Odyssey 2018 The Speaker and Language Recognition Workshop
26-29 June 2018, Les Sables d'Olonne, France

AN AUDIO FINGERPRINTING APPROACH TO REPLAY ATTACK DETECTION ON ASVSPOOF 2017 CHALLENGE DATA

J. Gonzalez-Rodriguez, A. Escudero, D. de Benito-Gorron, B. Labrador and J. Franco-Pedroso

Audias-ATVS, Universidad Autonoma de Madrid (UAM), Spain
joaquin.gonzalez@uam.es

ABSTRACT

Replay attacks, where an impostor replays a genuine user utterance, are a major vulnerability of speaker verification systems. Two highly likely scenarios for replay attacks are either hidden recording of actual spoken access trials, or reusing previous genuine recordings in case of fraudulent access to transmission channels or storage devices. In both scenarios, an audio fingerprint-based approach comparing any access trial with all previous recordings from the claimed speaker perfectly fits the task of replay attack detection. However, ASVspoof 2017 rules did not allow the use of the original RedDots audio files (spoofed trials are replayed versions of RedDots), which disabled a fingerprint-based regular participation in the evaluation as those original files are necessary to build the bank of previous-access audio fingerprints. Then, we agreed with the organizers to run and submit on time a parallel fingerprint-based evaluation with exactly the same blind test data with an alternative but realistic (deployable) evaluation scenario. While we obtained an Equal Error Rate of 8.91% detecting replayed versus genuine trials, this result is not comparable for ranking purposes with those from actual participants in the Challenge as we used the original RedDots files. However, it provides insight into the potential and complementarity of audio fingerprinting, especially for high audio-quality attacks where state-of-the-art acoustic antispoofing systems show poor performance (the best ASVspoof 2017 system with global EER of 6.73% degraded to about 25% in condition C6 of high-quality replays), while our fingerprint-based antispoofer obtains an EER of 0.0% for the high-quality replays in condition C6, showing the complementarity of acoustic antispoofers for low-mid quality replays and fingerprint-based ones for mid-high quality replays.

1. INTRODUCTION

Once automatic speaker verification (ASV) technology is ready to be deployed in terms of discrimination ability between speaker's voices, a major concern for final users, vendors and their clients is its robustness to intentional fraudulent attacks from either skilled or naive impostors. Those attacks [1] can take the form of impersonation (mimicking), speech synthesis (text converted into speech similar to the target speaker), voice conversion (someone's speech into speech similar to the target speaker) or replay (use of pre-recorded speech from the target speaker). While speech synthesis and voice conversion have been object of relevant research efforts as those in ASVspoof 2015 [2], replay attacks are by far, as shown in table 6 in [1], the most accessible and effective among them, as no knowledge of speech technology is necessary for attackers and systems easily accept those recordings as genuine utterances from the claimed speaker. In fact, the incoming speech is a different acoustic version of a true genuine speaker utterance.

The first major initiative to systematically address replay attacks has been the recent ASVspoof 2017 Challenge [3, 4], which used a multiple-device replayed version [5] of the RedDots dataset [6], a short duration and variable phonetic content set of recordings intended for speaker recognition obtained from mobile devices across multiple countries with a large number of sessions over a long time-span.

Audio fingerprinting [7, 8, 9, 10, 11] is a solid domain, even with large commercial success as Shazam [9], which provides accurate, fast and efficient search of acoustically similar audio segments. The basic principle is to focus on acoustic (usually spectral) landmarks, which are robust to noise and reverberation, to be coded in some way (e.g. by pairs) into hashcodes. Each audio file is represented by a compact set of time-aligned hashcodes that are stored in a hashtable, allowing new (even very short) audio segments to be "fingerprinted" and easily searched and found in a big hashtable representing the audio repository. Remarkably, audio fingerprinting has shown to be very robust to replays by any audio device, allowing reliable detection of short music segments played through PA systems or low-quality loudspeakers.

With the objective of designing a speaker- and recording-independent evaluation, the rules of ASVspoof 2017 explicitly disallowed the use of the original RedDots recordings, which disabled the possibility of a regular participation in the evaluation using audio fingerprinting techniques for replay attack detection. However, we will show in this paper that those techniques perfectly fit the replay detection task in a real system, as previous access attempts can be already available, stored and indexed in hashtables, in any ASV system. While not abiding to the RedDots-prohibited evaluation rule, we have tested our system with ASVspoof 2017 evaluation data simulating a fingerprinting-based antispoofing scenario. We ran the same trials but in this alternative scenario, where RedDots original recordings were used as lookup hashtable, with our systems being finally scored by the ASVspoof 2017 organizers before publication of the evaluation keys.

