Constant Q Cepstral Coefficients: A Spoofing Countermeasure for Automatic Speaker Verification

Massimiliano Todisco, Héctor Delgado and Nicholas Evans
EURECOM, Sophia Antipolis, France

Abstract

Recent evaluations such as ASVspoof 2015 and the similarly-named AVspoof have stimulated a great deal of progress to develop spoofing countermeasures for automatic speaker verification. This paper reports an approach which combines speech signal analysis using the constant Q transform with traditional cepstral processing. The resulting constant Q cepstral coefficients (CQCCs) were introduced recently and have proven to be an effective spoofing countermeasure. An extension of previous work, the paper reports an assessment of CQCCs generalisation across three different databases and shows that they deliver state-of-the-art performance in each case. The benefit of CQCC features stems from a variable spectro-temporal resolution which, while being fundamentally different to that used by most automatic speaker verification system front-ends, also captures reliably the tell-tale signs of manipulation artefacts which are indicative of spoofing attacks. The second contribution relates to a cross-database evaluation. Results show that CQCC configuration is sensitive to the general form of spoofing attack and use case scenario. This finding suggests that the past single-system pursuit of generalised spoofing detection may need rethinking.

Keywords: spoofing, countermeasures, presentation attack detection, automatic speaker verification, constant Q transform, cepstral analysis

1. Introduction

Automatic speaker verification (ASV) technology has matured over recent years to become a low-cost and reliable approach to person recognition. Unfortunately, however, and as is true for all biometric modalities,

concerns regarding security and privacy vulnerabilities (Ratha et al., 2001; Alice, 2003; Campisi, 2013) can still form a barrier to exploitation. Vulnerabilities to spoofing, also known as presentation attacks, are one example whereby biometric systems can be manipulated by a fraudster impersonating another enrolled person. For medium to high security applications, such vulnerabilities to spoofing are clearly unacceptable.

A growing body of work has gauged the vulnerability of ASV systems to a diverse range of spoofing attacks (Evans et al., 2013; Wu et al., 2015). The major forms of attack known today include those of replay (Lindberg and Blomberg, 1999; Villalba and Lleida, 2011), voice conversion (Pellom and Hansen, 1999; Perrot et al., 2005), speech synthesis (Masuko et al., 1999; De Leon et al., 2012) and impersonation (Lau et al., 2004, 2005) all of which have been shown to degrade verification performance. The community has responded by designing countermeasure technologies to effectively mitigate vulnerabilities to spoofing.

The general countermeasure approach is essentially one of artefact detection, encompassing relatively standard feature extraction and statistical pattern recognition techniques. These aim to distinguish between natural and spoofed speech by capturing the tell-tale signs of manipulation. This might suggest that the design of spoofing countermeasures should better focus on the search for salient features rather than on the investigation of more advanced or complex classifiers.

This hypothesis is supported by the general findings of the recent ASVspoof 2015 challenge (Wu et al., 2015) and of the BTAS 2016 Speaker Anti-spoofing Competition (Korshunov et al., 2016a). The winning systems of both utilised non-conventional features in conjunction with a classical Gaussian mixture model (GMM) classifier. The winning submission to ASVspoof (Patel and Patil, 2015) used cochlear filter cepstral coefficients. Albeit in combination with standard Mel frequency cepstral coefficients (MFCCs), the winning submission to the BTAS 2016 competition used inverted MFCC features (Chakroborty et al., 2007) which were first investigated in the context of spoofing in (Sahidullah et al., 2015b). The latter and (Hanilçi et al., 2015), produced by the same team, in addition to that in (Alegre et al., 2013) adds further weight to the hypothesis that the performance of spoofing countermeasures is currently more dependent on the particular features rather than on the particular classifier.

As is argued in the following, this is perhaps not surprising. A spoofing attack must first of all manipulate successfully an ASV system into accepting

a fraudulent identity claim. It is a reasonable assumption that this will be achieved most efficiently by presenting to the system a speech signal whose corresponding features mimic as closely as possible those used for enrolment, i.e. to train the target speaker model. In most cases these are short-term, possibly Mel-scaled spectral estimates. A spoofing algorithm such as speech synthesis or voice conversion might then best be implemented using a similar feature representation at its heart. In this case, a spoofing countermeasure which uses the same or similar feature representation may not offer the best opportunities for detection.

Herein lies the research hypothesis investigated in this paper. It is supposed that the design of a spoofing countermeasure system which exploits a feature representation different to that of typical ASV systems may offer greater robustness to spoofing, in addition to greater generalisation to unforeseen spoofing attack. The most significant contribution of this paper is thus the investigation of an entirely new approach to feature extraction for ASV spoofing countermeasures with a broader focus on speech synthesis, voice conversion and replay spoofing attacks.

The new countermeasure is based upon the constant Q transform (CQT), initially proposed in the field of music processing (Brown, 1991). The CQT employs geometrically spaced frequency bins. In contrast to Fourier-based approaches which impose regular spaced frequency bins and hence a variable Q factor, the CQT ensures a constant Q factor across the entire spectrum. Furthermore, while Fourier approaches lack frequency resolution at lower frequencies and lack temporal resolution at higher frequencies, the CQT has higher frequency resolution at lower frequencies while providing a higher temporal resolution at higher frequencies. This paper investigates the use of the CQT transform for spoofing detection when coupled with traditional cepstral analysis. The latter facilitates the use of a conventional GMM for spoofing detection.

The new features are referred to as constant Q cepstral coefficients (CQCCs). Their utility for spoofing detection was first demonstrated using the ASVspoof 2015 database (Wu et al., 2014, 2015) for which they were shown to outperform the previous best result by 72% relative (Todisco et al., 2016). Since then, CQCCs have been shown to deliver competitive performance in utterance verification (Kinnunen et al., 2016; Delgado et al., 2016) and speaker verification (Sahidullah et al., 2016) tasks. This paper, an extension of the work in (Todisco et al., 2016), presents a much broader assessment based on three standard databases. They are the same ASVSpoof 2015 database

and two additional databases, AVspoof (Ergunay et al., 2015) and RedDots Replayed (Kinnunen et al., 2017). Also new to this paper is a cross-database assessment in a similar vein to the work in (Korshunov and Marcel, 2016) whereby a CQCC front-end optimised for one database is assessed using another. These results are revealing and point towards a new approach to deliver generalised countermeasures.

The remainder of the paper is organised as follows. Section 2 describes the three databases used for this work and reports derived, prior work. Section 3 presents the constant Q transform whereas the new CQCC features are described in Section 4. Section 5 describes the experimental setup whereas Section 6 presents experimental results. Conclusions are presented in Section 7.

2. Databases and prior work

This section reviews past work to develop spoofing countermeasures for automatic speaker verification (ASV). The focus is on three standard databases and derived work. The first two databases, namely ASVspoof 2015 (Wu et al., 2014, 2015) and AVspoof (Ergunay et al., 2015), are publicly available and have already been used for competitive evaluations. The third, namely RedDots Replayed (Kinnunen et al., 2017), is the most recent and will be made publicly available in 2017.

The major difference between the three databases relates to the variation in spoofing attacks. ASVspoof 2015 focuses on so-called logical access attacks, i.e. attacks injected into an ASV system post-sensor. Logical access attacks involve ASV systems in which the microphone is not controlled, i.e. outside the control of the system designers. Most telephony applications including mobile device and VoIP scenarios are examples of logical access control. The most potentially damaging spoofing attacks in this case are voice conversion and speech synthesis (Wu et al., 2015). Of course this does not exclude replay attacks which may also be used to spoof logical access control systems, including telephony applications.

The AVspoof database contains a mix of both logical access and physical access spoofing attacks, namely speech synthesis, voice conversion and replay attacks. With most physical access applications, say those involving access control to secure or sensitive infrastructure, the microphone is a fundamental part of the ASV system and under the control of the system designer. Attacks

against physical access systems are then applied at the sensor or microphone level; typically, they cannot be injected post-sensor.

The RedDots Replayed database contains a diverse mix of different replay attacks in a logical access scenario, i.e. captured and replayed speech which is injected into the ASV system post sensor. The three databases cover the full range of different spoofing attacks and two major use case scenarios. Further discussion on this topic and the impact of such differences on the study of spoofing and countermeasures is presented in (Alegre et al., 2014) and is beyond the scope of the current work.

Each database has different strengths: ASVspoof 2015 contains the greatest diversity of state-of-the-art speech synthesis and voice conversion algorithms; AVspoof offers the greatest coverage of different use case scenarios; RedDots Replayed contains the greatest variation of replay spoofing attacks. Ideally, a spoofing countermeasure should distinguish genuine speech from spoofed speech, no matter what the use case scenario and no matter what the nature of the spoofing attack. Consequently, this paper reports an assessment of spoofing countermeasure performance using all three databases identified above. The use of all three also allows a study of cross-database optimisation. The following describes each database and top-performing spoofing countermeasure systems.

2.1. ASVspoof 2015

The ASVspoof inititaive emerged from an Interspeech 2013 special session entitled 'Spoofing and Countermeasures for Automatic Speaker Verification' (Evans et al., 2013b), the findings of which showed a need for standard databases, metrics and protocols (Evans et al., 2013a). The ASVspoof 2015 database was subsequently collected and made publicly available in order to stimulate research progress (Wu et al., 2014, 2015).

Prior to 2015, the past work was characterised by spoofing attacks implemented with full knowledge of speaker verification systems and countermeasures implemented with full knowledge of spoofing attacks. This is clearly unrealistic in a practical sense. The use of a standard database avoided this problem and also allowed results produced by different researchers to be compared meaningfully. ASVspoof 2015 focused on the assessment of standalone spoofing detection in independence from ASV and also on the issue of generalisation. The latter is an important issue in spoofing detection, especially in the case of ASV which is vulnerable to different forms of spoofing attacks in addition to variations in attack algorithms. Generalisation is then

highly desirable since the nature of a spoofing attack will never be known in advance. Countermeasures should then be robust to unforeseen attacks.

