LEARNING VOICE REPRESENTATION USING KNOWLEDGE DISTILLATION FOR AUTOMATIC VOICE CASTING

Adrien Gresse, Mathias Quillot, Richard Dufour, Jean-François Bonastre

LIA, Avignon University, France

ABSTRACT

The search for professional voice-actors for audiovisual productions is a sensitive task, performed by the artistic directors (ADs). The ADs have a strong appetite for new talents/voices but cannot perform large scale auditions. Automatic tools able to suggest the most suited voices are of a great interest for audiovisual industry.

In previous works, we showed the existence of acoustic information allowing to mimic the AD's choices. However, the only available information is the ADs' choices from the already dubbed multimedia productions. In this paper, we propose a representation-learning based strategy to build a character/role representation, called p-vector. In addition, the large variability between audiovisual productions makes it difficult to have homogeneous training datasets. We overcome this difficulty by using knowledge distillation methods to take advantage of external datasets.

Experiments are conducted on video-game voice excerpts. Results show a significant improvement using the p-vector, compared to the speaker-based x-vector representation.

Index Terms— voice casting, knowledge distillation, p-vector, perceptual similarity, deep neural network

1. INTRODUCTION

In order to broadcast to the widest audience, audiovisual companies target the market on an international, multilingual and multicultural level. At the same time, audiovisual creation producers pay more and more attention to the voices they select for a particular character or role in order to reinforce the audience's sense of immersion. Voice dubbing is one of the most important solutions for audiovisual production localization and is capable of fostering this sense of immersion. Voice dubbing is about replacing the entire dialogs of the original creation by new voice-actors in the targeted language and cultural context. In this context, selecting the appropriate voices in a target language according to both the original voice and the role is a crucial task, referred as *voice casting*. Usually, a human expert called *artistic director* (AD) carries out the voice casting task in dubbing companies.

The major difficulty of voice dubbing lies in the fact that the "similarity" sought between an original voice and a dubbed voice is far from being a simple acoustical resemblance. It includes sociocultural characteristics of both source and target languages and countries. Moreover, there is no well-established vocabulary for describing voices, characters and immersive effects. There are two limitations to the way ADs perform the voice casting task: the ADs are embedding their own sociocultural characteristics in the casting task with the correlated subjectives biases, and 2), the ADs can't listen to and memorize a very large set of voices. As a result, an AD usually works with a short list of actors they have listened to and/or with whom they have already worked.

Automatic tools able to measure the potential adequacy between an original voice in a source language/cultural context and a dubbed voice in a target language and context are of a great interest for audiovisual industry. They will help the ADs to remedy the highlighted problems and to open the door for fresh voice talents, for example by preselecting a reasonable number of candidates inside a very large set of voices.

Perceptual voice similarity in the context of voice dubbing has been studied in [1, 2]. The authors show the importance of certain para-linguistic features (*e.g.* age, gender, speaker state, voice-quality). In [3], the authors propose to estimate the "dubbing" proximity of two voices (one in the source language and one in a target language) using a i-vector/PLDA based speaker-recognition approach. [4] makes the assumption that traces of the casting task performed by the ADs are present in the existing dubbed audiovisual creations. The proposed approach makes it possible to distinguish the *target* pairs (*i.e.* a voice in a source language associated with the corresponding character voice in the target language) from *nontarget* ones (*i.e.* voices corresponding to different characters). A limit of this work is that the use of binary-supervised learning gives poor generalization capacities to the model, considering that interpolation could only be based on counter-examples.

Recent works in speaker recognition [5, 6, 7, 8] showed that deep neural network embeddings and end-to-end learning can outperform i-vectors [9]. In this article, we propose to learn an original latent representation, called p-vector, from a character/role-based neural network approach. The p-vectors should help the system to have better assimilation of the character dimension and consequently to better handle unknown voices. It constitutes the first contribution of this paper.

Nevertheless, a brake on the use of such a neural network approach is the need for a large amount of in-domain data, which is critical for many tasks, including the one we are dealing with in this work. The only information that we can use for a supervised learning approach is the operator's past voice selection from existing dubbing. In addition, voices used in our previous works come from a small number of characters. In this paper, we propose to remedy this problem by applying knowledge distillation methods with the use of additional data, coming from a close domain, to extract the character/role specific information. More generally, we think the knowledge extracted, for example, from video-games could be transferred to other contexts, such as TV show characters.

This paper is organized as follows. We first present the approach and the generalized knowledge distillation framework in Section 2. Then we detail the corpus and we describe the experimental protocol we set-up in Section 3. We present our results and discuss them in Section 4. Finally, conclusions and perspectives are given in Section 5.