2. ADEQUACY OF AUDIO FINGERPRINTING FOR REPLAY ATTACK DETECTION

There are three methods to obtain speech from genuine users that can be used for a replay attack. The first method is the use of segments of normal, spontaneous speech from the speaker in any natural context, unrelated to ASV access. This speech could be obtained surreptitiously through covert recordings with hidden microphones or wiretapping, or in the case of public characters simply accessing to public recordings of their voices in the media. However, as most speaker recognition-based access control systems rely upon text-dependent short passwords or passphrases, it will not be easy to find or construct the desired segments, even knowing the password, from those

10.21437/Odyssey.2018-43

natural conversations or interviews, being unlikely to have successful replay attacks in text-dependent systems from this method. But, if an attacker succeeds in this way, having no previous version of that genuine utterance in the system, a fingerprint detector will not be able to cope with this (less likely) attack.

The second method of speech obtention, representing the most likely scenario for obtaining the spoken password, is recording the speaker during an access by voice to the system, using close covert or far superdirective microphones. The attacker would obtain an acoustically different version of a true spoken access, which if not severely degraded will probably be enough to be given access to the system.

The third method, less likely as depends on highly skilled impostors acting as hackers, consists in accessing to a digital repository of previous recordings of access trials from the speaker or directly having access to the on-line audio recorded by the genuine speaker device. If the hacker succeeds, access to the ASV system will be granted, as the audio obtained is a perfect copy of a spoken utterance from the claimed speaker.

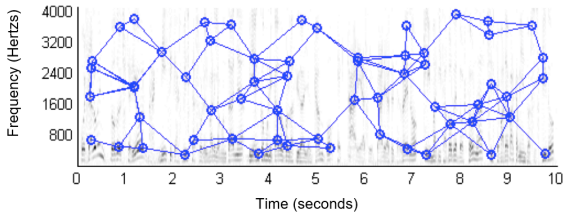

Figure 1. *Sample fingerprint in our broadcast quality fingerprinting system described in Section 3.*

Audio fingerprinting techniques, which allow for accurate, fast and efficient search of acoustically similar audio segments, perfectly fits the two latter more realistic, likely and dangerous scenarios, enabling the detection of playbacks of previous access trials to the system. Obviously, the audio fingerprint of every training utterance and access trial must be indexed and saved immediately, which can be done very easily as shown later. Moreover, as shown in Section 3.2, this replay detection through fingerprinting shows an amazing performance, probably error free, for medium and good quality replays, which are in fact those more likely to be granted access if the replay is not detected. The fingerprint detector will only miss severely degraded versions of the original recordings, but those are less problematic as they will be probably rejected anyway by the speaker verification system because of its low quality.

3. THE AUDIAS-ATVS-UAM AUDIO FINGERPRINTING SYSTEM FOR ASVSPOOF2017

We have developed an accurate and efficient audio fingerprinting system, originally intended for commercial radio ads detection. Our system [12] detects relevant pairs of spectral peaks (landmarks) and represents each pair through a hashcode 64 bits long. Hashcodes are obtained about 12 times per second on average, resulting in audio files being represented by a hashtable with just 96 bytes per second of audio (12 hashcodes/sec x 8 bytes/hashcode). For instance, the reference ads database used in [12], including 274 radio ads of 22 seconds length on average (200 Mbytes of audio in total, sampling at 16 kHz and using 16 bits per sample), is stored in a single hashtable of just 597 kbytes, a compression rate over 330 (or a reduction to 0.3% of the original size). In experiments with different length test segments with broadcast quality the detection performance was excellent with no false alarms, as shown in table 1. The system runs 10 times faster than real time in a low cost platform, as shown in table 2.

Table 1. *Performance of our broadcast ads detector for different length segments from 428 different-day repetitions of the 274 radio ads in the hashtable*

Segment Duration	Detection Rate (%)	False Positives (%)
10 seconds	100.0 %	0.0 %
5 seconds	100.0 %	0.0 %
1 second	97.6 %	0.0 %

Table 2. *Processing times per second of audio for 12 hashcodes per sec. searched in a 597 Kbyte hashtable (76497 hashcodes) with a 2-core AMD Athlon II 64 bits 2.2 GHz 4 GB RAM Windows 7 desktop computer*

Task	Time (ms)
Pre-processing	21.7
Landmark extraction	31.2
Landmark pairing	18.0
Hashcode generation	11.5
Search on hashtable	22.8
Scoring	2.0
Total	**105.4**
xRealTime	x10

3.1. Adaptation to RedDots and ASVspoof 2017 conditions

We tried to apply this ready-to-use broadcast-quality audio fingerprinting system detecting the files in ASVspoof 2017 evaluation data, consisting in genuine and replayed versions of RedDots recordings, into a hashtable generated with Part 01 of the original RedDots dataset (the part used to generate the replayed version). However, we found that even keeping the false positives to zero as above, the detection rate dramatically dropped, resulting in a relevant proportion of missed files. Two factors influenced this performance drop: firstly, the shorter length of the audio files as RedDots ones are just three seconds long on average, with plenty of files from one to three seconds length. Such short files produce fingerprints of 36 hashcodes on average, but only 12 or 24 hashcodes on plenty of 1 or 2 sec files, which increases the odds of not being detected. And secondly, the varying quality of the replayed files, which decreased the percentage of hashcodes detected because of the lower quality of replays.