2.1.1. Database, protocols and metrics

The ASVspoof 2015 database contains speech data collected from 106 speakers (45 male, 61 female) arranged in three disjoint subsets: training, development and evaluation. The training and development subsets are used for countermeasure optimisation whereas the evaluation subset is processed blindly, without further optimisation. Each subset contains a mix of genuine and spoofed speech, the latter of which is comprised of diverse spoofing attacks generated through either speech synthesis or voice conversion. A total of 10 different speech synthesis and voice conversion algorithms were used to generate spoofed data. In order to promote generalised countermeasures, only 5 of these were used to generate the training and development subsets whereas the evaluation subset was generated with the full 10. The first 5 are collectively referred to as *known* attacks, whereas the second 5, being present only in the evaluation set, are referred to as *unknown* attacks. Prior to the evaluation, only the key for the training and development subsets were available to participants; that for the evaluation subset was withheld meaning no information concerning unknown attacks was distributed to evaluation participants.

Table 1 summarizes the structure and contents of each subset, all of which contain both natural and spoofed speech for a differing number of non-overlapping speakers. Spoofed speech is derived from natural speech recordings by means of 10 different spoofing attacks (from S1 to S10). They take the form of popular speech synthesis and voice conversion algorithms described in (Wu et al., 2014). As a means of gauging generalisation, only attacks generated with algorithms S1 to S5 are included in the training and development subsets. Attacks generated with algorithms S6 to S10 are contained only within the evaluation subset. The official metric for ASVspoof 2015 is the equal error rate (EER) which is averaged cross all 10 spoofing attacks in the evaluation subset. Full details of the database, protocols and metrics are reported in (Wu et al., 2014).

2.1.2. Results

The ASVspoof 2015 evaluation results were presented at a special session of Interspeech 2015 (Wu et al., 2015). A brief description of the top 3 performing systems is presented below.

Table 1: *The ASVspoof 2015 database: training, development and evaluation partitions, number of male and female speakers, and number of genuine and spoofed speech utterances.*

	#Speakers		#Utterances	
Subset	Male	Female	Genuine	Spoofed
Training	10	15	3750	12625
Development	15	20	3497	49875
Evaluation	20	26	9404	184000

Table 2: *Equal error rate (%) results for the top 3 performing systems for the ASVspoof 2015 evaluation. The 3 first rows correspond to official evaluation results, while the last row is a post-evaluation result. Results are illustrated independently for known and unknown attacks and the average.*

System	Known	Unknown	Average
CFCC-IF (Patel and Patil, 2015)	0.408	**2.013**	**1.211**
i-vector (Novoselov et al., 2015)	**0.008**	3.922	1.965
DNN feat. (Chen et al., 2015)	0.058	4.998	2.528
Post-evaluation			
LFCC-DA (Sahidullah et al., 2015b)	0.11	1.67	0.89

- DA-IICT (Patel and Patil, 2015): a fusion of two GMM classifiers, one that uses MFCC features and another that uses cochlear filter cepstral coefficients and change in instantaneous frequency (CFCC-IF) features.
- STC (Novoselov et al., 2015): stacked i-vector features (based on MFCCs, Mel-Frequency Principal Coefficients and Cosine Phase Principal Coefficients) and a Support Vector Machine (SVM) classifier with a linear kernel.
- SJTU (Chen et al., 2015): filter bank energies with their deltas are fed into to a deep neural network to produce a new utterance representation (s-vector). Back-end scoring is performed using the Mahalanobis distance between s-vectors.

Results obtained by the three systems are illustrated in Table 2. All 3 systems achieve excellent results in the detection of known attacks, with all

EERs being below 0.5%. However, EERs for unknown attacks are significantly higher and all above 2%. The results of a fourth system are presented in the final row of Table 2. These results, the best reported to date, are post-evaluation results reported in (Sahidullah et al., 2015b). This system used the delta (D) and acceleration (A) coefficients corresponding to 20 Linear Frequency Cepstral Coefficients (LFCCs) and a classifier based on two 512-component GMMs trained with expectation maximisation (EM). While this system sacrifices performance in the case of known attacks, that for unknown attacks is well below 2%, a significant decrease in EER. Even so, the difference in performance for known and unknown attacks is significant and highlights the challenge to develop generalised countermeasures.

2.2. AVspoof

While only a single speech synthesis and voice conversion algorithm was used to generated spoofing attacks, the AVspoof database (Ergunay et al., 2015) contains spoofing attacks for three different use case scenarios: one logical access scenario and 2 physical access scenarios. The database is publicly available[1] and a version of it, supplemented with additional material, was used for a recent competition (Korshunov et al., 2016b).

2.2.1. Database, protocols and metrics

The AVspoof database contains data collected from 44 speakers (33 male and 13 female) each of whom participated in several recording sessions configured in different environmental conditions and setups. A replay attack requires playback and recording devices. In particular, in the AVspoof database recordings were collected using three different devices: a high-quality Audio Technica AT2020 USB microphone, a Samsung Galaxy S4 smartphone and an Iphone 3GS smartphone. Recordings are categorised into 3 different types: (a) read (pre-defined sentences), (b) pass (short pass-phrase) and (c) free (3 to 10 minutes of free speech).

The AVspoof database was used for the Speaker Anti-spoofing Competition held in conjunction with the 8th IEEE International Conference on Biometrics: Theory, Applications, and Systems (BTAS 2016). The competition focused only on physical access scenarios and only replay attacks. Table 3 summarizes the structure and contents of each subset, all of which contain

[1] `https://www.idiap.ch/dataset/avspoof`

Table 3: *The AVspoof 2015 database: training, development and evaluation partitions, number of male and female speakers, and number of genuine and spoofed speech utterances.*

	#Speakers		#Utterances	
Subset	Male	Female	Genuine	Spoofed
Training	10	3	4973	38580
Development	11	4	4995	38580
Evaluation	12	6	5576	44920

both natural and spoofed speech for a differing number of non-overlapping speakers. There are 10 attack scenarios including 4 replay, 3 speech synthesis and 3 voice conversion. Eight of these are referred to as *known* attacks whereas the remaining two are referred to as *unknown* attacks. The latter are not officially part of the AVspoof database and were introduced to the evaluation set for the BTAS 2016 competition. Replay attacks consist of speech which is first captured with one of the three recording devices. These recordings are then replayed using either smartphone loudspeakers, the loudspeaker of a laptop computer, or an independent, high-quality loudspeaker.

Speech synthesis attacks are all generated with the same 5-state, left-to-right hidden semi-Markov model (HSMM) speech synthesis algorithm and the adaptation of a universal or average voice model towards specific target speakers. Adaptation is performed using speech recorded with one of three different microphones. For the logical access scenario, synthetic speech is used directly (without re-recording). For the two physical access scenarios, synthetic speech is first re-played using either the loudspeaker of a laptop computer or the independent, high-quality loudspeaker, before being recaptured by the high-quality microphone.

Voice conversion attacks are all created using the same joint-density Gaussian mixture model (GMM) algorithm implemented using the Festvox toolkit[2] and a conversion function which is learned for each same-gender, source-target pair. The use case scenarios are the same as for speech synthesis, thereby producing three different voice conversion attacks.

The official metric for AVspoof is the half total error rate (HTER) (Chingovska et al., 2014). This is obtained by using the development set to deter-

[2] `http://www.festvox.org/`

Table 4: *Results for the top-3 performing systems for the AVspoof evaluation. Results are illustrated independently for the development (Dev.) and evaluation (Eval.) sets. The final evaluation performance is then computed as the half total error rate (HTER).*

System	Dev. [EER]	Eval. [HTER]
IITKGP_ABSP (Korshunov et al., 2016b)	**0.00**	**1.26**
Idiap (Korshunov et al., 2016b)	**0.00**	2.04
SJTUSpeech (Korshunov et al., 2016b)	0.42	2.20

mine the threshold θ_{dev} at the equal error rate (EER) which is then used to determine the HTER for the evaluation set.

2.2.2. Results

A brief description of the top 3 performing systems is presented below. All three are described in the same, joint competition publication (Korshunov et al., 2016b).

- IITKGP_ABSP (Korshunov et al., 2016b): based on the score-level fusion of two sub-systems using two different spectral features: (MFCCs) and inverted MFCCs (IMFCCs) (Chakroborty et al., 2008), respectively.
- Idiap (Korshunov et al., 2016b): based on long-term spectral mean and standard deviation features used with an LDA-based classifier.
- SJTUSpeech (Korshunov et al., 2016b): based on normalised, 39-dimensional PLP features and a deep neural network classifier.

Evaluation results for these three systems are illustrated in Table 4 where, according to the standard metrics, performance for the development set is expressed in terms of the EER, whereas that for the evaluation set is expressed in terms of the HTER. As is the case for the ASVspoof 2015 database, results for the development set are extremely promising, with two of the three systems achieving 0% EER. Albeit that different metrics are used for development and test sets, performance degrades for the evaluation set, with HTERs of between 1% and 2.5%. These results also illustrate the challenge to develop generalised countermeasures.

2.3. RedDots Replayed database

The RedDots Replayed database (Kinnunen et al., 2017) was developed in the context of the H2020 OCTAVE project[3] in order to support the development of countermeasures against replay spoofing attacks. While the AVspoof database captures modest variation in replay attack setup, the RedDots Replayed database was collected via crowd-sourcing using different playback and recording devices. Furthermore, while AVspoof recordings were made in a single room with variation in background noise, RedDots Replayed recordings were made in a range of very different acoustic environments.

