INTERSPEECH 2023
20-24 August 2023, Dublin, Ireland

VoxTube: a multilingual speaker recognition dataset

Ivan Yakovlev, Anton Okhotnikov, Nikita Torgashov,
Rostislav Makarov, Yuri Voevodin, Konstantin Simonchik

ID R&D Inc., New York, USA

{yakovlev,ohotnikov,torgashov,makarov,voevodin,simonchik}@idrnd.net

Abstract

The objective of this paper is to advance the development of technologies in the fields of speaker recognition and speaker identification by introducing a large labeled audio database VoxTube collected from the open-source media.

We propose a fully automated unsupervised approach for audio labeling that requires any pre-trained speaker recognition model. Collected with this approach from videos with CC BY license the VoxTube dataset contains more than 5.000 speakers with more than 4 million utterances pronounced in more than 10 languages. In our paper we show the VoxTube's high generalization ability across multiple domains by evaluating the accuracy metrics on various speaker recognition benchmarks. We also show how well this dataset complements an already existing VoxCeleb2 dataset.

Index Terms: dataset, speaker recognition, speaker verification, speaker identification

1. Introduction

Despite the fact of a good quality on the publicly available benchmarks and low amount of errors in scientific challenges [1], [2], [3], [4], speaker recognition is still a challenging task when it comes to the industry deployments. The voice biometrics providers are required to deliver a solution unbiased towards multiple domains combining varying features such as language and age of a speaker, as well as to be robust in any acoustic environment and towards any audio recording equipment that can include varying preprocessing frontends. There are multiple competitions focused on expanding the boundaries of speaker recognition in each domain independently [5], [6], [7], [8]. Nevertheless, in order to produce robust speaker embeddings and to provide the decent performance within any domain it is important to have a big, representative and well-labeled training data for supervised training techniques.

As it is shown in recent publications, amount of training speech data is important for both, self-supervised and supervised training approaches [9], [10]. That being said, self-supervisedly trained models show themselves as good feature extractors that used for a following fine-tuning for a specific task. This is also relevant for a supervisedly trained models, where the backbone is pre-trained on the speech data with a speaker recognition training loss and then it is used as a feature extractor for auxiliary tasks such as spoofing detection, language classification and so on [11]. Thus one big and labeled dataset could be utilized for both self-supervised and supervised training techniques. However, the labeling could become the bottleneck as it is usually cost expensive and slow, as it requires human assessors work.

Being motivated by the work of the researchers from the Oxford VGG group [12], [13] we came to an idea of expanding and enhancing the power of open-source media and VoxCeleb2 [13] dataset and came-up with a customly developed unsupervised collection and labeling pipeline for YouTube data that utilizes the audio clips only. Such pipeline requires a pre-trained speaker embedding extractor model that could be successfully trained on the VoxCeleb2 data [14].

Our paper is organised in the following way. In Chapter 2 we give an overview of existing speaker recognition datasets. Chapter 3 sheds a light on our data collection and labeling method. In this chapter we also provide the description and main statistics of the collected dataset. Chapter 4 contains our baseline CNN model architecture overview as well as the experimental setups. In Chapter 5 we present the accuracy metrics for VoxTube training data and report the results of training our baseline model using VoxTube and VoxCeleb2 together across publicly available testing benchmarks. Finally, Chapter 6 sums up our conclusions.

The VoxTube dataset can be downloaded from the project web page[1] at https://idrnd.github.io/VoxTube/.

2. Related works

The popular choices of data for training the speaker recognition models for a microphone (16 kHz) channel are LibriSpeech [15], Mozilla Common Voice [16], VoxCeleb2 [13]. However, there are a few limitations in each of them. For example, LibriSpeech contains the english language utterances only. While Mozilla Common Voice has a great lingual coverage and a lot of speakers, it lacks the variability of recording sessions within each speaker. In this sense VoxCeleb2 is a good choice for speaker embeddings model training: it delivers a big amount of speakers with a decent lingual coverage. The details of VoxCeleb2 dataset are presented in [13] as well as a collection method which includes the face recognition model.

While the VoxCeleb2 is a current number one pick dataset when it comes to speaker recognition, there is still room for improvements, for example expanding the lingual coverage by collecting more non-English speakers or increasing the number of speakers in the dataset. As a result, we came to a conclusion to expand the VoxCeleb2 dataset by collecting the additional non-overlapping speakers. Moreover, we introduce a new collection scheme that does not require a pre-trained face verification model which limits the data collection of genres other than interviews, for example, video gaming or devices unpacking genre.