We had to redesign our system to cope with ASVspoof 2017 evaluation data, allowing for a higher density of not-so-relevant landmarks (compared to the previous selection for broadcast audio) producing now about 70 hashcodes per second. Our hashtable for ASVspoof 2017 will compress the 3854 files in RedDots Part 01 with a total of 615034 hashcodes stored in 4.8 Mbytes, representing 366 Mbytes of audio at a compression rate about 76 (or a reduction to 1.3% of the original size), lower than in the former broadcast quality system as we have significantly increased the average number of hashcodes extracted per second.

Even in this higher density configuration, the computing performance is faster than real time in a desktop computer (see Table 3), and three aspects relative to computation times in fingerprinting processing reinforces its convenience to replay detection in ASV: firstly, in the reported experiments every input file is searched through the whole hashtable. However, in an actual ASV system, we will look just into the claimed-speaker repository of previous recordings, as a given attack is intended for a single registered speaker, reducing the search space in our experiments by the number of registered speakers. Secondly, searching in a big hashtable is an easy task to parallelize, for instance simply dividing the table into the number of available CPUs and using different parallel search threads (tables 2 and 3 report single thread searches). And finally, adding new hashcodes into the hashtable, what should be done with every new access to the system to keep the hashtable updated, takes just some milliseconds per second of audio (8 ms in the computer of table 3).

Table 3. *Processing times per second of audio extracting 70 hashcodes per second searched in the 4.8 Mbyte RedDots hashtable (615034 hashcodes) with a 4-core i5 Ubuntu 64 bits 3.1 GHz 8 GB RAM desktop computer*

Task	Time (ms)
Landmarks & hashcode generation	7
Search on hashtable	570
Total	**577**
xRealTime	x1.73

3.2. Detection performance with ASVspoof 2017 data

In order to keep the false alarm rate to 0% in this higher hashcode density scenario, we checked the maximum score obtained in different-file experiments (file A vs file B) both for RedDots (all 3854 files) vs RedDots and ASVspoof 2017 evaluation data (subset of 4780 files) versus RedDots, as shown in table 4, showing that in all cases the score (number of hashcodes detected) was under 7.

Table 4. *Distribution of different-file trials where "false" hashcodes are detected. Thresholds greater than 5 guarantee FA=0% for replays.*

# hashcodes detected	RedDots vs RedDots	#files	Replayed RedDots vs RedDots	#files
0	0.05%	2	0%	0
1	16.79%	647	28.28%	1353
2	54.72%	2109	54.54%	2607
3	25.09%	967	15.84%	757
4	2.31%	89	0.98%	47
5	0.91%	35	0.33%	16
6	0.13%	5	0%	0
7	0%	0	0%	0
total	*100%*	*3854*	*100%*	*4780*

In our detector, for scores higher than the threshold, the higher the number of hashcodes detected, the higher our confidence in the file being a replay. Then, in order to produce a meaningful score for the ASVspoof 2017 evaluation tools, expecting higher scores for higher confidence being a genuine file, our audio fingerprinting score will be reported as the negative of the number of hashcodes detected.

In figure 2, we show the histogram of normalized scores (number of hashcodes detected per second) obtained by each evaluation file in ASVspoof 2017 when compared to the hashtable with all genuine RedDots files. As all evaluation files are either genuine or replayed RedDots files, they all should be "found" in the RedDots hashtable. However, 12.3% of evaluation files do not score higher than the threshold (Th=5) remaining undetected. We hypothesize that this group of undetected files will mostly correspond to those replayed with the lower quality devices, which as explained in table 5 and figure 2 in [4], probably correspond to evaluation conditions C1 and C2. Curiously, C1-C2 represents in table 5 in [4] a 12.8% of the trials, quite similar to our 12.3% of undetected files.

Among the detected files, the exponential decay in figure 2 (from 0 to -40) seems to correspond to replayed files of varying quality, while the secondary lobe (from -40 to -80) seems high quality genuine files, averaging 60 hashcodes per second.

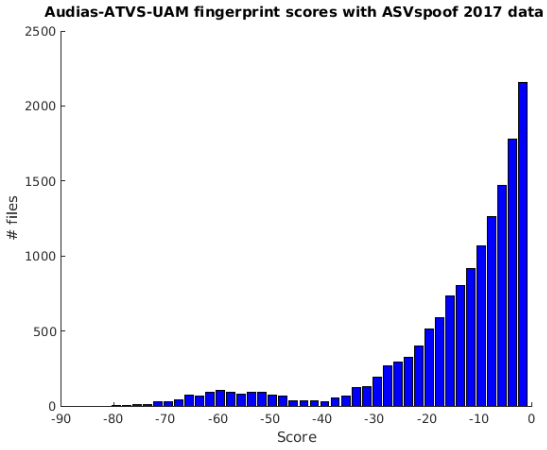

Figure 2. *Histogram of (the negative of the) number of hashcodes detected per second for ASVspoof 2017 evaluation data.*

In figure 3, the histogram of percentage of hashcodes detected in each evaluation file is shown, where the main distribution (percentages 0% to 70%) seems to correspond to replay files, while percentages over 90% (with the peak in 100%) seem to correspond to genuine files.