2.3.1. Database, protocols and metrics

The RedDots Replayed database was derived from the Quarter 4 Release of the original RedDots database (Lee et al., 2015). It contains speech data of 62 speakers (49 male and 13 female speakers) from 21 countries which was collected during 572 sessions. RedDots Replayed was created using only the male-speaker subset of 'part 01' of the original database which corresponds to 10 common pass-phrases spoken by 45 speakers. Re-recordings were performed in two different conditions: controlled and variable. Controlled condition recordings were all collected in a silent office/room. In contrast, variable condition recordings were essentially uncontrolled and varied. The database is divided into disjoint training and evaluation subsets. As illustrated in Table 5, the training set contains genuine and replayed speech from 10 speakers. The evaluation set contains genuine and replayed speech from 35 speakers. All data in the training set was collected in controlled conditions whereas that in the evaluation set was collected in a mix of controlled and variable conditions. The number of utterances in each case is also illustrated in Table 5. The default metric is the EER. Full details are available in the original work (Kinnunen et al., 2017).

2.3.2. Results

The RedDots Replayed database will be released in 2017. Except for baseline results in (Kinnunen et al., 2017), no other results have yet been published in the open literature. Results for the baseline replay attack detector based on linear frequency cepstral coefficient (LFCC) features are illustrated in Table 6. The setup corresponds to the best LFCC spoofing detection

[3] `https://www.octave-project.eu`

Table 5: *The RedDots Replayed database: training and evaluation partitions, number of speakers (male only), and number of genuine and spoofed speech utterances.*

	#Speakers	#Utterances	
Subset	Male	Genuine	Spoofed
Training	10	1508	2346
Evaluation	35	9232	16067

Table 6: *Baseline countermeasure performance for the RedDots Replayed database in terms of EER for controlled, variable and pooled condition trials.*

Feature	Controlled	Variable	Pooled
LFCC (Kinnunen et al., 2017)	5.88	4.43	5.11

configuration reported in (Sahidullah et al., 2015a) for the ASVspoof 2015 database. EERs in the order of 5% are higher than for ASVspoof 2015 and AVspoof databases and would suggest that the development of countermeasures against replay attacks is a pressing concern.

3. From Fourier to constant Q

This section describes the motivation behind the use of constant Q transforms for the analysis of speech signals. The starting point for the discussion is the time-frequency representation. This is followed by a treatment of the short-term Fourier transform before a description of the constant Q transform.

3.1. Time-frequency representation

In digital audio signal processing applications, time-frequency representations are ubiquitous tools. The uncertainly principle dictates that time and frequency content cannot be measured precisely at the same time (Gabor, 1946), hence the well know relation:

$$\Delta f \Delta t \geq 1/4\pi \tag{1}$$

The parameter for this trade-off between time and frequency resolution is the window length N; Δf is proportional to $1/N$ whereas Δt is proportional to N. Equation 1 implies that, if a signal is dispersed in frequency,

then its temporal representation is compressed in time, and vice versa. Put differently, the product $\Delta f \Delta t$ is a constant; time and frequency resolutions cannot be reduced simultaneously. This means that the same time-domain signal can be specified by an infinite number of different time-frequency representations. Among these, the short-time Fourier transform (STFT) is the most popular.

3.2. The short-term Fourier transform

The STFT performs a Fourier Transform on a short segment which is extracted from a longer data record upon its multiplication with a suitable window function. A sliding window is applied repetitively in order to analyse the local frequency content of the longer data record as a function of time (Oppenheim et al., 1999).

The STFT is effectively a filter bank. The Q factor is a measure of the selectivity of each filter and is defined as the ratio between the center frequency f_k and the bandwidth δf:

$$Q = \frac{f_k}{\delta f} \tag{2}$$

In the STFT the bandwidth of each filter is constant and related to the window function. The Q factor thus increases when moving from low to high frequencies since the absolute bandwidth f is identical for all filters.

This is in contrast to the human perception system which is known to approximate a constant Q factor between 500Hz and 20kHz (Moore, 2003). At least from a perceptual viewpoint, the STFT may thus not be universally ideal for the time-frequency analysis of speech signals.

3.3. The constant Q transform

A more perceptually motivated time-frequency analysis known as the constant Q transform (CQT) was developed over the last few decades. The first was introduced in 1978 by Youngberg and Boll (Youngberg and Boll, 1978) with an alternative algorithm being proposed by Kashima and Mont-Reynaud Kashima (Mont-Reynaud, 1986). In these approaches, octaves are geometrically distributed while the centre frequencies of each filter are linearly spaced.

CQT was refined some years later in 1991 by Brown (Brown, 1991). In contrast to the earlier work, the centre frequencies of each filter are also geometrically distributed, thereby following the equal-tempered scale (Radocy

and Boyle, 1979) of western music. For this reason, Brown's algorithm is widely used in music signal processing. The approach gives a higher frequency resolution for lower frequencies and a higher temporal resolution for higher frequencies. As illustrated in Figure 1, this is in contrast to the fixed time-frequency resolution of Fourier methods. From a perceptual point of view, geometrically spaced frequencies mean that the centre frequency of every pair of adjacent filters has an identical frequency ratio and is perceived as being equally spaced. Over the last decade the CQT has been applied widely to the analysis, classification and separation of audio signals with impressive results, e.g. (Costantini et al., 2009; Jaiswal et al., 2013; Schorkhuber et al., 2013).

The CQT is similar to a wavelet transform with relatively high Q factors ($\sim$100 bins per octave.) Wavelet techniques are, however, not well suited to this computation (Mallat, 2008). For example, methods based on iterative filter banks would require the filtering of the input signal many hundreds of times (Vetterli and Herley, 1992).

3.4. CQT computation

The CQT $X^{CQ}(k,n)$ of a discrete time domain signal $x(n)$ is defined by:

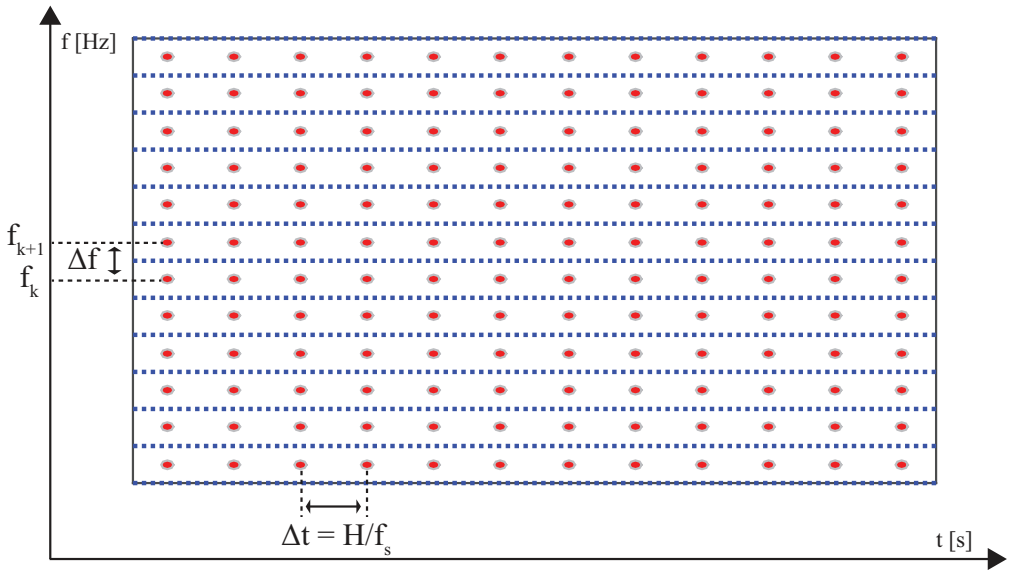
$$X^{CQ}(k,n) = \sum_{j=n-\lfloor N_k/2 \rfloor}^{n+\lfloor N_k/2 \rfloor} x(j) a_k^*(j-n+N_k/2) \tag{3}$$

where $k = 1, 2, ..., K$ is the frequency bin index, $a_k^*(n)$ is the complex conjugate of $a_k(n)$ and N_k are variable window lengths. The notation $\lfloor \cdot \rfloor$ infers rounding down towards the nearest integer. The basis functions $a_k(n)$ are complex-valued time-frequency atoms, defined according to:

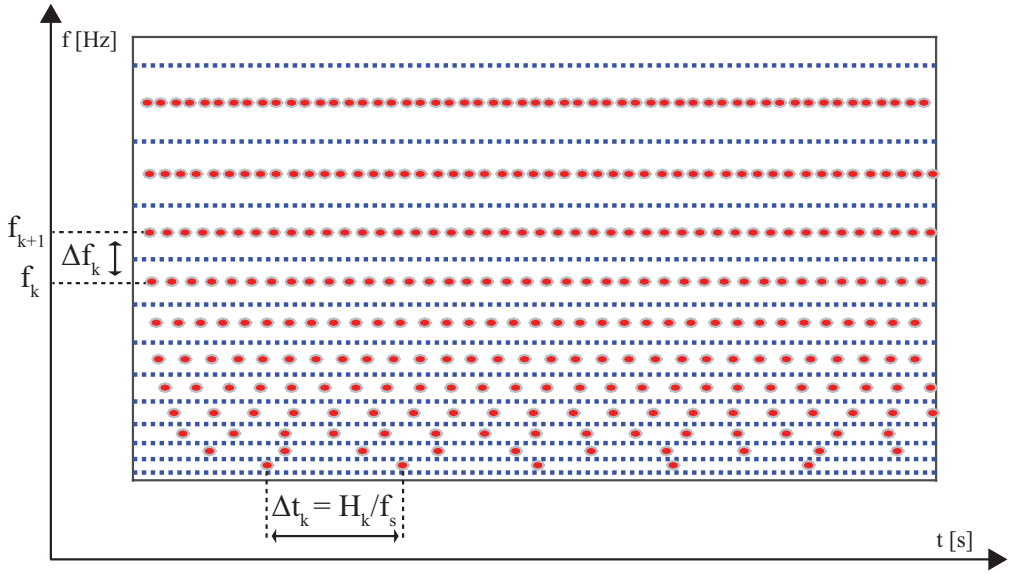
$$a_k(n) = \frac{1}{C}(\frac{n}{N_k})\exp[i(2\pi n \frac{f_k}{f_s} + \Phi_k)] \tag{4}$$

where f_k is the center frequency of the bin k, f_s is the sampling rate, and $w(t)$ is a window function (e.g. Hann window). Φ_k is a phase offset. The scaling factor C is given by:

$$C = \sum_{l=-\lfloor N_k/2 \rfloor}^{\lfloor N_k/2 \rfloor} w\left(\frac{l+N_k/2}{N_k}\right) \tag{5}$$



