INTERSPEECH 2020
October 25–29, 2020, Shanghai, China

POCO: a Voice Spoofing and Liveness Detection Corpus based on Pop Noise

Kosuke Akimoto[*], *Seng Pei Liew*[*†], *Sakiko Mishima*[*], *Ryo Mizushima*[*], *Kong Aik Lee*[†]

NEC Corporation, Japan

{kosuke_a, s.mishima, r-mizushima}@nec.com, {sengpei.liew, kongaik.lee}@gmail.com

Abstract

We present a new database of voice recordings with the goal of promoting research on protection of automatic speaker verification systems from voice spoofing, such as replay attacks. Specifically, we focus on the liveness feature of live speech, i.e., pop noise, and the corresponding voice recordings without this feature, for the purpose of combating spoofing via liveness detection. Our database includes simultaneous recordings using a microphone array, as well as recordings at various distances and positions. To the best of our knowledge, this is the first publicly available database that has been particularly designed to study the liveness features of voice recordings under various conditions.[1]

Index Terms: replay attack, spoofing attack, pop noise, microphone array, voice corpus

1. Introduction

Utilizing automatic speaker verification (ASV) as a form of biometrics authentication is finding real-life applications and gaining traction. For example, the Android operating system now allows users to unlock their smartphones by voice.[2] Voice biometrics solutions are also deployed to support financial services over the phone.[3] Moreover, recent advances, such as the use of x-vector [1], have dramatically improved the ASV performance and show potential for mass adoption.

On the other hand, ASV is susceptible to spoofing attacks. Spoofing may be conducted simply by replaying the recorded voice or through impersonation [2, 3]. More sophisticated attacks utilizing speech synthesis techniques are also feasible. Recent voice conversion and text-to-speech methods have advanced to the levels that are capable of producing nearly natural speeches [4, 5].

Defending ASV against spoofing attacks is an active research area. Most strategies focus on detecting artificial features in the spoofed speech which is not found in natural speech [6, 7]. This typically involves selecting a discriminative feature (front-end), and designing a good classifier (back-end), although end-to-end methods involving deep learning are gaining traction as well. See [8] for a comprehensive review.

The approaches mentioned above focus on exploiting the features found in the collected speech signals. Another defense strategy against spoofing attack is to make use of supplementary information such that liveness detection is possible. This approach aims to verify that an input is from a live speaker and not just the recording via additional information obtained during data acquisition. Various strategies have been proposed. In [9], two microphones were used to capture the time-difference-of-arrival (TDoA) changes in a sequence of phoneme sounds. [10] used the built-in speaker of a smartphone as a Doppler radar to transmit high-frequency acoustic sound and used the microphone to monitor the unique articulatory gesture of the user. Liveness detection using throat microphones has also been proposed [11].

In this work, we focus on liveness detection using pop noise. Pop noise is a phenomenon where the loudspeaker reproduces commonly unwanted noises due to the microphone picking up a variety of breathing noises. While such noise is reduced in far-field liveness detection settings, the microphone is usually placed close to the user and is able to capture such kind of acoustic features in most ASV applications. It is therefore a suitable feature for differentiating a live speech from a spoofed/replayed one within the liveness detection framework. Moreover, unlike the aforementioned detection methods, detecting pop noise is relatively simple, i.e., utilizing built-in microphones is in principle sufficient.

The major purpose of this paper is to present a new database, **POCO** (*POp noise COrpus*), to allow systematic study of pop noise. Existing publicly available databases have been focusing on spoofed speeches instead of features related to liveness. For example, the *Reddots Replayed* data set [12] provides a wide variety of recordings and re-recordings to facilitate the study of anti-spoofing. The 2019 ASVspoof challenge [13] includes data simulating physical environments and synthesized speech data (utilizing state-of-the-art speech synthesis methods). Our work aims to close this gap in the literature by providing a pop noise database that assesses the liveness feature of ASV.

This paper is organized as follows. We first motivate our study by comparing the current work with other works in the literature. Then, we illustrate how we collected the data. We explain in detail our strategies, processing techniques, as well as equipment and material used. Subsequently, we present the results of analysis based on several baseline methods. Finally, we conclude with suggestions of possible future research directions utilizing this database.