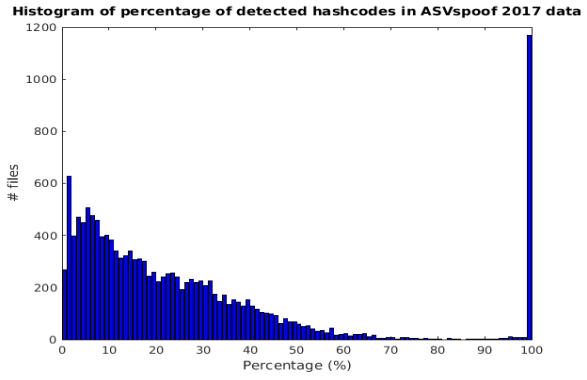

Figure 3. *Histogram of percentage of hashcodes detected for ASVspoof 2017 evaluation data.*

4. AN ALTERNATIVE BLIND EVALUATION WITH ASVSPOOF 2017 DATA

We have submitted three different systems (*primary*, *contrastive1* and *contrastive2*), simulating different antispoofing scenarios but always from the same fingerprint-based detection scores. As we built our hashtable with all RedDots files, the genuine trials in the evaluation are also detected as replays as they are already represented in the hashtable. This is not possible in a real ASV system, as if the access trial is really genuine it would never be in the list of previous access attempts from the claimed speaker, and the fingerprint detector will never obtain a score higher than the detection threshold. In order for our score lists to be processed and scored by the organizers evaluation tools, our "impossible" scores in genuine trials must be replaced in some form.

This genuine score replacement has been done in two ways: in our *primary* submission, we used an unrealistic approach (not possible in a real ASV system) taking advantage of our findings in figure 3, where scores which represent a percentage of detected hashcodes higher than 80% seemed to correspond to genuine trials. Once hypothesized as genuines, those scores are replaced with scores from the baseline system provided by the organizers [13], simulating that if undetected we rely in a secondary acoustic-only antispoofing system, as those in the regular evaluation. This approach, being unrealistic as used previous knowledge of genuine trials, allowed direct scoring of our primary submission without help from the organizers.

However, we needed the cooperation of the evaluation organizers to build our final score files in realistic, deployable conditions. We asked the organizers, the only ones with the knowledge of the evaluation keys, to replace our scores in genuine trials either by the baseline score (*contrastive1*) or by a positive integer (*contrastive2*), simulating what would happen in a real ASV system with those genuine trials. The former simulates the cascade of a fingerprint-based replay antispoofing system followed by an acoustic-only one, while the latter simulates a fingerprint-only antispoofing system.

4.1. Submitted results with ASVspoof 2017 data

Results from our submissions are summarized in table 5. Our main realistic submission, *contrastive1*, obtains an EER of 8.91%, even after having a 12.3% of missed detections (Section 3.2) in order to guarantee FA=0%. Those 12.3% of undetected trials plus all the genuine ones are replaced by baseline system scores (which has an EER of 28.89%).

Our also realistic fingerprint-only antispoofing system, *contrastive2*, obtains an EER of 11.49% with no use of additional acoustic antispoofing systems.

And our *primary*, even though unrealistic as using the fingerprint of genuines for genuine detection, gives an idea of the potential of audio fingerprinting for replay detection.

Table 5. *Results of Audias-ATVS-UAM blind submissions (using RedDots as hashtable)*

Submission	Realistic	Approach	EER (%)
primary	NO	Fingerprint + % of hashcodes detected	3.57%
contrastive1	YES	Fingerprint + baseline	8.91%
contrastive2	YES	Fingerprint	11.49%

5. POST-EVAL EXPERIMENTS

Once the evaluation labels were released by the organizers, we have performed different experiments to extend our analysis of the abilities of a fingerprinting approach for antispoofing.

First of all, in figure 4 we can observe a scatter plot of fingerprint versus baseline (acoustic) score. In acoustic-only antispoofing, all scores would be collapsed into the x-axis of the scatter plot, as shown for the baseline system by the overlap of genuine (blue) and spoof (red) trials which results in an EER about 29%. As easily observed, adding the fingerprint score dimension to the problem greatly simplifies the detection of replays, as fingerprint scores above 5 guarantee the file being a replay, which is obtained for 87.7% of spoof trials, reducing the confusable area to the lower part of the scatter plot (scores under 6).