(a) FFT



(b) CQT

Figure 1: *A comparison of the time-frequency resolution of the STFT (a) and CQT (b). For the STFT, the time and frequency resolutions, Δt and Δf, are constant. Here, H is the duration of the sliding analysis window (hop size). In contrast, the CQT employs a variable time resolution Δt_k (which is greater for higher frequencies) and a variable frequency resolution Δf_k (which is greater for lower frequencies). Now, the duration of the sliding analysis window H_k varies for each frequency bin. f_s is the sampling rate and k is the frequency bin index. Red dots correspond to the filter bank centre frequencies f_k (bin frequencies).*

Since a bin spacing corresponding to the equal-tempered scale is desired, the center frequencies f_k obey:

$$f_k = f_1 2^{\frac{k-1}{B}} \tag{6}$$

where f_1 is the center frequency of the lowest-frequency bin and B determines the number of bins per octave. In practice, B determines the time-frequency resolution trade-off. The Q factor is then given by:

$$Q = \frac{f_k}{f_{k+1} - f_k} = (2^{1/B} - 1)^{-1} \tag{7}$$

The window lengths $N_k \in \mathbb{R}$ in Equations 3 and 4 are real-valued and inversely proportional to f_k in order that Q is constant for all frequency bins k, i.e.:

$$N_k = \frac{f_s}{f_k} Q \tag{8}$$

The work in Schrkhuber et al. (2014) introduced an additional parameter γ that gradually decreases the Q factors for low frequency bins in sympathy with the filters of the human auditory system. In particular, when $\gamma = \Gamma = 228.7 * (2^{(1/B)} - 2^{(-1/B)})$, the bandwidths equal a constant fraction of the ERB critical bandwidth (Glasberg and Moore, 1990).

Example CQT results are illustrated in Figure 2 which shows STFT and CQT-derived spectrograms for an arbitrarily selected speech signal from the ASVspoof database. The pitch $F0$ of the utterance varies between 80Hz and 90Hz; the difference is only 10Hz. The frequency resolution of the conventional STFT is not sufficient to detect such small variations; 512 temporal samples at a sampling rate of 16kHz correspond to a spectral separation of 31.25Hz between two adjacent STFT bins. This same is observed for the second partial which varies between 160Hz and 180Hz where the difference is 20Hz. The spectral resolution of the STFT can of course be improved using a larger window, but to the detriment of time resolution. The CQT efficiently resolves these different spectral contents at low frequency.

4. CQCC extraction

This section describes the extraction of constant Q cepstral coefficients. Cepstral analysis on CQT was already proposed by Brown (Brown, 1999)

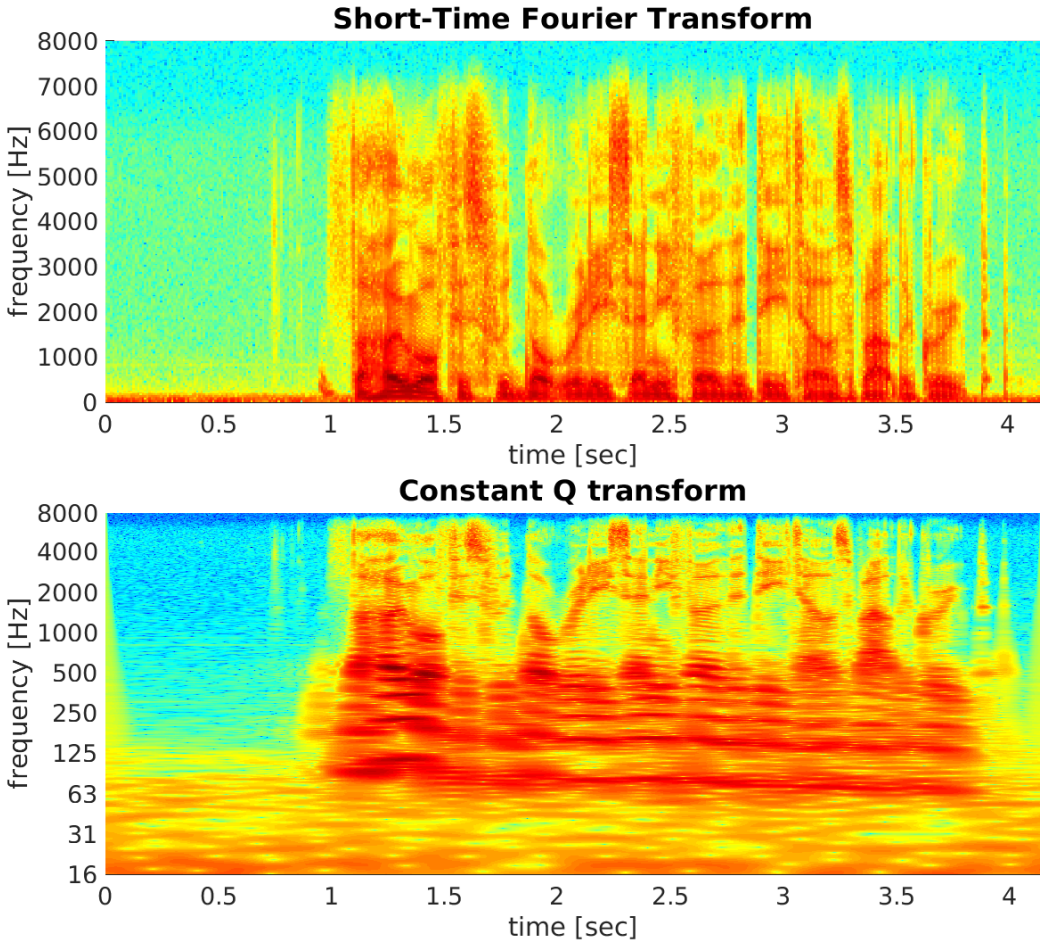


Figure 2: *Spectrograms of the utterance 'the woman is a star who has grown to love the limelight' for a male speaker in the ASVspoof database. Spectrograms computed with the short-time Fourier Transform (top) and with the constant Q transform (bottom).*

for the identification of musical instruments with a discrete success. Differently from Brown's approach, our algorithm performs a linearisation of the frequency scale of the CQT, so that the orthogonality of the DCT basis is preserved. The discussion starts with a treatment of conventional cepstral analysis before the application to CQT.

4.1. Conventional cepstral analysis

The cepstrum of a time sequence $x(n)$ is obtained from the inverse transformation of the logarithm of the spectrum. In the case of speech signals, the spectrum is usually obtained using the discrete Fourier transform (DFT) whereas the inverse transformation is normally implemented with the discrete cosine transform (DCT). The cepstrum is an orthogonal decomposition of the spectrum. It maps N Fourier coefficients onto $q \ll N$ independent cepstrum coefficients that capture the most significant information contained within the spectrum.

The Mel-cepstrum applies prior to cepstral analysis a frequency scale based on auditory critical bands (Davis and Mermelstein, 1980). It is the most common parametrisation used in speech and speaker recognition. Such features are referred to widely as Mel-frequency cepstral coefficients (MFCCs) which are typically extracted according to:

$$MFCC(q) = \sum_{m=1}^{M} \log\left[MF(m)\right] \cos\left[\frac{q\left(m-\frac{1}{2}\right)\pi}{M}\right] \tag{9}$$

where the Mel-frequency spectrum is defined as

$$MF(m) = \sum_{k=1}^{K} \left|X^{DFT}(k)\right|^2 H_m\left(k\right) \tag{10}$$

where k is the DFT index, $H_m(k)$ is the triangular weighting-shaped function for the m-th Mel-scaled bandpass filter. $MFCC(q)$ is applied to extract a number of coefficients less than the number of Mel-filters M. Typically, $M = 25$ and q varies between 13 and 20.