2. APPROACH

2.1. A character-based representation

In recent years, Deep Neural Networks have been proposed to learn task-oriented representational spaces allowing to disentangle the factors that explicate hidden data variability [10]. We propose to learn a dedicated representation called p-vector on professional acted voices. The p-vectors space (p stands for "personnage" in French) is optimized on a character/role discrimination task. It allows to project voice segments in a way that maximizes the character variability.

In general, input representation has a strong impact on the performance of machine learning applications. Here, we adopt the x-vector representation, originally introduced in automatic speaker recognition [8]. A large amount of data from many speakers are used to build the *speaker embeddings* space. Audio segments are projected into this space and characterized by x-vectors. It can be seen as a compact and fixed size representation of a variable length acoustic parameters vectors sequence. We make the hypothesis that the speaker embeddings contain entangled information corresponding to the character/role dimension. Hence, we propose to build a new representational space (p-vector) able to discriminate between the different characters.

2.2. Knowledge distillation

In the context of this work, we have to deal with relatively small number of data. We propose to use knowledge distillation in order to exploit additional data from a close domain to tackle this problem.

The generalized distillation framework [11] unifies two techniques that both introduce a teacher to guide a student model through its learning process. The first technique introduces the concept of *Privileged Information* [12] by adding a novel element $x_i^\star$ to the feature-label pair $(x_i,\, y_i)$ where $i \in [1...N]$, with N the number of samples. The second technique, referred as *Knowledge Distillation* [13], allows a simple neural network to solve a complicated task by distilling the knowledge from a cumbersome model. More generally, the teacher offers an opportunity for the student model to learn about decision boundary which are not contained in the training sample [11]. Typically, a neural network using a *softmax* activation function provides a probability for each class obtained with the following formula:

$$q_i = \frac{\exp\left(z_i/T\right)}{\sum_j \exp\left(z_j/T\right)}$$

where T refers to the temperature and z_i denotes the output computed for each class in the final layer. A higher value of T gives a softer probability distribution over all classes. Distillation consists in raising the temperature until the teacher model produces proper probabilities vector q_i that can be used to train a student model. The point is that the q_i coming from the teacher, also referred as soft-targets, contains much more information than a simple one-hot encoding.

As illustrated in Figure 1, we fit the student model to hard-targets (the one-hot encoded character labels) and soft-targets coming from the teacher. To do so, we use an imitation parameter denoted λ that controls the priority between the soft probabilities imitation and the usual hard labels predictions during the student model training. This is achievable by minimizing the following loss:

$$\mathcal{L} = \frac{1}{N}\sum_{i=1}^{N}\left[(1-\lambda)l(y_i,\, q_i) \,+\, \lambda l(s_i,\, q_i)\right]$$

where l denotes the *cross-entropy* loss and s_i refers to the soft-targets from the teacher model.

The teacher-student framework has been used in different works [14, 15, 16, 17, 18, 19] for a wide variety of tasks such as noise-robust speech recognition, domain adaptation, and speaker normalization. The proposed approach originally extends this framework to acted voices and specifically to character/role representation.

Given the limited number of character labels in our corpus, we train the teacher model on an additional dataset which contains more character labels. We suppose it could help the student model to learn a robust, more general, representation by fitting to the soft-targets from the teacher. Also, we suppose the student model could bypass hard-targets limitation by simply ignoring them to some extent.

3. EXPERIMENTAL PROTOCOL

3.1. Corpus

The voices from the *Mass Effect 3* role-playing game compose the main corpus. Originally released in English, this video-game has been translated and revoiced in other languages. In our experiments, we use the English and French versions of the audio sequences, representing 7.5 hours of speech in each language. Voice segments are 3 seconds long on average, each segment corresponds to a unique vocal interaction. Each English and French dataset contains $10,000$ voice segments. A character is then defined by a unique French-English couple of voice-actors. To avoid any bias in terms of speaker identity, we consider only a small subset of 31 different characters (13 female characters and 18 males), where we are certain that none of the actors plays more than one character.

To remedy the limited amount of characters in the *Mass Effect 3* corpus, we use additional data from another multilingual video-game called *Skyrim*. We limit this corpus to the English and French dialogues that are totalizing 120 hours of speech. We have $50,000$ segments in each language that are labelized according to 30 different character (7 females and 23 males). Since we do not have enough guarantee on the French-English correspondence of the segments and we are not sure that an actor plays a unique role, we do not use this corpus in the evaluation step. It only serves to transfer knowledge from the teacher to the student model in the distillation process. We make sure that there is no intersection between actors from *Skyrim* and *Mass-Effect 3* to prevent speaker-bias in the test set. Besides, all voice segments are high-quality studio-recorded audio files and we remove all segments shorter than one second.