[1] https://github.com/IDRnD/VoxTube

10.21437/Interspeech.2023-1083

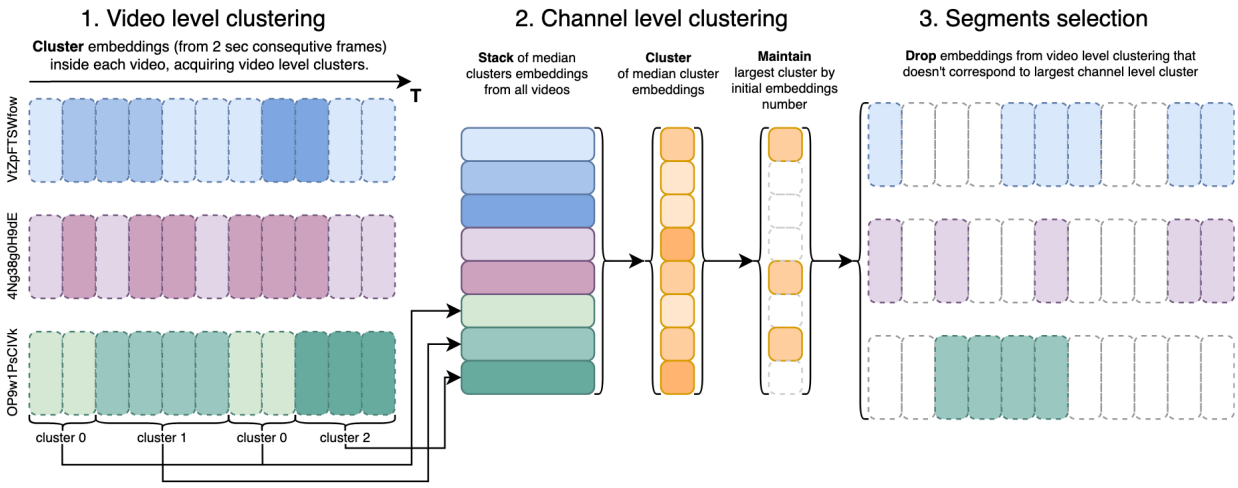

Figure 1: *Scheme of filtration pipeline for one channel*

3. VoxTube

3.1. Description

The VoxTube dataset was inspired by the VoxCeleb2 [13] dataset, it also uses YouTube as its main and only source of data. VoxTube contains over 4M utterances from more than 5K speakers extracted from CC BY videos uploaded to YouTube.

In Table 1 we show the main statistics of a VoxTube dataset. The language and gender distributions of the dataset are shown in figure 2. VoxTube has a 60%/40% gender distribution same as in VoxCeleb2. The language distribution in VoxTube differs from the distributions in VoxCeleb-(1,2) datasets. VoxTube also contains English as a predominant language accounting for approximately 30 percent. Yet such languages as Russian, Spanish and Portuguese are widely represented in VoxTube. We assume that language extension to the existing VoxCeleb language distribution can give a noticeable performance boost in a wide range of testing domains when combined with the VoxCeleb2 dataset. While collecting, we have also validated that VoxTube has no speakers overlap with VoxCeleb-(1,2).

Table 1: *VoxTube dataset statistics*

Dataset	VoxCeleb2	VoxTube
# of POIs	6,112	5,040
# of videos	150,480	306,248
# of utterances	1,128,246	4,439,888
# of hours	2,442	4,933
Avg # of videos per POI	25	61
Avg # of utterances per POI	185	881
Avg length of utterances (s)	7.8	4.0

3.2. Collection pipeline

Borrowing the main idea of searching speakers on YouTube from VoxCeleb2 paper [13], we came up with a hypothesis, that there is a number of YouTube channels, that have predominantly one person speaking. Mostly, such channels are easy to spot visually given only grid of video previews, or automatically by clustering sequences of audio speaker embeddings from videos and filtering out channels that don't have a predominant cluster. Based on this idea we've built our data collection pipeline.

Stage 1. Channels acquisition and filtering. We processed metadata for multiple YouTube channels and filtered them by a minimal number of available CC BY videos, subscribers, channel topic (video blogging, DIY, unboxing, gaming, education, etc.) and some other attributes presented in meta.

Stage 2. Audio extraction. We extracted audio from all the CC BY videos for each channel that passed Stage 1 of filtering. All audios were decoded into 16-bit PCM wav format from webm and mp4 containers that video hosting platforms use to store video altogether with audio data. That being said, at this stage we completely dropped any visual information from the source.