2. Comparison with other works

The use of pop noise to mitigate attacks on ASV has received little attention, arguably due to a lack of open-source database. [14] is the first utilizing pop noise to perform liveness detection. Other studies, i.e., [15, 16], extend this idea by utilizing phoneme-based pop noise and multi-channel pop noise detection. [17] has implemented a liveness detection system based on pop noise on smartphones.

Unfortunately, none of the studies mentioned above is reproducible as these studies do not make their data publicly available. We aim to provide a common database such that different methodologies can be compared meaningfully based on it.

The size of our database is larger and more diverse than that

[*]Equal contribution with authors in alphabetical order.

[†] SPL is currently at LINE Corporation, Japan; KAL is currently at Institute for Infocomm Research, A*STAR, Singapore

[1]The database can be found at https://github.com/aurtg/poco

[2]https://support.google.com/android/answer/9075927?hl=en

[3]https://www.sestek.com/case-studies/denizbank-vocal-passphrase-case-study/

http://dx.doi.org/10.21437/Interspeech.2020-1243

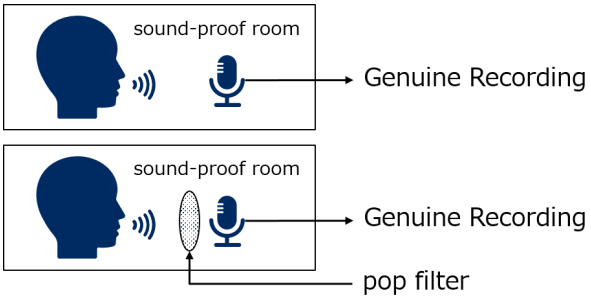
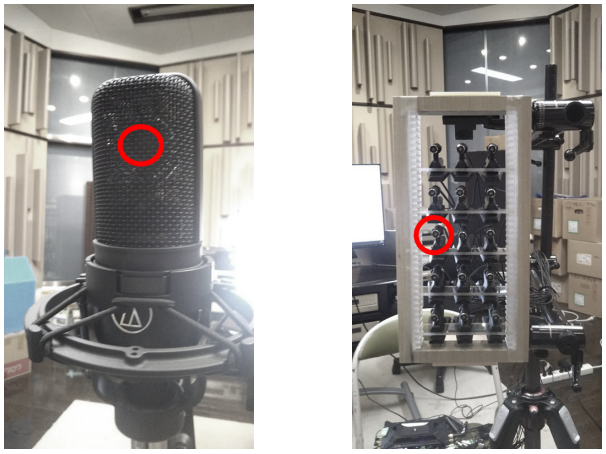


Figure 1: *An illustration of the setting used for the POCO data collections. Recording with pop noise, i.e., genuine recording without a pop filter (upper figure). This corresponds to the* ***RC-A*** *and* ***RC-B*** *subsets of data. Also shown is recording without pop noise, i.e., genuine recording with a pop filter (lower figure). This corresponds to the* ***RP-A*** *subset of data.*



(a) *AT4040.* (b) *AT9903.*

Figure 2: *Microphones used in the data collection (audio-technica AT9903 and audio-technia AT4040). Red circles indicate the position the subject's mouth was directed to during data collections.*

presented in [14]. Seventeen female subjects were hired in [14], whereas our database contains approximately 4 times more subjects with nearly equal ratio of genders. Another difference is the language used in the recordings: Japanese was used in [14] while utterances in English were recorded in POCO.

Moreover, our collected data contain several other distinctive features. First, we performed studies on pop noise with a microphone array. Voice manipulation device with multiple microphones are becoming popular, so this is in line with the architecture of modern voice controlled systems. For example, the Google Home Mini has two microphones, the Xiami Mi AI Speaker has six microphones, and the Amazon Echo Dot has seven microphones. Studying how pop noise depends on the direction of sound source can be an important research direction. Thus far, we find that only [18] has studied this aspect in the context of spoofing attack.

Second, we experimented with different distances between the microphone and the speaker's mouth. False negatives are expected to rise with respect to the microphone-mouth distance for any liveness detection algorithm based on pop noise. This serves as a future research direction to study the trade-off between security and usability within this framework.