3.2. Sequences extraction

We perform a usual acoustic parameterization of the audio signal that we transform into 60-dimensional feature sequences containing 20 MFCCs including the log of the energy plus the first- and second-order derivatives ($\Delta \,+\, \Delta\Delta$). We use a Hamming sliding window of 20ms with a 10ms overlap to compute the parameters. We perform a cepstral mean normalization and a voice activity detection to remove the low-energy frames that mainly correspond to silence. An x-vector extractor has been built with the Kaldi toolkit [20] and trained on the Voxceleb corpus [21].

3.3. Training protocol

The quantity of voice segments in the *Mass Effect 3* corpus is not well balanced among the different characters because of their relative importance in the video-game. Consequently, we select only

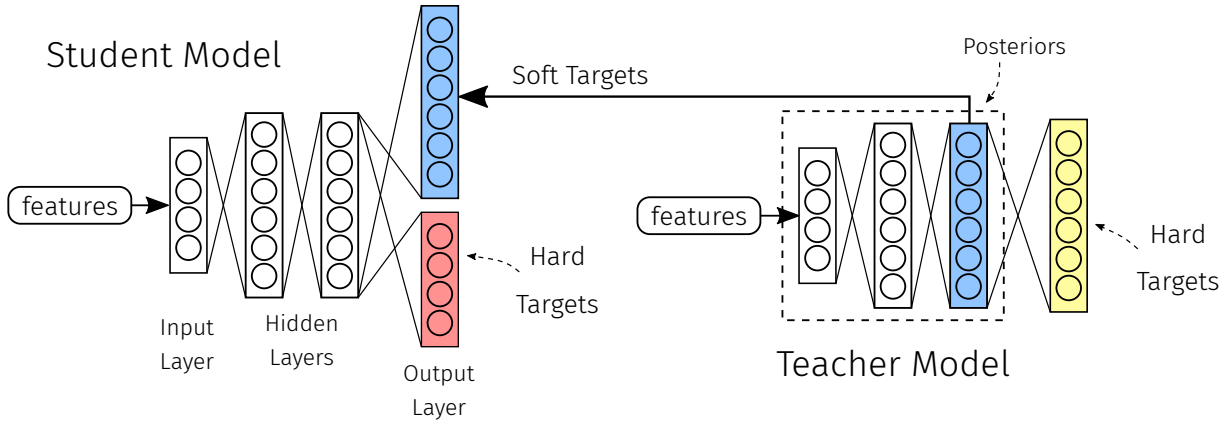

Fig. 1. The teacher model is trained to predict good soft-targets so that we can use them to train the student model. The teacher and student models can be trained either on the same or different corpus.

16 characters that all have at least 90 voice segments from both English and French voice-actors. Segments are all randomly picked out. Moreover, we create a k-fold cross-validation on this set of characters in order to have 4 of them in each fold. Thus, we have 4 distinct cases denoted A, B, C, and D that cover every character, each case involving 12 training characters and 4 characters kept-out for the evaluation. These 4 characters are completely unknown in the training part (they are not sharing any label or a speaker with one of the training data), making the voice-pairing task described in 3.4 extremely difficult. 20% of training data are used for validation. Regarding the additional corpus, we picked out the same number of segments for all the 30 characters and we also divided it into two parts with the same ratio assigned to validation. As we said before, no data from *Skyrim* are used for the test.

Both teacher and student models follow a similar neural network architecture. We create a Multi-Layer Perceptron (MLP) using the Keras toolkit [22]. We connect a 512-dimensional input layer to two hidden layers of dimension 256 plus an extra embedding layer (*i.e.* corresponding to p-vectors) of dimension 64 for the student model to a final output layer with a *softmax* activation function. Hidden layers are combined to a hyperbolic tangent activation. We apply a 0.25 dropout rate to hidden layers except for the embedding layer which is set to 0.5. We use a *Xavier* initialization of the model parameters [23] and we use the *Adadelta* optimizer with its default configuration to solve the minimization of the *cross-entropy* loss function. Moreover, we use a batch size of 12 examples and we train the model during 300 epochs while we monitor the loss function on the validation set to avoid overfitting.