4.2. Constant Q cepstral coefficients

Cepstral analysis cannot be applied using (6) directly since the k bins in $X^{CQ}(k)$ are on a different scale to those of the cosine function of the DCT; they are respectively geometrically and linearly spaced. Inspired by the signal

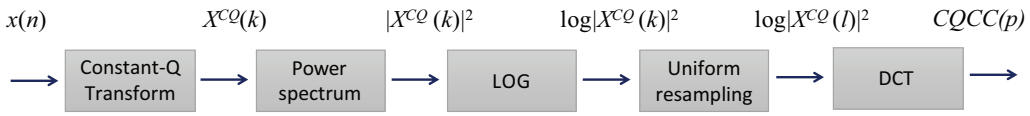


Figure 3: *Block diagram of CQCC feature extraction.*

reconstruction works in (Wolberg, 1988; Maymon and Oppenheim, 2011), this problem is solved here by converting geometric space to linear space. Since the k bins are geometrically spaced, the signal reconstruction can be viewed as a downsampling operation over the first k bins (low frequency) and as an upsampling operation for the remaining $K-k$ bins (high frequency). We define the distance between f_k and $f_1 = f_{min}$ as:

$$\Delta f^{k \leftrightarrow 1} = f_k - f_1 = f_1 \left(2^{\frac{k-1}{B}} - 1\right) \tag{11}$$

where $k = 1, 2, ..., K$ is the frequency bin index. The distance $\Delta f^{k \leftrightarrow 1}$ increases as a function of k. We now seek a period T_l for linear resampling[4]. This is equivalent to determining a value of $k_l \in 1, 2, ..., K$ such that:

$$T_l = \Delta f^{k_l \leftrightarrow 1} \tag{12}$$

To solve 12 we only need to focus on the first octave; once T_l is fixed for this octave, higher octaves will naturally have a resolution two times greater than that of the lower octave. A linear resolution is obtained by splitting the first octave into d equal parts with period T_l and by solving for k_l:

$$\frac{f_1}{d} = f_1 \left(2^{\frac{k_l-1}{B}} - 1\right) \rightarrow k_l = B \log_2(1 + \frac{1}{d}) \tag{13}$$

The new frequency rate is then given by:

$$F_l = \frac{1}{T_l} = \left[f_1 \left(2^{\frac{k_l-1}{B}} - 1\right)\right]^{-1} \tag{14}$$

There are thus d uniform samples in the first octave, $2d$ in the second and $2^j d$ in the $(j-1)^{th}$ octave. The algorithm for signal reconstruction uses a

[4]Whereas the period usually relates to the temporal domain, here it is in the frequency domain.

polyphase antialiasing filter (Jacob, 2014) and a spline interpolation method to resample the signal at the uniform sample rate F_l.

Constant Q cepstral coefficients (CQCCs) can then be extracted in a more-or-less conventional manner according to:

$$CQCC(p) = \sum_{l=1}^{L} \log \left| X^{CQ}(l) \right|^2 \cos \left[\frac{p\left(l - \frac{1}{2}\right)\pi}{L} \right] \tag{15}$$

where $p = 0, 1, ..., L-1$ and where l are the newly resampled frequency bins. The extraction of CQCCs is summarised in Figure 3.

Finally, an open-source Matlab implementation of CQCC extraction is publicly available[5]. Used in combination with the databases and protocols described in Section 2, it can be used to reproduce all results reported later in this paper.

5. Experimental setup

Presented in the following is an overview of the experimental setup including details of the feature extraction and classifier configurations.

5.1. Feature extraction

The CQT is applied with a maximum frequency of $F_{max} = F_{NYQ}$, where F_{NYQ} is the Nyquist frequency of 8kHz. The minimum frequency is set to $F_{min} = F_{max}/2^9 \simeq 15$Hz (9 being the number of octaves). The number of bins per octave B is set to 96. These parameters result in a time shift or hop of 8ms. Parameter γ is set to $\gamma = \Gamma$ (see Section 4). Re-sampling is applied with a sampling period of $d = 16$. All paramters were empirically optimised on the development data and set to minimise the spoofing detection equal error rate.

Investigations using two different CQCC features dimensions are reported: 19 and 29 all with appended C_0. These dimensions are chosen since they are common in speech and speaker recognition, respectively. The higher number is included to determine whether higher order coefficients contain any additional information useful for the detection of spoofing.

From the static coefficients, dynamic coefficients, namely delta and delta-delta features are calculated and optionally appended to static coefficients, or

[5] `http://audio.eurecom.fr/content/software`

used in isolation. Experiments were performed with all possible combinations of static and dynamic coefficients.

5.2. Classifier

Given the focus on features, all experiments reported in this paper use Gaussian mixture models (GMMs) in a standard 2-class classifier in which the classes correspond to natural and spoofed speech. The two GMMs are trained on the genuine and spoofed speech utterances of the training dataset, respectively. We use 512-component models, trained with an expectation-maximisation (EM) algorithm with random initialisation. EM is performed until likelihoods converge.

The score for a given test utterance is computed as the log-likelihood ratio $\Lambda(X) = \log L(X|\theta_n) - \log L(X|\theta_s)$, where X is a sequence of test utterance feature vectors, L denotes the likelihood function, and θ_n and θ_s represent the GMMs for natural and spoofed speech, respectively. The use of GMM-based classifiers has been shown to yield among the best performance in the detection of natural and spoofed speech (Patel and Patil, 2015; Sahidullah et al., 2015b; Kinnunen et al., 2017).

6. Experimental results

Presented in the following is an assessment of CQCC features for spoofing detection. It expands on previously reported work (Todisco et al., 2016) through new results for the AVspoof and RedDots Replayed databases. The new experiments have three objectives. The first is to assess the performance of CQCC features in different use case scenarios (physical access control and logical access control). Second, performance is assessed against greater variation in spoofing attack types and algorithms. Third, generalisation is assessed through cross-database experiments in a similar vein to the work in (Korshunov and Marcel, 2016) (front-end optimisation on one database and evaluation based on another).

Results are first presented in turn for each of the three databases alone. In each case, the first set of results refers to the development subsets for which the CQCC front-end is independently optimised. The second set of results refers to the corresponding evaluation subsets (ASVspoof 2015 and AVSpoof only since there are no independent development and evaluation subsets for the RedDots Replayed database). A comparison of CQCC performance to

Table 7: *Spoofing detection performance for the ASVspoof 2015 development subset using CQCC features. Performance measured in terms of average EER (%) and illustrated for different feature dimensions and combinations of static and dynamic coefficients. S=static, D=dynamic, A=acceleration.*

Feature	$19 + C_0$	$29 + C_0$
S	0.3850	0.3619
D	0.0942	0.0412
A	0.0518	**0.0100**
SDA	0.0947	0.0735
SD	0.2331	0.1622
SA	0.1564	0.0948
DA	**0.0381**	0.0154

competing approaches in the literature are then presented in each case and aim to assess the potential of the CQCC front-end in terms of generalisation.

The third set of experiments involving cross-database experiments are reported last. While extensive experimentation was performed separately for each database with a multitude of different front-end configurations, the presentation below focuses on the most revealing, common CQCC configurations. They include either 19 or 29 CQCC coefficients appended by energy (C_0 or 0th cepstral coefficient) and 7 different combinations of static (S), delta (D) and acceleration (A) parameters.

6.1. ASVspoof 2015

The first set of results presented here relate to the ASVspoof 2015 database. Protocols are exactly the same as those described in Section 2.1. Results reported here are the same as those published previously in (Todisco et al., 2016).

6.1.1. Development and evaluation results

Results for the ASVspoof 2015 development subset are illustrated in Table 7. First, no matter what the combination of S, D or A parameters, better performance is achieved with the higher dimension features, indicating the presence of useful information in the higher order cepstra. Second, dynamic and acceleration coefficients give considerably better results than static coefficients. Acceleration coefficients give better results than dynamic

Table 8: *Spoofing detection performance for the ASVspoof 2015 evaluation subset using CQCC features. System performance for known and unknown attacks measured in terms of average EER (%) for the four best system configurations found for the development set.*

#coef.	$19 + C_0$		$29 + C_0$	
Feat.	Known	Unknown	Known	Unknown
A	0.0484	**0.4625**	0.0185	**0.6724**
DA	**0.0228**	0.8263	**0.0098**	0.8384

coefficients though, for the lower dimension features, their combination gives better performance than either alone. The fact that dynamic and acceleration coefficients outperform static features seems reasonable given that spoofing algorithms such as voice conversion and speech synthesis tend not to model well the more dynamic information in natural speech.

Results for the ASVspoof 2015 evaluation subset are illustrated in Table 8 for both 19 and 29 dimension features with appended C_0 and for the best A and DA combinations. Results are illustrated separately for known and unknown attacks. While results for DA combinations are superior in the case known spoofing attacks, the use of A features alone provides better performance in the case of unknown spoofing attacks. Since performance improves with more dynamic information, experiments where also run with the derivatives of acceleration coefficients. While small improvements were observed, they were not consistently beneficial and thus these are not reported here.

These results show that performance degrades significantly in the face of unknown attacks. This interpretation would be rather negative, however. Presented in the following is a comparison of CQCC to other results in the literature. These show that, even if performance for unknown spoofing attacks is worse than for known attacks, CQCC features still deliver excellent performance. Even so, the difference between performance for known and unknown attacks remains and shows that the quest for generalised countermeasures is far from being a solved.

Table 9: *Spoofing detection performance for the ASVspoof 2015 evaluation subset using CQCC features. Performance in terms of EER (%) illustrated independently for each of the 10 ASVspoof attacks and for (i) systems reviewed in Section 2.1.2 and (ii) CQCC_A features (19 CQCCs + C_0, A coefficients only). Results for known and unknown attacks and the global average.*

	Known Attacks						Unknown Attacks						All
System	S1	S2	S3	S4	S5	Avg.	S6	S7	S8	S9	S10	Avg.	Avg.
CFCC-IF	0.101	0.863	0.000	0.000	1.075	0.408	0.846	0.242	0.142	0.346	8.490	2.013	1.211
i-vector	0.004	0.022	0.000	0.000	0.013	0.008	0.019	0.000	0.015	0.004	19.57	3.922	1.965
DNN feat.	0.032	0.109	0.032	0.032	0.086	0.058	0.173	0.049	0.121	0.049	24.601	4.998	2.528
LFCC-DA	0.027	0.408	0.000	0.000	0.114	0.110	0.149	0.011	0.074	0.027	8.185	1.670	0.890
CQCC-A	0.005	0.106	0.000	0.000	0.130	0.048	0.098	0.064	1.033	0.053	1.065	0.462	0.255

6.1.2. Comparative assessment and generalisation

Table 9 compares the performance of CQCC features to that of the 4 best performing previous approaches[6] reported in Section 2.1.2. Performance is illustrated individually for each of the 10 different spoofing attacks in addition to the average for known, unknown and pooled trials. CQCC results relate to 19th order features with C_0 and A coefficients only.