Stage 3. Extracting embeddings. First, we removed the silence regions from audio using a simple Voice Activity Detection (VAD) model. Then, with a help of a pre-trained on the VoxCeleb2 [13] dataset ResNet48 [11] speaker recognition model we extracted embeddings for each audio with consecutive and non-overlapping 2-seconds windows. This resulted in speech representation tensors of a shape (T, D), where T = (duration of audio) / 2 seconds, and D - is an embedding length.

Stage 4. Filtration. We have applied a two-stage filtration (see fig. 1) based on the Hierarchical Agglomerative Clustering (HAC) over the extracted embeddings. Details of the filtration stage are covered in the following subsection. This stage outputs are: ready for training audio segments of a predominant speaker cluster per channel and a corresponding median embedding per such cluster.

Stage 5. Duplicates removal. Finally, we compared median embeddings between each other and performed duplicates removal by dropping one channel in a pair that had a high cosine similarity score compared to the minimal threshold value. In similar manner, we removed the speakers that duplicate speakers from VoxCeleb-(1,2) at this stage.

3.3. Filtration

Filtration is heavily based on the assumption that most videos within a channel contain a significant part (more than 30%) of

a predominant speaker speech, which is the case for half of YouTube channels. The main goal of the filtration stage is to decide what speaker if present should be picked as a target speaker and to eliminate any non-relevant segments that may contain a collateral speaker and various noises. Both aims could be successfully achieved automatically via filtration. The filtration algorithm (fig. 1) could be split into 3 parts:

Stage 1. Video-level embeddings clustering. In the first part of filtration, we consider each video independently from other videos within the same channel. We cluster a sequence of audio embeddings using HAC. For this, we utilized the scikit-learn [17] implementation of HAC. We used a cosine similarity as a metric, average linkage, and a distance threshold equal to 0.6, which is the only tuneable hyperparameter here. The distance threshold is directly related to the pre-trained verification model that is used to generate speech representations. This stage output is a cluster segmentation map and an average embedding of each cluster. As an average, we used a median embedding due to its robustness to the outliers.

Stage 2. Channel-level embeddings clustering. In the second stage of filtration we consider whole channel and stack all video-level median cluster embeddings into one array and perform clustering with the same algorithm and parameters as described in Stage 1. We then choose the largest channel-level cluster by tracing back video-level clusters sizes and summing them up for each channel-level cluster. We declare the largest channel-level cluster to be a cluster of a predominant speaker.

Stage 3. Segments selection. Finally, we perform segments selection by tracing down the timestamps of video-level embeddings that belong to the largest channel-level cluster.

4. Experiments

To evaluate the impact of the VoxTube dataset on speaker recognition task we have run a couple of experiments using Convolutional Neural Network architecture ResNet [18].

4.1. Architecture

We conducted our experiments based on the ResNet48 architecture, which is a modification of ResNet34 [14]. Details of the architecture are presented in Table 2.

4.2. Data augmentation

As sources of augmentation we used the MUSAN [19], DEMAND [20] and DCASE [21] noise datasets and a database of real room impulse responses (RIRs) [22]. The following 5 types of noise augmentations were applied on-the-fly during the training:

- **Music**: A single music file was randomly picked from the MUSAN and summed with the original audio with 5-15 dB SNR. The duration of noise was matched against the duration of a training utterance.
- **Noise**: Randomly selected noise file from MUSAN was added to the original recording with 0-15 dB SNR.
- **Babble**: Between three and seven utterances of unique speakers were randomly picked from MUSAN, summed together, and then added to the original signal with 10-20 dB SNR.
- **DEMAND**: Randomly picked noise file from the DEMAND was summed with the training utterance with 0-15 dB SNR.
- **DCASE**: Randomly picked noise file from the DCASE was summed with the training utterance with 0-15 dB SNR.