We fit the teacher model to the features and labels from the additional dataset (*Skyrim*), considered as privileged information. The teacher model can be seen as a character/role discriminator. Then we use the teacher to compute the *Mass Effect 3* soft-targets and train the student model on both hard- and soft-targets from this corpus. The student learns to fit the 12 hard-targets and the 30 soft-targets depending on the λ parameter which controls the influence between soft- and hard-targets imitation during the training process. Finally, p-vectors are extracted from the student embedding layer.

3.4. Evaluation

To challenge the inherent quality of the learned representation, we first perform a clustering analysis using the k-means algorithm on the extracted p-vectors. We expressly set $k = 4$ to reflect the number of character labels present in the test set. Every voice segment then being gathered within the same cluster are assigned to the most represented character so that one cluster has one character label. Thus, a F-measure score is computed on the segment label hypothesis. Note that multiple clusters may be assigned to the same character, which constitutes a flaw. However, it remains a particular case indicating a bad result.

In addition, we evaluate the approach on a voice-pairing task with the *Mass Effect 3* corpus using the similarity scoring system proposed in [4]. Here, we test the capacity to make a significant distinction between *target* (*i.e.* same character) and *nontarget* (*i.e.* different character) pairs when we train the similarity model on p-vectors.

4. RESULTS

We use different values for the distillation temperature $T \in [1..5]$ and the best results are observed with $T = 4$ on average. In addition, we check the different values in the range $[0, 1]$ for the imitation parameter λ and we get the best results using $\lambda = 0.3$ when averaging on A, B, C, and D. To challenge the distillation method, we train a p-vector system with $T = 1$ and $\lambda = 0$ that corresponds to no distillation and an exclusive hard-target imitation (the additional corpus is not taken in account). Results presented in Tables 1 and 2 are on average under the proposed approach, which shows the benefice of the distillation.

4.1. Clustering analysis

	A	B	C	D
baseline (x-vector)	0.54	0.52	0.36	0.71
p-vector (no distillation)	0.66	0.72	**0.59**	0.66
p-vector + distillation	**0.78**	**0.78**	0.40	**0.77**

Table 1. F-scores obtained on test p-vectors.

Table 1 presents the results, in terms of F-measure, of the clustering analysis using p-vectors. We observe that p-vectors have much better F-scores than the baseline x-vectors, which is not

surprising since x-vectors are designed to focus on the vocal identities of both English and French voice-actors more than on their character/role. We observe relatively good results, up to 0.78 for the best case, which indicates the ability to automatically recognize unknown characters/roles with p-vectors. Regarding the case C, we hypothesize that the low F-scores may result from the inherent similarity of the characters –all are male soldiers– involved in this particular test. Surprisingly, the p-vector system without distillation performs better in this specific case.

In Figure 2, we illustrate a 2-dimensional projection of the p-vectors space thanks to the t-SNE algorithm. Unsurprisingly, we see a clear distinction between male and female characters in case A, B and D (C contains only male soldiers), same-gender characters are properly separated too. Considering the case D we observe that each voice-actors of both characters *Hackett* (blue) and *Illusive Man* (orange) have a strong vocal identity, which could facilitates clustering analysis and explains the unexpected high F-score (0.71) with the x-vector baseline system.

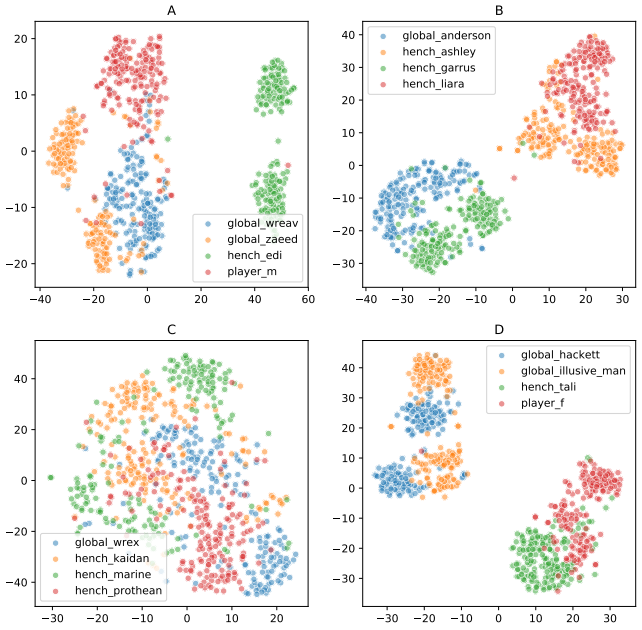

Fig. 2. Representation in the p-vectors space of the different voice-segments of each character in A, B, C and D.