3. Data Collections

3.1. Experimental Setups

Let us consider the attack scenario of voice spoofing. In order to perform voice spoofing, an attacker initially has to obtain a sample of the desired speaker's voice. This can be done via recording (eavesdropping) or speech synthesis. Then, the attacker replays the recording, which is to be captured by the targeted ASV device. In realistic cases, eavesdropping is usually performed at long distance, such that pop noise is not recorded in the attacker's recording device (for example, an attacker may secretly record the target's conversation using his or her phone at a public space). Furthermore, speech synthesis tools do not usually synthesize the effects of pop noise. This defines the setting of our data collections, where the illustration is shown in Figure 1.

We emulated scenarios explained above experimentally. We used 2 types of microphones, as described in Table 1 . We also utilized the F8 Zoom Multitrack Field Recorder which enables simultaneous recordings. Sampling rate is 22,050, and bit rate is 24bit. Our data set is organized into 3 parts. We next explain each of these in detail.

Recording with microphone A (RC-A): We recorded high-quality voices with the audio-technica AT4040 microphone. This subset of data represents genuine speaker recordings with pop noise. The speaker-microphone distance was fixed to be 10 cm.

Recording with microphone array (RC-B): We performed recordings with the microphone array (audio-technica AT9903 with windscreen taken off). We used 15 microphones in total arranged as in Figure 2b. The distance between the microphones was 1 cm horizontally and 1.75 cm vertically. The signals recorded by these microphones are synchronised by a pulse signal. The microphones were labeled mic 1, mic 2, ..., mic 15, starting from top left moving to right, the next row, etc.. This subset of data also represents genuine speaker recordings with pop noise. The subject's mouth was positioned to be approximately center left of the microphone array, i.e., mic 7. The reason we chose this positioning is that we expected the pop noise effect to be left-right symmetrical. We collected data with the following speaker-microphone distances: 5 cm, 10 cm, and 20 cm.

Eavesdropping (RP-A): We emulated the scenario where an attacker performs replay of a recording obtained at long distance, i.e., without pop noise. This was emulated in our experiment by recording the speaker's voice using the audio-technica AT4040 microphone with a pop filter (TASCAM TM-AG1) located between the subject and the microphone. The speaker-microphone distance was fixed to be 10 cm.

We consider this setup as the ideal eavesdropping scenario, where the speaker's voice is "replayed" perfectly, i.e., without any artificial artifacts of loudspeaker or speech synthesis tool, albeit without pop noise. One should be careful and take this into account when interpreting her results using **RP-A** as the above assumption may not be realistic enough depending on the use case.

3.2. Recording Subjects and Texts

We recruited 66 subjects (34 female and 32 male) to perform the recordings. The subjects were of various levels of English

Table 1: *Microphone settings*

Device	Frequency response [Hz]	Directionality
audio-technica AT9903	30-18,000	Omnidirectional
audio-technica AT4040	20-20,000	Cardioid

Table 2: *Words and the corresponding International Phonetic Alphabet (IPA) recorded in this work.*

IPA	Word	IPA	Word	IPA	Word	IPA	Word
b	bug	d	dad	f	fat	g	gun
h	hop	dʒ	exaggerate	k	kit	l	live
m	summer	n	funny	p	pin	r	run
s	sit	t	tip	tʃ	chip	ʃ	sham
v	five	w	quick	z	his	ʒ	division
θ	thongs	ð	leather	ŋ	pink	j	you
æ	laugh	eɪ	pay	s	end	i:	be
ɪ	busy	aɪ	spider	ɒ	honest	oʊ	open
ʊ	wolf	ʌ	monkey	u:	who	ɔɪ	join
aʊ	shout	ə	about	eə	chair	ɑ:	arm
ɜ:	bird	ɔ:	paw	ɪə	steer	ʊə	tourist

fluency and spoke different accents. The subjects' age ranged from 18 to 61. We asked each of the subject to speak the words listed in Table 2, which cover all 44 phonemes in English.

3.3. Recordings

Recording Environment: The subjects were instructed to speak at the center of a sound-proof room, following instructions projected on a monitor.

The subject was asked to speak the words listed in Table 2 consecutively for each set of the recordings mentioned in Section 3.1. We asked the subjects to repeat each set of the recordings 3 times. In order to avoid bias, we randomized the sequence of words for each session of recordings.

3.4. Data Post-processing

We split the recordings (containing multiple words) to the utterance (single-word) level. To do this, we utilized the DeepSpeech API [19] and the Librosa package [20].

