.1109/ICASSP.2019.8682739.
23. Liu Y., Tian Y., He L., Liu J., Johnson M.T. (2015), Simultaneous utilization of spectral magnitude and phase information to extract supervectors for speaker verification anti-spoofing, [in:] *Interspeech 2015*, pp. 2082–2086, Dresden, Germany.
24. Novoselov S., Kozlov A., Lavrentyeva G., Simonchik K., Shchemelinin V. (2016), STC anti-spoofing systems for the ASVspoof 2015 challenge, [in:] *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 5475–5479, doi: 10.1109/ICASSP.2016.7472724.
25. Patel T.B., Patil H.A. (2015), Combining evidences from mel cepstral, cochlear filter cepstral and instantaneous frequency features for detection of natural vs. spoofed speech, [in:] *Interspeech 2015*, pp. 2062–2066, Dresden, Germany.
26. Rafi B.S.M., Murty K.S.R. (2019), Importance of analytic phase of the speech signal for detecting replay attacks in automatic speaker verification systems, [in:] *ICASSP 2019 – 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 6306–6310, doi: 10.1109/ICASSP.2019.8683500.
27. Reynolds D.A., Quatieri T.F., Dunn R.B. (2000), Speaker verification using adapted Gaussian mixture models, *Digital Signal Processing*, **10**(1–3): 19–41, doi: 10.1006/dspr.1999.0361.
28. Sahidullah M., Kinnunen T., Hanilçi C. (2015), A comparison of features for synthetic speech detection, [in:] *Interspeech 2015*, pp. 2087–2091, Dresden, Germany.
29. Singh M., Pati D. (2019), Combining evidences from Hilbert envelope and residual phase for detecting replay attacks, *International Journal of Speech Technology*, **22**(2): 313–326, doi: 10.1007/s10772-019-09604-x.

30. Srinivas K., Das R.K., Patil H.A. (2018), Combining phase-based features for replay spoof detection system, [in:] *2018 11th International Symposium on Chinese Spoken Language Processing (ISCSLP)*, pp. 151–155, doi: 10.1109/ISCSLP.2018.8706672.

31. Tian X., Wu Z., Xiao X., Chng E.S., Li H. (2016), Spoofing detection from a feature representation perspective, [in:] *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 2119–2123, doi: 10.1109/ICASSP.2016.7472051.

32. Todisco M., Delgado H., Evans N. (2017), Constant Q cepstral coefficients: a spoofing countermeasure for automatic speaker verification, *Computer Speech & Language*, **45**: 516–535, doi: 10.1016/j.csl.2017.01.001.

33. Todisco M. *et al.* (2019), ASVspoof 2019: future horizons in spoofed and fake audio detection, [in:] *Interspeech 2019*, pp. 1008–1012, doi: 10.21437/Interspeech.2019-2249.

34. Tom F., Jain M., Dey P. (2018), End-to-end audio replay attack detection using deep convolutional networks with attention, [in:] *Proceedings of the Annual Conference of the International Speech Communication Association, Interspeech*, pp. 681–685, doi: 10.21437/Interspeech.2018-2279.

35. Wu Z., Chng E.S., Li H. (2012), Detecting converted speech and natural speech for anti-spoofing attack in speaker recognition, [in:] *13th Annual Conference of the International Speech Communication Association 2012, Interspeech 2012*, pp. 1698–1701, Portland, OR, USA.

36. Wu Z. *et al.* (2017), ASVspoof: The automatic speaker verification spoofing and countermeasures challenge, *IEEE Journal of Selected Topics in Signal Processing*, **11**(4): 588–604, doi: 10.1109/JSTSP.2017.2671435.

37. Xiao X., Tian X., Du S., Xu H., Chng E.S., Li H. (2015), Spoofing speech detection using high dimensional magnitude and phase features: the NTU approach for ASVspoof 2015 challenge, [in:] *Interspeech 2015*, pp. 2052–2056, Dresden, Germany.

38. Yang J., Das R.K. (2020), Long-term high frequency features for synthetic speech detection, *Digital Signal Processing*, **97**(1): 1–11, doi: 10.1016/j.dsp.2019.102622.

39. Yang J., Das R.K., Li H. (2018), Extended constant-Q cepstral coefficients for detection of spoofing attacks, [in:] *2018 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, pp. 1024–1029, doi: 10.23919/APSIPA.2018.8659537.

40. Yang J., Liu L. (2018), Playback speech detection based on magnitude–phase spectrum, *Electronics Letters*, **54**(14): 901–903, doi: 10.1049/el.2018.0739.

41. Yang J., Liu L., He Q. (2019), Discriminative feature based on FWMW for playback speech detection, *Electronics Letters*, **55**(15): 861–864, doi: 10.1049/el.2019.1025.

42. Yang J., Xu L., Ren B., Ji Y. (2020), Discriminative features based on modified log magnitude spectrum for playback speech detection, *EURASIP Journal on Audio, Speech, and Music Processing*, doi: 10.1186/s13636-020-00173-5.

43. Zeinali H. *et al.* (2019), Detecting spoofing attacks using VGG and SincNet: BUT-Omilia submission to ASVspoof 2019 challenge, [in:] *Interspeech 2019*, pp. 1073–1077, doi: 10.21437/Interspeech.2019-2892.

44. Zhang C., Yu C., Hansen J.H.L. (2017), An investigation of deep-learning frameworks for speaker verification antispoofing, *IEEE Journal of Selected Topics in Signal Processing*, **11**(4): 684–694, doi: 10.1109/JSTSP.2016.2647199.