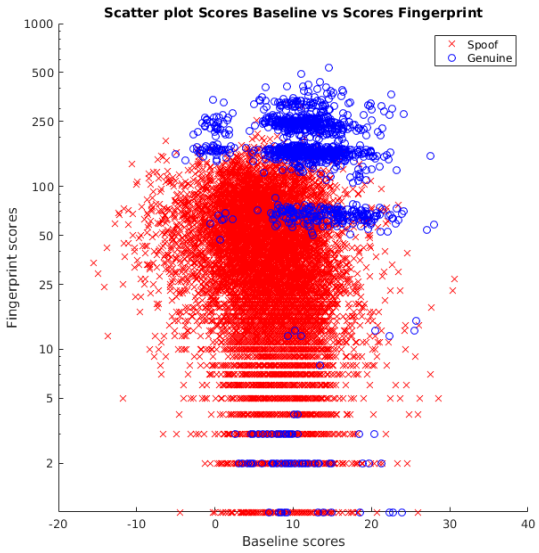

Figure 4. *Scatter plot of fingerprint score versus baseline scores for ASVspoof 2017 evaluation data.*

5.1. Influence of different values of fingerprint threshold

In our submission, from the results in table 4, we arbitrarily decided to set the fingerprint detection threshold to 5 (detection of 5 pairs of time aligned hashcodes). A higher threshold, as shown in table 6, will mean that more spoofed trials will not be detected, then the final decision will depend for more trials on the baseline system, which is an error prone system, resulting in higher EER.

Table 6. *Results in EER (%) for different fingerprint thresholds (Th=5 corresponds to submitted systems)*

Threshold	Contrastive 1	Contrastive 2
1	1.43%	1.44%
2	4.87%	4.96%
3	7.10%	7.79%
4	8.11%	10.02%
5	**8.91%**	**11.49%**
6	10.05%	14.42%
7	10.87%	16.42%

However, when we lower the threshold we obtain better evaluation results for worse performance of the fingerprint detector, which was designed as antispoofer for taking error-free decisions (we only "decided" a spoof if we score above 5, which means a correct detection from table 4). However, we see that the fingerprint in the evaluation conditions is much more reliable committing errors (threshold under 5) than the baseline system which follows, resulting in EERs lower than 1.5%.

5.2. Performance with high quality replays

Remarkably, fingerprint replay detection performance correlates with audio quality in an inverse manner than regular systems participating to ASVspoof 2017 do. As shown in figure 2 in [4], when high quality devices are used both for playback and recording (condition 6, C6), the replay attack is highly likely to be successful. Even for the best antispoofing system in the evaluation which showed a remarkable 6.73% global EER, the performance degraded in condition C6 to an EER around 25%. However, we will show that high quality replay conditions represent the easiest task for the fingerprint detector.

Unfortunately, conditions C1-C6 [4] trial lists are not available but detailed information about playback/recording devices and environment for every replay is provided, so we have selected the two following high-quality conditions:

- HQ1: professional and high-quality playback and recording devices (details in Annex I), with no restriction about the environment, resulting in 1982 medium/high-quality replays.
- HQ2: loopcable playback and recording devices (details in Annex I), resulting in 361 very high-quality replays, where HQ2 is a subset of HQ1. HQ2 seems to be (almost) identical to condition C6 in [4] (C6 was reported to consist of 365 loopcable replays).

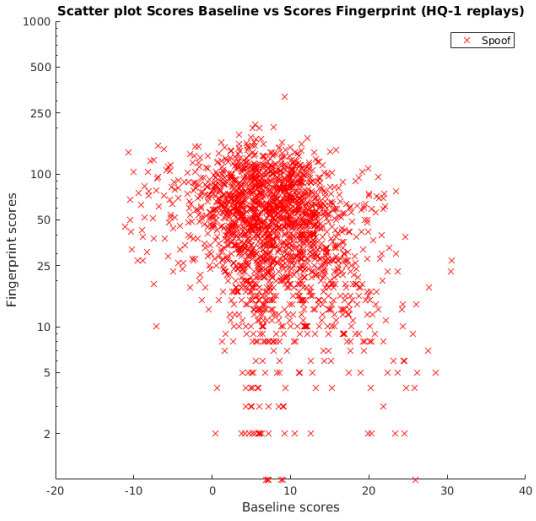

Figure 5. *Scatter plot of fingerprint score versus baseline scores for condition HQ1 (1982 replays) in ASVspoof 2017 evaluation data.*

As shown in figures 5 and 6, high-quality replays will be easily detected, especially for HQ2 (fig. 6) where there are no replays scoring close or under 5 so all replays will be perfectly detected in this HQ2 (~ C6) condition.