Table 2: *ResNet48 architecture*

Layer name	Output (C × F × T)	Structure
Conv2D	C × 80 × T	96, 3×3, stride=1
ResBlock-1	C × 80 × T	$\begin{bmatrix} 3\times3, 96 \\ 3\times3, 96 \end{bmatrix} \times 6$
ResBlock-2	C × 40 × T/2	$\begin{bmatrix} 3\times3, 128 \\ 3\times3, 128 \end{bmatrix} \times 8$
ResBlock-3	C × 20 × T/4	$\begin{bmatrix} 3\times3, 160 \\ 3\times3, 160 \end{bmatrix} \times 6$
ResBlock-4	C × 10 × T/8	$\begin{bmatrix} 3\times3, 256 \\ 3\times3, 256 \end{bmatrix} \times 3$
Flatten (C, F)	2560 × T/8	-
StatsPooling	5120	-
Dense	256	-
AM-Softmax	Num. of speakers	-

We also applied a 30% probability to reverberate each training sample via convolution with a randomly picked response from RIRs.

4.3. Implementation details

4.3.1. Input features

As input, we used 80-dimensional Mel filter bank log-energies with a 25 ms frame length and 10 ms step with 512 FFT size over the 20-7600 Hz frequency range. When testing the models, the 8-second input segments were used.

4.3.2. Loss function

As a loss function we used an Additive Margin Softmax (AM-Softmax) loss [23]. This loss function reduces an interclass variance via the margin penalty that is applied to the target class logit. We used the scale 40 and the maximum value of margin was set to 0.3 according to [14].

4.3.3. Training

We trained the speaker recognition models using a batch size of 512 and 30 training epochs, with each epoch consisting of 5000 steps. For each batch, we sampled 512 unique speakers and took a single utterance for each speaker. From each utterance in the batch, we randomly cropped a 2-second audio segment. During training, we adjusted the learning rate and a margin in the AM-Softmax loss function. The learning rate scheduler had three phases: warmup, plateau, and decay. In the warmup phase, the learning rate was linearly increased from 1e-5 to 0.1, while the margin was set to 0 for the first two epochs. In the plateau phase, the learning rate was fixed at 0.1 and the margin was linearly increased from 0 to 0.3 for the next six epochs. Once the margin reached its maximum value, the learning rate was decreased exponentially with a rate of 0.5 every two epochs in the decay phase. We also applied L2-norm regularization with 1e-4 coefficient to all model weights.

For the joined training on both datasets VoxCeleb2 and VoxTube we increased the number of epochs and slightly

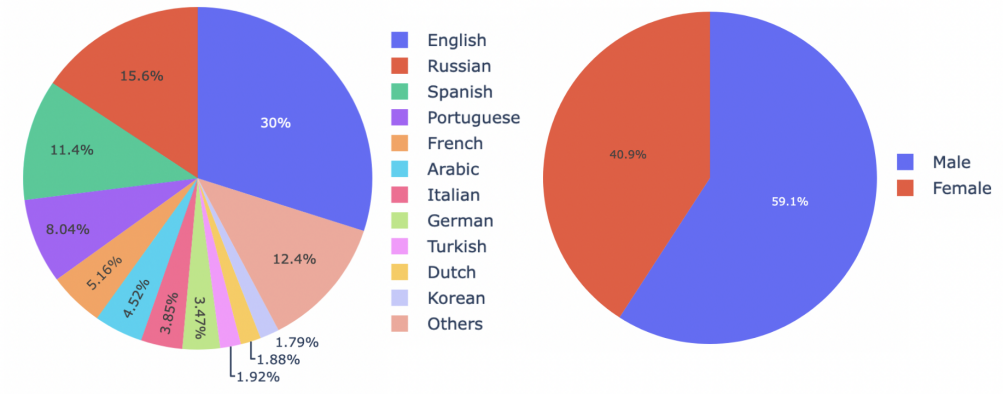

Figure 2: *VoxTube language and gender distributions*

Table 3: *Testing results for ResNet48 trained with VoxTube and VoxCeleb2*

Train data	**VoxCeleb1-test**		**LibriSpeech-test**		**SdSV21-dev**		**FFSVC20-dev**	
	EER[%]	$DCF_{0.01}$	EER[%]	$DCF_{0.01}$	EER[%]	$DCF_{0.01}$	EER[%]	$DCF_{0.01}$
VoxCeleb2	**1.36**	**0.128**	1.24	0.080	3.12	0.212	7.68	0.687
VoxTube	2.68	0.264	1.40	0.080	2.46	0.213	6.97	0.676
VoxCeleb2 + VoxTube	1.46	0.134	**1.17**	**0.059**	**2.39**	**0.201**	**6.38**	**0.607**

changed the length of each phase: the warmup phase was extended to 3 epochs, the plateau phase was extended to 10 epochs, and we dropped learning rate each 4 epochs during the decay phase.