We first captured the first and last frames of an utterance using the DeepSpeech API. Then, we split and saved the recordings with an interval of 0.5 seconds before and after the captured frames of the utterance.

The processed audio samples relying solely on the API inevitably contained noises such as the noises of coughing and sneezing. At the last stage of audio post-processing, we listened to and tuned manually all collected samples to discard such unwanted samples. Since the **RC-B** recordings of the microphone array were performed simultaneously, we inspected only one of the microphones' recording and performed the same post-processing on all other microphones' recordings based on it.

We have processed in total 66 (participants) $\times$ 44 (words) x 3 (sessions) $\times$ 47 (recordings) utterances.[4] After discarding unwanted noises and unclear utterances, we were left with a total of 402,391 utterances.

4. Experimentation

Pop noise appears as an irregular high energy region at very low frequency, as shown in Figure 3. We performed baseline analyses on the POCO data and present the results in this Section. Data analysis was performed at the utterance level.

[4]**RC-B** was collected with 15 microphones at 3 different distances; **RC-A** and **RP-A** were collected with 1 microphone each, ending up with 47 recordings in total.

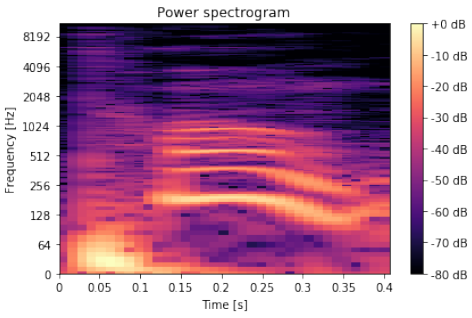


Figure 3: *An example of pop noise for the utterance "thong". High energy region is observed from the Figure at low frequency from 0 to 0.1 second.*

4.1. Signal processing

In order to detect pop noise, we first processed the recordings as follows. Given a recording sample, we first removed the silence frames using the Librosa package. Then, we applied the Short-Time Fourier Transform (STFT) to convert the signal into frequency components. An analysis window of size N, corresponding to precision 10Hz in the frequency domain was selected, and a hop size of $M = N/8$ was used over the time frames.

4.2. Detection methods

We hereby present two methods of detecting pop noise.

Shiota et al's algorithm: In order to detect pop noise, we devised a method similar to those presented in [14]. As we are interested in the low-frequency region of the signal, we define the measure $F_{L,avg}$ as the average of the Fourier transform (FT) bins within the interval $[0, F_{L,max}]$ for each frame. We chose $F_{L,max}$ to be 40Hz. Then, we computed the mean and standard deviation of energy over the frames. We counted the number of frames, M, containing $F_{L,avg}$ larger than three times its standard-deviation of the energy distribution. In [14], $M \geq 1$ was used as the criterion determining whether a sample contains pop noise or not. We found that choosing the threshold $M > 2$ was a better choice at obtaining optimal precision and recall.

Machine learning algorithm: We treated the task of detecting pop noise as a binary classification problem and solved it using machine learning methods. Samples from the **RC-A** data were labeled positive, and samples from the **RP-A** data were labeled negative. Let us now describe how we chose the input features to be fed to the machine learning models.

We first normalized $F_{L,avg}$ as defined above to zero mean and unit standard deviation with respect to the frames. Then, we chose the 10 frames with the largest normalized $F_{L,avg}$ and concatenate them in the descending order to treat them as the features of the machine learning models. We used the linear support-vector machine to classify the data. We also found that it performed better than the random forest algorithm.

4.3. RC-A and RP-A

Combining **RC-A** and **RP-A**, we present the performance of the pop noise detection algorithms described above.