Focusing first on known attacks, all four systems deliver excellent error rates of below 0.41%. CQCC features are third in the ranking according to an average EER of 0.05%. Voice conversion attacks S2 and S5 are the most difficult to detect. Speech synthesis attacks S3 and S4, however, are perfectly detected by all systems.

It is for unknown attacks where the difference between system performance is greatest. Whereas attacks S6, S7 and S9 are detected reliably by all systems, there is considerable variation for attacks S8 and S10. S8 is the only tensor-based voice conversion algorithm. Performance for attack S10, the only unit-selection-based speech synthesis algorithm, varies considerably; past results range from 8.2% to 26.1%. However, results for CQCC features still compare favourably. While the performance for S6, S7 and S9 is worse than that of other systems, error rates are still low and below 0.1%. While the error rate for S8 of 1.0% is considerably higher than for other systems, the EER for S10 of 1.1% is significantly better. The latter corresponds to a relative improvement of 87% with regard to the next best performing system. The average performance of CQCC features for unknown attacks is 0.5%. This corresponds to a relative improvement of 72% over the next best system. Difference in performance stem from differences in the time-frequency resolution between the STFT and CQT. For the STFT, the time and frequency resolution are constant. In contrast, the CQT has variable time and frequency resolutions: time resolution is greater for higher frequencies whereas frequency resolution is greater for lower frequencies. The resolution of the CQT captures information more salient to the task of spoofing detection, hence better performance.

The average performance across all 10 spoofing attacks is illustrated in the final column of Table 9. The average EER of 0.26% is significantly better than figures reported in previous work. The picture of generalisation is thus

[6]The authors thanks Md Sahidullah and Tomi Kinnunen from the University of Eastern Finland for kindly providing independent results for each spoofing attack.

Table 10: *Spoofing detection performance for the AVspoof development and evaluations sets using CQCC features. Performance measured in terms of average EER (%) for the Development set and in terms of HTER (%) for the Evaluation set and illustrated for different feature dimensions and combinations of static and dynamic coefficients. S=static, D=dynamic, A=acceleration.*

Features	Development set - EER		Evaluation set - HTER	
	19+0th	29+0th	19+0th	29+0th
SDA	**0.00**	**0.00**	**0.67**	0.82
SD	**0.00**	**0.00**	1.14	0.88
SA	**0.00**	**0.00**	0.79	**0.72**
DA	2.24	1.84	5.44	4.70
A	2.52	2.14	5.65	4.58
D	2.40	2.14	4.61	4.69
S	**0.00**	**0.00**	1.08	0.91

not straightforward. While performance for unknown attacks is worse than it is for known attacks, CQCC features nonetheless deliver the most consistent performance across the 10 different spoofing attacks in the ASVspoof 2015 database. Even if it must be acknowledged that this work was conducted post-evaluation, to the authors' best knowledge, CQCC features give the best spoofing detection performance reported to date.

6.2. AVspoof

Reported here are results for the AVspoof database which is described in Section 2.2. Protocols are those used for the Speaker Anti-spoofing Competition held in conjunction with BTAS 2016 (Korshunov et al., 2016b).

6.2.1. Development and evaluation results

Results for the same feature dimensions and 7 different combinations of static (S), delta (D) and acceleration (A) illustrated in Table 10 show CQCC spoofing detection performance for the AVspoof database and for both development and evaluation subsets. In contrast to results obtained for the ASVspoof 2015 database, the use of static coefficients is crucial to reliable detection; all configurations which include static coefficients give better performance than those without. This finding, while contradicting that for

the ASVspoof 2015 database, relates to the difference in use case scenario. Whereas they have little role to play in the detection of logical access spoofing attacks, static coefficients are pertinent to the detection of physical access attacks such as those in the AVspoof database. For the development set, all configurations with static coefficients deliver perfect spoofing detection performance with an EER of 0%. For the evaluation set, results are computed in terms of HTER with the threshold computed at the EER operation point in the development set. Given that several configurations delivered 0% EER on the development set, there is no a unique threshold value to choose. In those cases, we have selected the threshold as the average of the minimum score of the target (natural speech) trials and the maximum score among the non-target (spoofed speech) trials. The best performing SDA configuration with 19 coefficients and C_0 provides an HTER of 0.67%. This result would suggest that dynamic coefficients still have an important role in spoofing detection performance.

Table 11 shows performance individually for each of the 10 different spoofing attacks in the AVspoof evaluation subset. All results relate to an operating point where the threshold is set according to the EER for the development set. Training data for attacks A1-8 are provided in the development set whereas attacks A9 and A10 are present only in the evaluation set. The latter are thus referred to as unknown attacks. The HTER for each of the known attacks is 0.29%. This is because the false acceptance rate (FAR) for all attacks is 0%, while the false rejection rate (FRR) (related only to genuine trials, the result of a common threshold and shared for all experiments) is 0.59%, hence the same HTER. This results in the same HTER value. The same result is obtained for the first unknown attack A9, however the HTER for attack A10 is considerably higher at 23.92%. This stems from the increase in FAR which is 47.25%. This is caused by the particularly high-quality nature of attacks A10 which leave very little convolutive artefacts for detection, hence the higher error rate. The pooled HTER for all attacks is 0.67%.

6.2.2. Comparative assessment and generalisation

Table 12 shows the performance of CQCC features independently for each of the different spoofing attacks grouped into known and unknown attacks. Results are for 19 CQCCs + C_0 and for the SDA combination. Focusing first on known attacks, all three systems deliver excellent pooled HTER rates in the order of 2% and below. CQCC features deliver by far the lowest HTER of 0.29%. Performance for unknown attacks varies considerably with some

Table 11: *Spoofing detection performance for the AVspoof evaluation subset using CQCC features. Performance in terms of FRR (%), FAR (%) and HTER (%) (using the threshold obtained for the development set) illustrated independently for each of the 10 AVspoof attacks and for pooled results. All results correspond to CQCC_SDA features (19 CQCCs + C_0, SDA combination). 'SS' stands for speech synthesis spoofing attacks, 'VC' for voice conversion, and 'RE' for replay. 'LP' indicates a laptop loudspeaker was used for replay, 'PH1' for a Samsung Galaxy S4 phone, 'PH2' for an iPhone 3GS, 'PH3' for an iPhone 6S, and 'HQ' for high quality speakers.*

Attack	FRR	FAR	HTER
A1 - SS-LP-LP	0.59	0.00	0.29
A2 - SS-LP-HQ-LP	0.59	0.00	0.29
A3 - VC-LP-LP	0.59	0.00	0.29
A4 - VC-LP-HQ-LP	0.59	0.00	0.29
A5 - RE-LP-LP	0.59	0.00	0.29
A6 - RE-LP-HQ-LP	0.59	0.00	0.29
A7 - RE-PH1-LP	0.59	0.00	0.29
A8 - RE-PH2-LP	0.59	0.00	0.29
A9 - RE-PH2-PH3 (unknown attack)	0.59	0.00	0.29
A10 - RE-LP-PH2-PH3 (unknown attack)	0.59	47.25	23.92
Overall (pooled)	0.59	0.65	0.67

Table 12: *Spoofing detection performance for the AVspoof evaluation subset using CQCC features. Performance in terms of average HTER (%) illustrated independently for each of the 10 AVspoof attacks and for (i) systems reviewed in Section 2.2.2 and (ii) CQCC_SDA features (19 CQCCs + C_0, SDA combination). Results for known, unknown and pooled trials.*

	Known Attacks									Unknown Attacks			All
System	A1	A2	A3	A4	A5	A6	A7	A8	Pooled	A9	A10	Pooled	Pooled
IITKGP_ABSP	0.68	0.68	0.74	0.81	8.58	1.81	0.68	3.59	0.98	6.49	23.06	14.75	1.26
Idiap	0.27	0.27	0.33	0.27	15.83	0.58	0.33	25.18	1.05	50.08	46.64	48.36	2.04
SJTUSpeech	1.88	1.75	1.73	1.81	10.34	10.02	1.52	2.05	2.08	2.84	18.09	10.46	2.20
CQCC-SDA	0.29	0.29	0.29	0.29	0.29	0.29	0.29	0.29	0.29	0.29	23.92	12.10	0.67

Table 13: *Spoofing detection performance for the RedDots Replayed database using CQCC features. Performance measured in terms of average EER (%) and illustrated for different feature dimensions and combinations of static and dynamic coefficients. S=static, D=dynamic, A=acceleration.*

Feature	19 + C_0	29 + C_0
SDA	6.48	5.93
SD	6.82	5.77
SA	6.09	5.53
DA	**2.81**	**1.85**
A	3.27	2.92
D	5.88	5.16
S	7.05	6.69

results in the order of 50% HTER. CQCC features perform well with the best result for A9 but third best result for A10. Pooled results show that CQCC delivers an HTER of 12.1%, only marginally worse than the best result of 10.5%. The HTER pooled across all known and unknown attacks is 0.67%. This corresponds to a relative improvement of 47% over the next best system.

6.3. RedDots Replayed

Reported here are results for the RedDots Replayed database which is described in Section 2.3. Protocols are those used in (Kinnunen et al., 2017). There is no development dataset for this database hence the following relates to the single evaluation set alone.