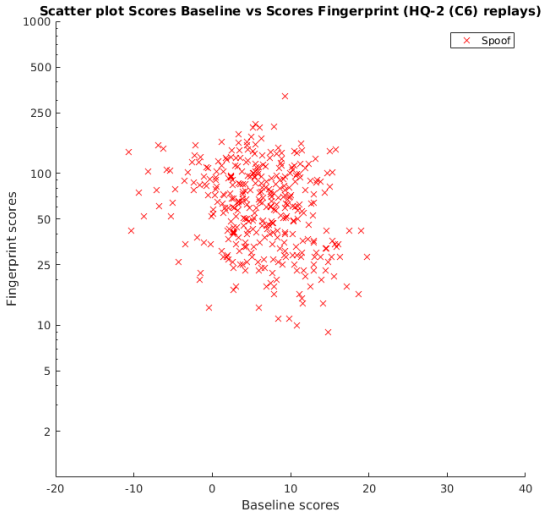

Figure 6. *Scatter plot of fingerprint score versus baseline scores for condition HQ2-C6 (361 replays).*

Moreover, in order to check the objective quality of HQ1 and HQ2 conditions, we time aligned the original and replayed files and computed their average cepstral distance, as shown in figures 7 and 8, clearly showing much smaller cepstral distances than the rest of the trials.

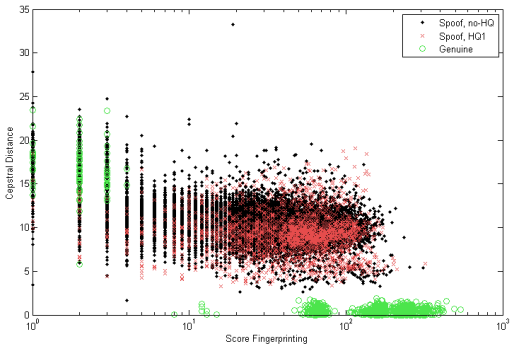

Figure 7. *Scatter plot of average cepstral distance versus fingerprint score for ASVspoof 2017 evaluation data (HQ1 replays in red).*

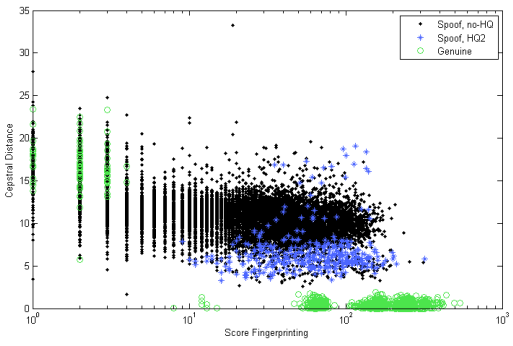

Figure 8. *Scatter plot of average cepstral distance versus fingerprint score for ASVspoof 2017 evaluation data (HQ2 replays in blue, HQ2 being a subset of HQ1).*

In table 7, we report the results of all the submitted systems in the HQ1 & HQ2 (~C6) conditions, where we expect an excellent performance of the reply detector. As shown, for the condition C6/HQ2 where the best acoustic antispoofing system in the evaluation degraded in EER from a global 6.93% to about 25% in C6 [4], the submitted fingerprint-based realistic systems have shown to be free of errors, which confirms the remarkable performance of fingerprint-based antispoofers in high quality replay conditions.

Table 7. *Analysis of results in EER (%) of submitted systems for different sets of reply files depending on their quality (HQ2 equivalent to condition C6 in [4])*

Replay	Baseline	Primary	Contrast1	Contrast2
All	29.61%	5.03%	8.91%	11.49%
HQ1	33.60%	1.66%	3.18%	3.18%
HQ2	26.59%	0.28%	**0.00%**	**0.00%**

5.3. Performance of ASV + Fingerprint Antispoofing

In this section, we will describe the joint performance of a phrase-and-speaker-dependent i-vector-based ASV system designed for RedDots data with and without the replay conditions in ASVspoof 2017, and with and without the fingerprint-based antispoofer described in this paper.

The i-vector-based ASV system is developed from SRE and Switchboard data, and phrase-and-speaker dependent models are built from RedDots data with three different repetitions of the same phrase by the speaker. Using the remaining RedDots data to test the system without replays we obtain 3242 target trials and 1080774 non-target trials, reporting a (zero-effort spoof) EER of 1.49%, which will be used as reference system for later experiments.

As speakers in RedDots and ASVspoof trials are not clearly associated, we decide to associate speaker identities and phrases from RedDots and ASVspoof2017 in two steps:

1. Replay files scoring over 5 in the fingerprint detector are assigned to the speaker/phrase at the origin of the hashtable where the hashcodes are found. That process correctly assigns about 87% of the evaluation files.
2. Replay files scoring 5 or less in the fingerprint detector are compared to every speaker+phrase model in the ASV system, and the reply is arbitrarily assigned to the speaker+phrase providing the highest score. Some errors will be committed with this arbitrary decision, but as the quality of those files is low, even if correctly assigned the ASV is likely to produce verification errors, so it seems a plausible option for the assignment.

Once the speaker identities and phrases are assigned between RedDots and ASVspoof2017, in table 8 we report the significant degradation observed in EER when replay spoofing is considered. A good speaker discriminant i-vector-based system for RedDots data, reporting an EER of 1.49%, turns into an error full system in the presence of replays with an EER of 31.65%. But if we consider only the high-quality replays (C6/HQ2 condition), the EER goes up to 48.09%, turning the ASV into an uninformative and unusable system at all.