Shiota et al's algorithm: For each word, we defined the number of true positive as the number of **RC-A** events detected as pop noise, true negative as the number of **RP-A** not detected as pop noise, false positive as the number of **RP-A** events detected as pop noise, and false negative as the number of **RC-A** events

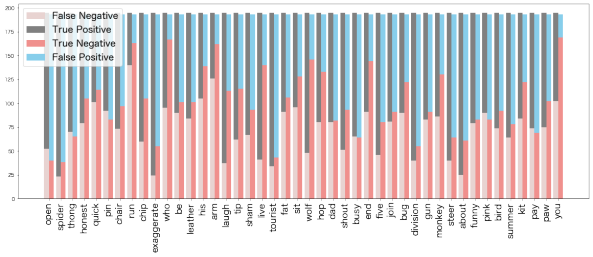


Figure 4: *Distribution of confusion matrix components, where the **RC-A** and **RP-A** data are used. The Shiota et al's algorithm described in text is used to determine if an utterance contains pop noise. Ground truth labels are fixed as follows: utterances from **RC-A** (**RP-A**) is labeled as positive (negative).*

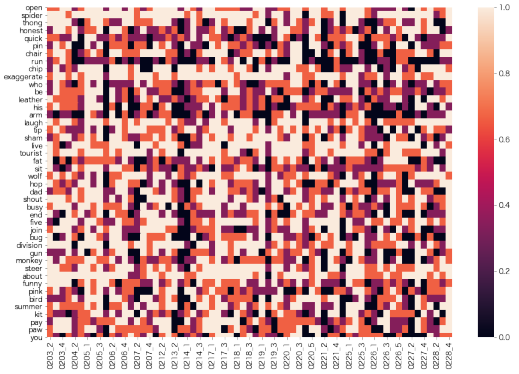


Figure 5: *Word-speaker accuracy matrix, where the **RC-A** and **RP-A** data are used. The Shiota et al's algorithm described in text is used to determine if an utterance contains pop noise. Ground truth labels are fixed as follows: utterances from **RC-A** (**RP-A**) is labeled as positive (negative).*

not detected as pop noise.

We show the results in Figure 4. Words that are expected to produce pop noise, such as "exaggerate", "wolf" and "laugh", true positives were relatively high, indicating that the pop noise feature is well captured by the algorithm.

Words such as "open" and "be" do not produce pop noise. Hence, for these words, the number of true positive and true negative are expected to be roughly the same. There are though exceptions to these, for example, "you".

Our observation was that, while the algorithm presented in [14] is able to approximately capture the properties of pop noise, it is rather simple and does not generalize well to all speakers. We also show the performance with respect to speaker in Figure 5. This could indicate that a more personalized pop noise detector should be designed to adapt to the unique characteristics of an individual in realistic applications.

Machine learning algorithm: We trained and performed a 5-fold cross-validation on the data. The mean accuracy score of each word is shown in Figure 6. It is observed that, for words producing pop noise, such as "wolf", "pay" and "paw", the accuracy scores are relatively high (i.e., up to 0.8). For other words, the difference in the accuracy scores is relatively mild. It should be noted that, the main purpose of this work is to provide a public database of pop noise, and the analysis presented here should be treated as the baseline to be compared with more sophisticated analyses in the future.

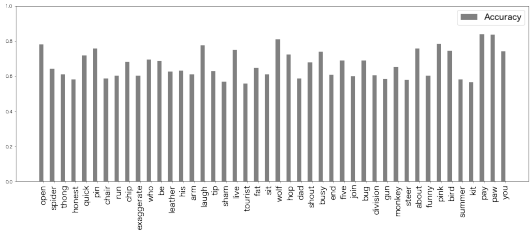


Figure 6: *Mean accuracy scores of each word, where the **RC-A** and **RP-A** data are used. The machine learning (SVM) algorithm described in text is used to determine whether an utterance contains pop noise or not. Ground truth labels are fixed as follows: utterances from **RC-A** (**RP-A**) is labeled as positive (negative).*

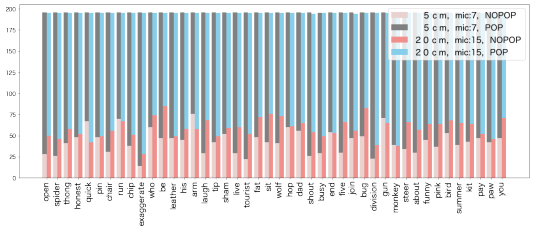


Figure 7: *Pop noise detection rates of each word, where the **RC-B** data are used, and the Shiota et al's algorithm is applied. Here, we compare the detection rates of mic 7, positioned 5 cm away from the subject, with those of mic 15, positioned 20 cm away from the subject.*

4.4. RC-B

Using **RC-B**, we briefly studied how the pop noise effect changes with respect to the position and angle of the microphone relative to the mouth. Utilizing the Shiota et al's algorithm described above, we compared the effects of pop noise between a microphone located at the center of the mouth (mic 7), positioned 5 cm away from the subject, and a microphone located at the bottom right (mic 15), positioned 20 cm away from the subject. It can be observed from Figure 7 that, mic 7 generally detects utterances with pop noise more frequently than mic 15. This is not surprising, as the pop noise feature is better captured by the microphone located nearer to the subject.