6.3.1. Evaluation results

Results for the RedDots Replayed database in Table 13 show spoofing detection performance for the same feature dimensions and 7 different combinations of static (S), delta (D) and acceleration (A) CQCC features. The first observation is that performance is generally poorer than that for both the ASVspoof 2015 and AVspoof databases. However, the trend is similar to that for the ASVspoof 2015 corpus: better performance is achieved with higher dimension features and A and DA coefficients. No matter what the dimension, the optimal configuration involves the combination of DA features.

Table 14 shows performance for the same optimal configurations but with results illustrated separately for the two acoustic conditions, namely con-

Table 14: *Spoofing detection performance for the RedDots Replayed database using CQCC features. Performance illustrated in terms of average EER (%) for controlled and variable acoustic environments and for the two feature dimensions both with a DA combination.*

	19 + C_0	29 + C_0
Controlled	2.56	**1.80**
Variable (unknown attack)	3.01	**1.92**

Table 15: *Spoofing detection performance for the RedDots Replayed database. Performance in terms of average EER (%) illustrated independently for (i) the baseline system in 2.3.2 and (ii) CQCC_DA features (29 CQCCs + C_0, DA combination). Results illustrated independently for each of the two acoustic environments and pooled trials.*

Feature	Controlled	Variable	Pooled
LFCC (Kinnunen et al., 2017)	5.88	4.43	5.11
CQCC-DA	**1.80**	**1.92**	**1.85**

trolled and variable, the latter is the unknown attack (i.e., not preset in the training set). While results in Table 13 already show that the higher dimension feature gives better performance, those in Table 14 show that the higher dimension feature also shows less variation across different acoustic environments; the performance across controlled and variable conditions is similar. This is despite the lack of variable condition data in the training set.

6.3.2. Comparative performance

Table 15 presents a comparison of CQCC features to the baseline results reported in (Kinnunen et al., 2017). CQCC features give universally better performance. The pooled EER for CQCC features of 1.85% is a relative improvement over the baseline of 64%. It should be noted, however, that the RedDots Replayed database will only be made publicly available in 2017; there is no other work in the literature against which performance comparisons can be made.

6.4. Cross-database evaluation

The aim here is to observe the degradation in performance when features optimised using one database are used on another. This analysis provides some insight into which features might give the most reliable and consistent performance in a practical situation where the variation in spoofing attacks

is likely to be greater than that reflected in any of the three databases alone. It also serves to evaluate over-fitting which might be characterised by large variations in performance for a single configuration.

Table 16 shows spoofing detection performance in terms of average EER (%) for the ASVspoof 2015 and AVspoof evaluation subsets and the RedDots Replayed database. Figures in bold face show the optimal feature configuration for each database. Focusing on differences in feature configuration, the first observation is that the optimal configuration for each dataset is different. Second, dynamic and/or acceleration coefficients are universally helpful; all three configurations contain either one or the other. Third, static coefficients are only used in one configuration.

Turning next to differences for each database, the immediate observation is that performance varies significantly. For the ASVspoof 2015 database, the difference between the best and worst performance, while low in real terms, is equivalent to a 3-fold increase in EER (0.26% to 0.76%). The relative degradation for the AVspoof database is even greater, with the difference between the best and worst performance being over an 8-fold increase in HTER (0.67% to 5.65%). For RedDots Replayed, the difference between best and worst results corresponds to a 3.5-fold increase in EER.

The question then is, which features are best? This question would require much further work to answer. Another question is indeed whether or not it is even a sensible one to ask. While an average of the results in each row of Table 16 might be revealing, it would probably be misleading too. The size of each dataset is different, meaning that results would be skewed inappropriately by results for the smallest dataset. Fundamentally, though, the search for a single feature might not even be a sensible pursuit since both use case scenarios and spoofing attacks are different. Different problems may then require different solutions. A physical access scenario may call inherently for a different front-end than a logical access scenario. Spoofing attacks such as speech synthesis and voice conversion call for a different front-ends than replay attacks where artefacts originate not from signal processing, but from what are essentially channel differences.

Accordingly, while CQCC features outperform the previous state of the art for all three datasets, further work is required to develop a spoofing countermeasure with genuine practical utility. Spoofing countermeasures are essentially only as secure as their weakest vulnerability; once a vulnerability is found, say to replay attacks, fraudsters would likely focus their efforts on that one vulnerability alone. Therefore, a countermeasure solution must

Table 16: *Spoofing detection performance in terms of average EER (%) and HTER (%) for the ASVspoof and AVspoof evaluation subsets and the RedDots Replayed database. Performance is illustrated for the three respective optimal CQCC feature configurations but across all three datasets. Figures along the diagonal illustrated in bold indicate the optimal feature configuration for each dataset.*

Feature configuration	ASVspoof 2015 EER	AVspoof HTER	RedDots Replayed EER
CQCC_A 19 + C_0	**0.26**	5.65	3.27
CQCC_SDA 19 + C_0	0.76	**0.67**	6.48
CQCC_DA 29 + C_0	0.42	4.70	**1.85**

necessarily offer resilience to *all* potential forms of spoofing attack. Generalisation remains key. However, this work shows that an effective solution may involve not a single front-end, but multiple front-ends, possibly in the form of a bank of classifiers, each tuned to the reliable detection of different spoofing attacks. Whether or not this would be feasible in practice, and whether or not such a bank of classifiers would be able to detect spoofing attacks reliably without introducing false alarms, is the subject of our ongoing work.

7. Conclusions

The coupling of conventional cepstral analysis with the variable spectro-temporal resolution of the constant Q transform was shown previously to outperform competing approaches to spoofing detection. The past work evaluated the new constant Q cepstral coefficients (CQCCs) using the ASVspoof 2015 database for which they were shown to outperform the previous state of the art by 72% relative. The ASVspoof 2015 dataset focuses on speech synthesis and voice conversion spoofing attacks in a logical access control use case scenario.

This paper extends the past work with similar evaluations using the AVspoof and RedDots Replayed databases. Together they reflect a broader range of use case scenarios, including physical access control, and also a far greater number of different spoofing attacks. Results for the AVspoof database show a relative performance improvement of 47% over the previous best results. Those for the RedDots Replayed database show a relative improvement of 64% over the previous best results. Together, these results show that CQCC features are more effective than previous approaches in

capturing the tell-tale signs of manipulation artefacts which are indicative of spoofing attacks.

The contributions in this paper extend further. Also reported is a cross-database evaluation which assesses the performance of CQCC features on one database using front-ends which are optimised on another. These results show that, while being superior to past results, performance is sensitive to the precise CQCC configuration. These results call into question the search for a single, generalised front-end which is effective in detecting different spoofing attacks in different use case scenarios. The same results might then suggest that spoofing attacks of a different nature call fundamentally for a different solution and that, consequently, future work should investigate a bank-of-classifiers solution to spoofing detection. This work will involve more than classical fusion, however, in order to manage properly the potential for negative impacts on usability, i.e. increases in false alarms.

Acknowledgements

The paper reflects some results from the OCTAVE Project (#647850), funded by the Research Executive Agency (REA) of the European Commission, in its framework programme Horizon 2020. The views expressed in this paper are those of the authors and do not engage any official position of the European Commission.

References

Alegre, F., Evans, N., Kinnunen, T., Wu, Z., Yamagishi, J., 2014. Anti-spoofing: Voice databases. In: Li, S. Z., Jain, A. K. (Eds.), Encyclopedia of Biometrics. Springer US.

Alegre, F., Vipperla, R., Amehraye, A., Evans, N., 08 2013. A new speaker verification spoofing countermeasure based on local binary patterns. In: INTERSPEECH. Lyon.

Alice, I., 2003. Biometric recognition: Security and privacy concerns. IEEE Security & Privacy.

Brown, J., January 1991. Calculation of a constant Q spectral transform. Journal of the Acoustical Society of America 89 (1), 425–434.

Brown, J., 1999. Computer identification of musical instruments using pattern recognition with cepstral coefficients as features. The Journal of the Acoustical Society of America 105 (3), 1933–1941.

Campisi, P., 2013. Security and Privacy in Biometrics. Springer.

Chakroborty, S., Roy, A., Saha, G., 2007. Improved closed set text-independent speaker identification by combining mfcc with evidence from flipped filter banks. International Journal of Signal Processing 4, 114–122.

Chakroborty, S., Roy, A., Saha, G., 2008. Improved closed set text-independent speaker identification by combining mfcc with evidence from flipped filter banks. International Journal of Electrical, Computer, Energetic, Electronic and Communication Engineering 2 (11), 100 – 107.

Chen, N., Qian, Y., Dinkel, H., Chen, B., Yu, K., 2015. Robust deep feature for spoofing detection - the SJTU system for ASVspoof 2015 challenge. In: INTERSPEECH.

Chingovska, I., Anjos, A., Marcel, S., Dec. 2014. Biometrics evaluation under spoofing attacks. IEEE Transactions on Information Forensics and Security 9 (12), 2264–2276.

Costantini, G., Perfetti, R., Todisco, M., Sep. 2009. Event based transcription system for polyphonic piano music. Signal Process. 89 (9), 1798–1811.

Davis, S., Mermelstein, P., Aug 1980. Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. IEEE Transactions on Acoustics, Speech and Signal Processing 28 (4), 357–366.

De Leon, P. L., Pucher, M., Yamagishi, J., Hernaez, I., Saratxaga, I., Oct. 2012. Evaluation of speaker verification security and detection of HMM-based synthetic speech. Audio, Speech, and Language Processing, IEEE Transactions on 20 (8), 2280–2290.

Delgado, H., Todisco, M., Sahidullah, M., Sarkar, A. K., Evans, N., Kinnunen, T., Tan, Z.-H., Dec. 2016. Further optimisations of constant Q cepstral processing for integrated utterance verification and text-dependent speaker verification. In: SLT 2016, IEEE Workshop on Spoken Language Technology. San Diego.