Table 8. *Degradation in Speaker detection results in EER(%) of a speaker-and-phrase dependent i-vector-based ASV in three replay spoofing conditions: zero-effort spoofing, replay spoofing with all qualities in ASVspoof 2017, and replay spoofing only in very high-quality (loopcable, ~C6) replay condition*

	RedDots vs RedDots	RedDots vs All Replays	RedDots vs C6/HQ2 Replays
EER	1.49%	31.65%	48.09%

However, as shown in table 9, when we pre-screen the evaluation trials with fingerprint-based replay anti-spoofers, the global performance still degrades from 1.49% up to 4.38%, significant but really much better than the 31.65% obtained without antispoofing. But moreover, as no spoof detection errors have been reported in high quality trials, the increase in error is due to low quality trials, where our score in contrastive 1 depends just on the error-prone baseline system. That means that combining a better-than-the-baseline acoustic antispoofer for low-mid-quality trials, and a fingerprint-based one for mid-high-quality trials, seems a perfect combination for replay detection.

Table 9. *Reduction in Speaker detection EER (%) with ASVspoof 2017 data in a speaker-and-phrase dependent i-vector-based ASV when fingerprint-based replay antispoofing is applied (Contrastive 1 & 2)*

	Without antispoofing		Fingerprint-based replay antispoofing	
	ASV without Replays	ASV with Replays	Contrast 1	Contrast 2
EER	**1.49%**	**31.65%**	**4.38%**	**4.57%**
P_{FA} replays	-	-	9.76%	12.26%
P_{Miss} genuines	-	-	8.86%	0.0%

As shown in table 9, the increase in speaker EER from 1.49% to about 4.5% is due to the 9.76% and 12.26% of replays which are not detected (and then falsely accepted) respectively in contrastive 1 and 2, and to the 8.86% of not-detected genuines. For all those trials we rely then on the performance of the baseline system, which finally degrades slightly the speaker detection EER of the ASV system in the presence of replay spoofing.

Additionally, for high-quality replays, the ASV EER of 48.09% without antispoofing (see Table 8) turns with the fingerprint-based antispoofing into a false acceptance rate of the ASV of 0% (all HQ2/C6 replays are detected) and miss detection rate of 1.49% (if the acceptance threshold of the ASV is set to the EER threshold), which strongly reinforces the idea of using fingerprint-based antispoofing for high-quality replays.

6. CONCLUSIONS

In this paper, we have shown that audio fingerprinting perfectly fits the replay attack detection and countermeasure task. Our results in ASVspoof 2017 are not directly comparable with those from regular participants as we used in our hashtable the

genuine recordings of the replays in the evaluation data, which was not allowed. However, having and keeping updated a hashtable of previous access trials to an ASV system is totally realistic, while not allowed in the evaluation. With 47 out of 48 participants obtaining EERs from 12.39% to 45.82% (table 4 in [4], and a winner system with EER of 6.73% [4], our EER of 8.91% in the same replay detection task, with the same evaluation data, in realistic but alternative conditions, confirm the potential of audio fingerprinting for replay detection. Moreover, for high-quality replays, where acoustic replay antispoofers severely degrade their performance, the fingerprint detector perfectly detects all replays.

Being computationally feasible, and extremely accurate in its detections in high and medium quality replay scenarios, the most interesting fact is that fingerprinting perfectly complements ASV systems in their most vulnerable point, which is the replay of previous access attempts from legitimate users with high quality devices. In this case, state-of-the-art acoustic-only antispoofing is vulnerable to replays and ASV systems will easily accept those attacks once the spoof is not detected. However, we have shown that the detection of high-quality replayed versions of previous access attempts is a simple task to a good fingerprint detector.

Interestingly, only some low-quality replays are not detected by the fingerprint detector, but those are easily detected by state-of-the-art acoustic antispoofers, which makes the combination of both antispoofing technologies a perfect match for replay antispoofing.

7. ACKNOWLEDGEMENTS & CLARIFICATION

The authors thank spanish MEC for supporting this research under TEC2015-68172-C2-1-P, and the organizers of ASVspoof 2017 for on time scoring of our systems even with so many regular participants during the evaluation.

This paper does not correspond to a regular participation in the ASVspoof 2017 Challenge, and its performance should not be directly compared with actual participants in the Challenge. Our approach, based on recordings of previous access attempts by the speakers, is realistic and deployable and was blindly tested on time with the same ASVspoof 2017 test data, but does not comply with the rules of the ASVspoof 2017 Challenge.

8. REFERENCES

1. Z. Wu, N. Evans, T. Kinnunen, J. Yamagishi, F. Alegre & H. Li, “Spoofing and countermeasures for speaker verification: a survey”, Speech Communication, 66, 130-153, 2015.