4.4. Evaluation

We evaluated our models' performance in various languages and environments. As testing datasets, we used several text-independent datasets from recent competitions. We provide a brief description of each dataset below:

- **VoxCeleb1-test** [12]: This is a widely known speaker verification multilingual test set. We used a test subset consisting of 40 speakers.
- **LibriSpeech-test** [15]: Audiobooks-based dataset of English language speech. We used a *test-clean* subset with 40 speakers and 5.4 hours of speech.
- **SdSV21-dev Task 2** [24]: This dev set contains two languages - Persian and English. Enrollments are always utterances in Farsi, and the test utterance can be either in Farsi or English. Also, enrollment models have a large variance in the number of utterances: from 2 to 20. For evaluation we used only text-independent Task 2 development part.
- **FFSVC20-dev Track 2** [25]: This is a large dataset in Chinese Mandarin. It was recorded on several devices (phone and microphone arrays) located at different distances from 0.25 to 5 meters in parallel. For evaluation, we used text-independent Track 2 development part.

We used a cosine similarity backend for embeddings scoring. All test utterances shorter than 8 seconds were supplemented using the repetition of the source signal, and for the longer utterances we extracted multiple windows (without overlap) and then averaged embeddings within the utterance. We computed a cosine similarity score between a pair of enrollment and verification speaker models. In the case when there are more than one utterance in the speaker enrollment model, we firstly averaged embeddings within the enrollment model, before calculating the cosine similarity. Evaluation of systems' performance is done using the Equal Error Rate (EER) and a minimum Detection Cost Function with $P_{Target} = 0.01$.

5. Results

Our testing results are presented in Table 3. For each of three experiments, we utilized the same ResNet48 model architecture with different training data: VoxCeleb2 only, VoxTube only, and our third experiment included both datasets. As we can see, the model trained on VoxTube data provides better metrics for non-English language benchmarks compared to VoxCeleb2 training. We can also see that VoxTube domain is not highly correlated with a domain of VoxCeleb2 since the VoxCeleb1-test results are much better for the VoxCeleb2 training. In general, we can say that the VoxTube dataset provides high overall generalization ability across many testing domains. When VoxTube is combined together with the VoxCeleb2 dataset we get a 20-30% performance boost in most testing domains compared to VoxCeleb2 or VoxTube training only.

6. Conclusions

In this paper we introduced a new language-wide text-independent speaker recognition dataset VoxTube and its impact on various open-source benchmarks. We have demonstrated our underlying data collection and filtering approach that is based on audio segments clustering. We have shown that the VoxTube dataset complements well the existing and non-overlapping dataset VoxCeleb2 and can steadily improve the performance across the broad range of testing domains. We hope this new training dataset will be adopted, alongside the VoxCeleb2, as a baseline in the speech processing research community to train the models on.

7. References

[1] S. O. Sadjadi, C. Greenberg, E. Singer, D. Reynolds, L. Mason, and J. Hernandez-Cordero, "The 2019 NIST Speaker Recognition Evaluation CTS Challenge," in *Proc. The Speaker and Language Recognition Workshop (Odyssey 2020)*, 2020, pp. 266–272.

[2] M. McLaren, L. Ferrer, D. Castan, and A. Lawson, "The 2016 speakers in the wild speaker recognition evaluation." in *Interspeech*, 2016, pp. 823–827.

[3] M. K. Nandwana, J. Van Hout, M. McLaren, C. Richey, A. Lawson, and M. A. Barrios, "The voices from a distance challenge 2019 evaluation plan," *arXiv preprint arXiv:1902.10828*, 2019.

[4] H. Zeinali, K. A. Lee, J. Alam, and L. Burget, "Sdsv challenge 2020: Large-scale evaluation of short-duration speaker verification." in *INTERSPEECH*, 2020, pp. 731–735.

[5] J. S. Chung, A. Nagrani, E. Coto, W. Xie, M. McLaren, D. A. Reynolds, and A. Zisserman, "Voxsrc 2019: The first voxceleb speaker recognition challenge," *ISCA Challenges*, 2019.

[6] A. Nagrani, J. S. Chung, J. Huh, A. Brown, E. Coto, W. Xie, M. McLaren, D. A. Reynolds, and A. Zisserman, "Voxsrc 2020: The second voxceleb speaker recognition challenge," *arXiv preprint arXiv:2012.06867*, 2020.

[7] A. Brown, J. Huh, J. S. Chung, A. Nagrani, and A. Zisserman, "Voxsrc 2021: The third voxceleb speaker recognition challenge," *arXiv preprint arXiv:2201.04583*, 2022.