Ergunay, S., Khoury, E., Lazaridis, A., Marcel, S., Sept 2015. On the vulnerability of speaker verification to realistic voice spoofing. In: IEEE 7th International Conference on Biometrics Theory, Applications and Systems (BTAS). pp. 1–6.

Evans, N., Kinnunen, T., Yamagishi, J., 2013. Spoofing and countermeasures for automatic speaker verification. In: INTERSPEECH. pp. 925–929.

Evans, N., Yamagishi, J., Kinnunen, T., 05 2013a. Spoofing and countermeasures for speaker verification: a need for standard corpora, protocols and metrics. IEEE Signal Processing Society Newsletter, May 2013.

Evans, N. W. D., Kinnunen, T., Yamagishi, J., 08 2013b. Spoofing and countermeasures for automatic speaker verification. In: INTERSPEECH 2013, 14th Annual Conference of the International Speech Communication Association, August 25-29, 2013, Lyon, France. Lyon.

Gabor, D., 1946. Theory of communication. J. Inst. Elect. Eng. 93, 429–457.

Glasberg, B. R., Moore, B. C. J., 1990. Derivation of auditory filter shapes from notched-noise data. Hearing Research 47 (1), 103 – 138.

Hanilçi, C., Kinnunen, T., Sahidullah, M., Sizov, A., 2015. Classifiers for synthetic speech detection: a comparison. In: INTERSPEECH. pp. 2087–2091.

Jacob, P., 2014. Design and implementation of polyphase decimation filter. International Journal of Computer Networks and Wireless Communications (IJCNWC), ISSN, 2250–3501.

Jaiswal, R., Fitzgerald, D., Coyle, E., Rickard, S., June 2013. Towards shifted nmf for improved monaural separation. In: 24th IET Irish Signals and Systems Conference (ISSC 2013). pp. 1–7.

Kinnunen, K., Sahidullah, M., Falcone, M., Costantini, L., González Hautamäki, R., Thomsen, D., Sarkar, A. K., Tan, Z.-H., Delgado, H., Todisco, M., Evans, N., Hautamäki, V., Lee, K. A., 2017. Reddots replayed: A new replay spoofing attack corpus for text-dependent speaker verification research. In: ICASSP.

Kinnunen, T., Sahidullah, M., Kukanov, I., Delgado, H., Todisco, M., Sarkar, A., Thomsen, N., Hautamaki, V., Evans, N., Tan, Z.-H., Sept. 2016. Utterance verification for text-dependent speaker recognition: a comparative assessment using the RedDots corpus. In: INTERSPEECH 2016, Annual Conference of the International Speech Communication Association, September 8-12, 2016, San Francisco, USA. San Francisco.

Korshunov, P., Marcel, S., Sep. 2016. Cross-database evaluation of audio-based spoofing detection systems. In: Interspeech.

Korshunov, P., Marcel, S., Muckenhirn, H., 2016a. Overview of btas 2016 speaker anti-spoofing competition. In: 8th IEEE International Conference on Biometrics: Theory, Applications, and Systems (BTAS).

Korshunov, P., Marcel, S., Muckenhirn, H., Gonçalves, A. R., Mello, A. G. S., Violato, R. P. V., Simões, F. O., Neto, M. U., de Assis Angeloni, M., Stuchi, J. A., Dinkel, H., Chen, N., Qian, Y., Paul, D., Saha, G., Sahidullah, M., Sep. 2016b. Overview of btas 2016 speaker anti-spoofing competition. In: IEEE International Conference on Biometrics: Theory, Applications and Systems.

Lau, Y. W., Tran, D., Wagner, M., 2005. Knowledge-Based Intelligent Information and Engineering Systems: 9th International Conference, KES 2005, Melbourne, Australia, September 14-16, 2005, Proceedings, Part IV. Springer Berlin Heidelberg, Berlin, Heidelberg, Ch. Testing Voice Mimicry with the YOHO Speaker Verification Corpus, pp. 15–21.

Lau, Y. W., Wagner, M., Tran, D., Oct 2004. Vulnerability of speaker verification to voice mimicking. In: Proceedings of 2004 International Symposium on Intelligent Multimedia, Video and Speech Processing, 2004. pp. 145–148.

Lee, K., Larcher, A., Wang, G., Kenny, P., Brümmer, N., van Leeuwen, D. A., Aronowitz, H., Kockmann, M., Vaquero, C., Ma, B., Li, H., Stafylakis, T., Alam, M. J., Swart, A., Perez, J., 2015. The reddots data collection for speaker recognition. In: INTERSPEECH 2015, 16th Annual Conference of the International Speech Communication Association, Dresden, Germany, September 6-10, 2015. pp. 2996–3000.
URL http://www.isca-speech.org/archive/interspeech_2015/i15_2996.html

Lindberg, J., Blomberg, M., 1999. Vulnerability in speaker verification a study of technical impostor techniques. In: Proc. European Conference on Speech Communication and Technology (Eurospeech).

Mallat, S., 2008. A Wavelet Tour of Signal Processing, Third Edition: The Sparse Way, 3rd Edition. Academic Press.

Masuko, T., Hitotsumatsu, T., Tokuda, K., Kobayashi, T., 1999. On the security of HMM-based speaker verification systems against imposture using synthetic speech. In: In Proceedings of the European Conference on Speech Communication and Technology. pp. 1223–1226.

Maymon, S., Oppenheim, A. V., Oct 2011. Sinc interpolation of nonuniform samples. IEEE Transactions on Signal Processing 59 (10), 4745–4758.

Mont-Reynaud, B., 1986. The bounded-Q approach to time-varying spectral analysis.

Moore, B. C. J., 2003. An Introduction to the Psychology of Hearing. BRILL.

Novoselov, S., Kozlov, A., Lavrentyeva, G., Simonchik, K., Shchemelinin, V., 2015. STC anti-spoofing systems for the asvspoof 2015 challenge. In: INTERSPEECH.

Oppenheim, A. V., Schafer, R. W., Buck, J. R., 1999. Discrete-time Signal Processing (2Nd Ed.). Prentice-Hall, Inc., Upper Saddle River, NJ, USA.

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. pp. 2062–2066.

Pellom, B. L., Hansen, J. H. L., Mar 1999. An experimental study of speaker verification sensitivity to computer voice-altered imposters. In: Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP). Vol. 2. pp. 837–840 vol.2.

Perrot, P., Aversano, G., Blouet, R., Charbit, M., Chollet, G., March 2005. Voice forgery using ALISP: Indexation in a client memory. In: Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP). Vol. 1. pp. 17–20.

Radocy, R. E., Boyle, J. D., 1979. Psychological foundations of musical behavior. C. C. Thomas.

Ratha, N. K., Connell, J. H., Bolle, R. M., 2001. Enhancing security and privacy in biometrics-based authentication systems. IBM systems Journal 40 (3), 614–634.

Sahidullah, M., Delgado, H., Todisco, M., Yu, H., Kinnunen, T., Evans, N., Tan, Z.-H., Sept. 2016. Integrated spoofing countermeasures and automatic speaker verification: an evaluation on ASVspoof 2015. In: INTERSPEECH 2016, Annual Conference of the International Speech Communication Association, September 8-12, 2016, San Francisco, USA. San Francisco.

Sahidullah, M., Kinnunen, T., Hanilçi, C., 2015a. A comparison of features for synthetic speech detection. In: Proc. Interspeech. Dresden, Germany.

Sahidullah, M., Kinnunen, T., Hanilçi, C., 2015b. A comparison of features for synthetic speech detection. In: INTERSPEECH. pp. 2087–2091.

Schorkhuber, C., Klapuri, A., Sontacch, A., July/August 2013. Audio pitch shifting using the constant-Q transform. Journal of the Audio Engineering Society 61 (7/8), 425–434.

Schrkhuber, C., Klapuri, A., Holighaus, N., Drfler, M., 6 2014. A Matlab toolbox for efficient perfect reconstruction time-frequency transforms with log-frequency resolution. In: Fazekas, G. (Ed.), Audio Engineering Society (53rd Conference on Semantic Audio). AES (Vereinigte Staaten (USA)).

Todisco, M., Delgado, H., Evans, N., 2016. A new feature for automatic speaker verification anti-spoofing: Constant Q cepstral coefficients. In: Proc. Odysssey. Bilbao, Spain.

Vetterli, M., Herley, C., Sep 1992. Wavelets and filter banks: theory and design. IEEE Transactions on Signal Processing 40 (9), 2207–2232.

Villalba, J., Lleida, E., 2011. Biometrics and ID Management: COST 2101 European Workshop, BioID 2011, Brandenburg (Havel), Germany, March 8-10, 2011. Proceedings. Ch. Detecting Replay Attacks from Far-Field Recordings on Speaker Verification Systems, pp. 274–285.

Wolberg, G., 1988. Cubic Spline Interpolation: a Review. Columbia University.

Wu, Z., Evans, N., Kinnunen, T., Yamagishi, J., Alegre, F., Li, H., 2015. Spoofing and countermeasures for speaker verification: A survey. Speech Communication 66, 130 – 153.

Wu, Z., Kinnunen, T., Evans, N., Yamagishi, J., 2014. ASVspoof 2015: the first automatic verification spoofing and countermeasures challenge evaluation plan.

Wu, Z., Kinnunen, T., Evans, N., Yamagishi, J., Hanilci, C., Sahidullah, M., Sizov, A., 2015. ASVspoof 2015: the first automatic speaker verification spoofing and countermeasures challenge. In: INTERSPEECH. Dresden, Germany.

Youngberg, J., Boll, S., Apr 1978. Constant-q signal analysis and synthesis. In: IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP). Vol. 3. pp. 375–378.