5. Conclusion

We have presented and described the POCO database, which focuses on pop noise as a liveness feature for ASV. Recordings of voices with and without using a pop filter, and recordings utilizing a microphone array at various positions have been performed. We have also analyzed the collected data utilizing several methods.

We hope that this database can serve as a foundation for future liveness detection research that focuses on utilizing pop noise. Some possible future research directions include the following. One can consider strengthening security by additionally incorporating a replay detector, which acts as a defense against replay attacks, including the threat of replaying voices with pop noise. Some other sophisticated related impersonation attack scenarios are worth studying as well, e.g., bypassing the pop noise detector by imitating/synthesizing the noise of breathing while replaying the "clean" recording.

6. References

[1] D. Snyder, D. Garcia-Romero, G. Sell, D. Povey, and S. Khudanpur, "X-vectors: Robust DNN embeddings for speaker recognition," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2018, Calgary, AB, Canada, April 15-20, 2018*, 2018, pp. 5329–5333. [Online]. Available: https://doi.org/10.1109/ICASSP.2018.8461375

[2] J. Lindberg and M. Blomberg, "Vulnerability in speaker verification - a study of technical impostor techniques," in *Sixth European Conference on Speech Communication and Technology, EUROSPEECH 1999, Budapest, Hungary, September 5-9, 1999*, 1999. [Online]. Available: http://www.isca-speech.org/archive/eurospeech_1999/e99_1211.html

[3] Y. Gao, R. Singh, and B. Raj, "Voice impersonation using generative adversarial networks," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2018, Calgary, AB, Canada, April 15-20, 2018*, 2018, pp. 2506–2510. [Online]. Available: https://doi.org/10.1109/ICASSP.2018.8462018

[4] Y. Wang, R. Skerry-Ryan, D. Stanton, Y. Wu, R. J. Weiss, Z. Y. Jaitly, Y. Xiao, Z. Chen, S. Bengio, Q. Le *et al.*, "Tacotron: Towards end-to-end speech synthesis," in *INTERSPEECH*, 2017.

[5] W. Ping, K. Peng, and J. Chen, "Clarinet: Parallel wave generation in end-to-end text-to-speech," *ICLR*, 2018.

[6] P. L. D. Leon, V. R. Apsingekar, M. Pucher, and J. Yamagishi, "Revisiting the security of speaker verification systems against imposture using synthetic speech," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP 2010, 14-19 March 2010, Sheraton Dallas Hotel, Dallas, Texas, USA*, 2010, pp. 1798–1801. [Online]. Available: https://doi.org/10.1109/ICASSP.2010.5495413

[7] Z. Wu, C. E. Siong, and H. Li, "Detecting converted speech and natural speech for anti-spoofing attack in speaker recognition," in *INTERSPEECH 2012, 13th Annual Conference of the International Speech Communication Association, Portland, Oregon, USA, September 9-13, 2012*, 2012, pp. 1700–1703. [Online]. Available: http://www.isca-speech.org/archive/interspeech_2012/i12_1700.html

[8] M. Sahidullah, H. Delgado, M. Todisco, T. Kinnunen, N. W. D. Evans, J. Yamagishi, and K. Lee, "Introduction to voice presentation attack detection and recent advances," in *Handbook of Biometric Anti-Spoofing - Presentation Attack Detection, Second Edition*, 2019, pp. 321–361. [Online]. Available: https://doi.org/10.1007/978-3-319-92627-8_15

[9] L. Zhang, S. Tan, J. Yang, and Y. Chen, "Voicelive: A phoneme localization based liveness detection for voice authentication on smartphones," in *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, Vienna, Austria, October 24-28, 2016*, 2016, pp. 1080–1091. [Online]. Available: https://doi.org/10.1145/2976749.2978296