2. Z. Wu, J. Yamagishi, T. Kinnunen, C. Hanilçi, M. Sahidullah, A. Sizov, N. Evans, M. Todisco and H. Delgado, “ASVspoof: the Automatic Speaker Verification Spoofing and Countermeasures Challenge”, IEEE Journal on Selected Topics in Signal Proc., vol.11(4), 588-604, June 2017.

3. T. Kinnunen, N. Evans, J. Yamagishi, K. A. Lee, M. Sahidullah, M. Todisco, H. Delgado, “ASVspoof 2017: Automatic Speaker Verification Spoofing and Countermeasures Challenge Evaluation Plan”, http://www.spoofingchallenge.org/, 2016.

4. T. Kinnunen, M. Sahidullah, H. Delgado, M. Todisco, N. Evans, J. Yamagishi, K. A. Lee, “The ASVspoof 2017 Challenge: Assessing the limits of replay spoofing attack detection”, Procs. of Interspeech 2017, available at http://www.asvspoof.org/asvspoof2017overview.pdf, 2017.

5. T. Kinnunen, M. Sahidullah, M. Falcone, L. Costantini, R. Gonzalez Hautamäki, D. Thomsen, A. Sarkar, Z.-H. Tan, H. Delgado, M. Todisco, N. Evans, V. Hautamäki, K. A. Lee, “RedDots Replayed: A New Replay Spoofing Attack Corpus for Text-Dependent Speaker Verification Research”, Proc. ICASSP 2017, 5395-5399, 2017.

6. K. A. Lee, A. Larcher, G. Wang, P. Kenny, N. Brummer, ¨D.A. van Leeuwen, H. Aronowitz, M. Kockmann, C. Vaquero,B. Ma, H. Li, T. Stafylakis, M.J. Alam, A. Swart, and J. Perez, “The RedDots data collection for speaker recognition,” in Proc. Interspeech, Dresden, Germany, 2015, pp. 2996–3000.

7. A. Wang, ”An Industrial Strength Audio Search Algorithm” in ISMIR. p. 7-13, Oct, 2003.

8. P. Cano, E. Batlle, T. Kalker and J. Haitsma, “A review of audio fingerprinting,” VLSI signal processing systems for signal, image and video technology, vol. 41, no 3, pp. 271-284, 2005.

9. A. Wang, “The Shazam music recognition service” Communications of the ACM, vol. 49, no. 8, pp. 44-48, 2006.

10. M. Malekesmaeiliand, R. Ward, “A local fingerprinting approach for audio copy detection”, Signal Processing, vol. 98, pp. 308 – 321, 2014.

11. B. L. Sturm, “The state of the art ten years after a state of the art: Future research in music information retrieval” Journal of New Music Research, vol. 43, no 2, pp. 147-172, 2014.

12. A. Escudero and J. Gonzalez-Rodriguez, “Desarrollo y evaluación de un sistema de detección de publicidad pregrabada basado en audio fingerprinting”, Proc. of URSI 2016, Madrid, Sept. 2016.

13. M. Todisco, H. Delgado, N. Evans, “Constant Q Cepstral Coefficients: A Spoofing Countermeasure for Automatic Speaker Verification”, Computer Speech and Language, 45, 516-535, 2017.

ANNEX I

Conditions HQ1 and HQ2 are defined using the following restrictions in replay and recording devices and environment conditions:

- HQ1 uses the following playback and recording devices in any recording environment:

  Playback Device:

  P03: 'Genelec 8020C studio monitor'

  P04: 'Genelec 8020C studio monitor (2 speakers)'

  P15: 'Edirol MA-15D studio monitor'

  P22: 'Behringer Truth B2030A studio monitor'

  P23: 'Focusrite Scarlett 2i2 audio interface line output'

  P24: 'Focusrite Scarlett 2i4 audio interface line output'

P25: 'Genelec 6010A studio monitor'

Recording Device:

R01: 'Zoom H6 handy recorder'

R05: 'Røde NT2 microphone'

R06: 'Røde smartLav+ microphone'

R09: 'Zoom H6 handy recorder with Behringer ECM8000 microphone'

R10: 'Zoom H6 handy recorder with MSH-6 microphone'

R11: 'Zoom H6 handy recorder with XY microphone'

R21: 'AKG C3000 microphone'

R22: 'SE electronic 2200a microphone'

R23: 'Focusrite Scarlett 2i2 interface line input'

R24: 'Focusrite Scarlett 2i4 interface line input'

R25: 'Zoom HD1 handy recorder'

- HQ2, which is a subset of HQ1, uses only the following playback and recording devices:

  P23: 'Focusrite Scarlett 2i2 audio interface line output'

  P24: 'Focusrite Scarlett 2i4 audio interface line output'

  R23: 'Focusrite Scarlett 2i2 interface line input'

  R24: 'Focusrite Scarlett 2i4 interface line input'

In fact, the same HQ2 list can be obtained adding to HQ1 the restriction to the two following environments:

E23: 'Studio'

E25: 'Analog wire 02'