[8] J. Huh, A. Brown, J.-w. Jung, J. S. Chung, A. Nagrani, D. Garcia-Romero, and A. Zisserman, "Voxsrc 2022: The fourth voxceleb speaker recognition challenge," *arXiv preprint arXiv:2302.10248*, 2023.

[9] S. Chen, C. Wang, Z. Chen, Y. Wu, S. Liu, Z. Chen, J. Li, N. Kanda, T. Yoshioka, X. Xiao *et al.*, "Wavlm: Large-scale self-supervised pre-training for full stack speech processing," *IEEE Journal of Selected Topics in Signal Processing*, 2022.

[10] R. Makarov, N. Torgashov, A. Alenin, I. Yakovlev, and A. Okhotnikov, "Id r&d system description to voxceleb speaker recognition challenge 2022," 2022.

[11] A. Alenin, N. Torgashov, A. Okhotnikov, R. Makarov, and I. Yakovlev, "A subnetwork approach for spoofing aware speaker verification," 2022.

[12] A. Nagrani, J. S. Chung, and A. Zisserman, "Voxceleb: a large-scale speaker identification dataset," *arXiv preprint arXiv:1706.08612*, 2017.

[13] J. S. Chung, A. Nagrani, and A. Zisserman, "Voxceleb2: Deep speaker recognition," *arXiv preprint arXiv:1806.05622*, 2018.

[14] D. Garcia-Romero, G. Sell, and A. McCree, "Magneto: X-vector magnitude estimation network plus offset for improved speaker recognition," in *Proc. Odyssey 2020 The Speaker and Language Recognition Workshop*, 2020, pp. 1–8.

[15] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: An asr corpus based on public domain audio books," in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2015, pp. 5206–5210.

[16] R. Ardila, M. Branson, K. Davis, M. Henretty, M. Kohler, J. Meyer, R. Morais, L. Saunders, F. M. Tyers, and G. Weber, "Common voice: A massively-multilingual speech corpus," *arXiv preprint arXiv:1912.06670*, 2019.

[17] L. Buitinck, G. Louppe, M. Blondel, F. Pedregosa, A. Mueller, O. Grisel, V. Niculae, P. Prettenhofer, A. Gramfort, J. Grobler, R. Layton, J. VanderPlas, A. Joly, B. Holt, and G. Varoquaux, "API design for machine learning software: experiences from the scikit-learn project," in *ECML PKDD Workshop: Languages for Data Mining and Machine Learning*, 2013, pp. 108–122.

[18] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[19] D. Snyder, G. Chen, and D. Povey, "Musan: A music, speech, and noise corpus," *arXiv preprint arXiv:1510.08484*, 2015.

[20] J. Thiemann, N. Ito, and E. Vincent, "The diverse environments multi-channel acoustic noise database (demand): A database of multichannel environmental noise recordings," in *Proceedings of Meetings on Acoustics ICA2013*, vol. 19, no. 1. Acoustical Society of America, 2013, p. 035081.

[21] A. Mesaros, T. Heittola, and T. Virtanen, "A multi-device dataset for urban acoustic scene classification," in *Proceedings of the Detection and Classification of Acoustic Scenes and Events 2018 Workshop (DCASE2018)*, November 2018, pp. 9–13. [Online]. Available: https://arxiv.org/abs/1807.09840

[22] I. Szoke, M. Skacel, L. Mosner, J. Paliesek, and J. Cernocky, "Building and evaluation of a real room impulse response dataset," *IEEE Journal of Selected Topics in Signal Processing*, vol. 13, no. 4, p. 863–876, Aug 2019. [Online]. Available: http://dx.doi.org/10.1109/JSTSP.2019.2917582

[23] F. Wang, J. Cheng, W. Liu, and H. Liu, "Additive margin softmax for face verification," *IEEE Signal Processing Letters*, vol. 25, no. 7, pp. 926–930, 2018.

[24] H. Zeinali, K. A. Lee, J. Alam, and L. Burget, "Short-duration speaker verification (sdsv) challenge 2021: the challenge evaluation plan," 2019. [Online]. Available: https://arxiv.org/abs/1912.06311

[25] X. Qin, M. Li, H. Bu, R. K. Das, W. Rao, S. Narayanan, and H. Li, "The ffsvc 2020 evaluation plan," 2020. [Online]. Available: https://arxiv.org/abs/2002.00387