[10] L. Zhang, S. Tan, and J. Yang, "Hearing your voice is not enough: An articulatory gesture based liveness detection for voice authentication," in *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security, CCS 2017, Dallas, TX, USA, October 30 - November 03, 2017*, 2017, pp. 57–71. [Online]. Available: https://doi.org/10.1145/3133956.3133962

[11] M. Sahidullah, D. A. L. Thomsen, R. G. Hautamäki, T. Kinnunen, Z. Tan, R. Parts, and M. Pitkänen, "Robust voice liveness detection and speaker verification using throat microphones," *IEEE ACM Trans. Audio Speech Lang. Process.*, vol. 26, no. 1, pp. 44–56, 2018. [Online]. Available: https://doi.org/10.1109/TASLP.2017.2760243

[12] T. Kinnunen, M. Sahidullah, M. Falcone, L. Costantini, R. G. Hautamäki, D. A. L. Thomsen, A. K. Sarkar, Z. Tan, H. Delgado, M. Todisco, N. W. D. Evans, V. Hautamäki, and K. Lee, "Reddots replayed: A new replay spoofing attack corpus for text-dependent speaker verification research," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2017, New Orleans, LA, USA, March 5-9, 2017*. IEEE, 2017, pp. 5395–5399. [Online]. Available: https://doi.org/10.1109/ICASSP.2017.7953187

[13] M. Todisco, X. Wang, V. Vestman, M. Sahidullah, H. Delgado, A. Nautsch, J. Yamagishi, N. W. D. Evans, T. Kinnunen, and K. A. Lee, "Asvspoof 2019: Future horizons in spoofed and fake audio detection," *CoRR*, vol. abs/1904.05441, 2019. [Online]. Available: http://arxiv.org/abs/1904.05441

[14] S. Shiota, F. Villavicencio, J. Yamagishi, N. Ono, I. Echizen, and T. Matsui, "Voice liveness detection algorithms based on pop noise caused by human breath for automatic speaker verification," in *INTERSPEECH 2015, 16th Annual Conference of the International Speech Communication Association, Dresden, Germany, September 6-10, 2015*. ISCA, 2015, pp. 239–243. [Online]. Available: http://www.isca-speech.org/archive/interspeech_2015/i15_0239.html

[15] ——, "Voice liveness detection for speaker verification based on a tandem single/double-channel pop noise detector," in *Odyssey 2016: The Speaker and Language Recognition Workshop, Bilbao, Spain, June 21-24, 2016*, 2016, pp. 259–263. [Online]. Available: https://doi.org/10.21437/Odyssey.2016-37

[16] S. Mochizuki, S. Shiota, and H. Kiya, "Voice liveness detection using phoneme-based pop-noise detector for speaker verification," in *Odyssey 2018: The Speaker and Language Recognition Workshop, 26-29 June 2018, Les Sables d'Olonne, France*, A. Larcher and J. Bonastre, Eds. ISCA, 2018, pp. 233–239. [Online]. Available: https://doi.org/10.21437/Odyssey.2018-33

[17] Q. Wang, X. Lin, M. Zhou, Y. Chen, C. Wang, Q. Li, and X. Luo, "Voicepop: A pop noise based anti-spoofing system for voice authentication on smartphones," in *2019 IEEE Conference on Computer Communications, INFOCOM 2019, Paris, France, April 29 - May 2, 2019*. IEEE, 2019, pp. 2062–2070. [Online]. Available: https://doi.org/10.1109/INFOCOM.2019.8737422

[18] Y. Gong, J. Yang, J. Huber, M. MacKnight, and C. Poellabauer, "Remasc: Realistic replay attack corpus for voice controlled systems," *CoRR*, vol. abs/1904.03365, 2019. [Online]. Available: http://arxiv.org/abs/1904.03365

[19] A. Y. Hannun, C. Case, J. Casper, B. Catanzaro, G. Diamos, E. Elsen, R. Prenger, S. Satheesh, S. Sengupta, A. Coates, and A. Y. Ng, "Deep speech: Scaling up end-to-end speech recognition," *CoRR*, vol. abs/1412.5567, 2014. [Online]. Available: http://arxiv.org/abs/1412.5567

[20] B. McFee, C. Raffel, D. Liang, D. P. Ellis, M. McVicar, E. Battenberg, and O. Nieto, "librosa: Audio and music signal analysis in python," 2015.