4.2. Similarity task

In addition, we give the results when evaluating p-vectors with the similarity decision system in Table 2. The results are presented in terms of accuracy and Student's t-test. The statistical test confirms the significant difference between similarity scores of *target* and *nontarget* pairs since all of the associated p-values are under the rejecting threshold. On average, p-vector outperforms the x-vector baseline system on the similarity task, with a 57% mean accuracy and a 44.79 mean t-score over the four cases. Moreover, we see lower variations among the different test cases.

The hypothesis statistical test confirms that the observed differences between *target* and *nontarget* pairs are significant. It cannot be a mere coincidence since we apply strong constraints on the test set. Considering the difficulty of this task, we think they constitute a strong evidence that p-vectors contain character/role information.

		accuracy	t-score
baseline x-vector	A	0.60	64.58
	B	0.52	20.63
	C	0.54	26.86
	D	0.49	-6.19
	mean	0.54	26.47
p-vector (no distillation)	A	0.58	53.82
	B	0.54	20.70
	C	**0.57**	**49.86**
	D	0.54	23.34
	mean	0.55	36.93
p-vector + distillation	A	**0.63**	**80.00**
	B	**0.55**	**36.46**
	C	0.55	28.33
	D	**0.55**	**34.24**
	mean	0.57	44.79

Table 2. Performance of the voice-pairing task on test p-vectors. Validation accuracy is generally above 85%.

5. CONCLUSION

In this paper, we introduced a deep neural network embedding called p-vector for automatic voice casting. The proposed approach firstly projects a speech recording in a character/role discriminant neural network representational space. It uses knowledge distillation methods to overpass data limitation problems. We use p-vector representation to apply character-based similarity metrics. We propose a very constrained protocol to counterbalance the limited amount of evaluation data. We observe a substantial improvement using our neural network embedding over the x-vector baseline. These results demonstrate that p-vectors contain information dedicated to the character/role dimension. We achieved to differentiate the same- and different-character pairs given the results of the similarity metric. Also, we successfully retrieved characters from unknown voices with a satisfying F-measure performance.

However, due to the limitations of our database and despite the rigorous protocol we designed, some caution should be taken. Confirmation of our findings on a bigger database with more character-labels is needed before generalizing to every kind of audiovisual production, character or language/culture.

The teacher-student framework allows us to compute new soft-labels and it could be more effective on larger training datasets with numerous character labels and multiple actors per label. Moreover, p-vectors allow the initiation of new research on the explicability/explainability questions, in particular in the context of artistic directors choices. We wish to confront p-vectors to a simple binary decision to observe the potential impact of a particular feature on the character dimension. Future work will replace the similarity system that discriminates between same- and different-character pairs with explanatory features (e.g. gender, voice-quality, timbre, prosody).

6. ACKNOWLEDGMENT

This work is supported by the Digital Voice Design for the Creative Industry - TheVoice ANR project.

7. REFERENCES

[1] Nicolas Obin, Axel Roebel, and Grégoire Bachman, "On automatic voice casting for expressive speech: Speaker recognition vs. speech classification," in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2014.

[2] Nicolas Obin and Axel Roebel, "Similarity search of acted voices for automatic voice casting," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 24, pp. 1642–1651, 2016.

[3] Adrien Gresse, Mickael Rouvier, Richard Dufour, Vincent Labatut, and Jean-Francois Bonastre, "Acoustic pairing of original and dubbed voices in the context of video game localization," in *INTERSPEECH*, 2017.

[4] Adrien Gresse, Mathias Quillot, Richard Dufour, Vincent Labatut, and Jean-François Bonastre, "Similarity metric based on siamese neural networks for voice casting," in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019.

[5] Ehsan Variani, Xin Lei, Erik McDermott, Ignacio Lopez Moreno, and Javier Gonzalez-Dominguez, "Deep neural networks for small footprint text-dependent speaker verification," in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2014.

[6] David Snyder, Pegah Ghahremani, Daniel Povey, Daniel Garcia-Romero, Yishay Carmiel, and Sanjeev Khudanpur, "Deep neural network-based speaker embeddings for end-to-end speaker verification," in *Spoken Language Technology Workshop (SLT)*. IEEE, 2016.

[7] David Snyder, Daniel Garcia-Romero, Daniel Povey, and Sanjeev Khudanpur, "Deep neural network embeddings for text-independent speaker verification.," in *INTERSPEECH*, 2017.

[8] David Snyder, Daniel Garcia-Romero, Gregory Sell, Daniel Povey, and Sanjeev Khudanpur, "X-vectors: Robust dnn embeddings for speaker recognition," in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018.

[9] Najim Dehak, Patrick J Kenny, Réda Dehak, Pierre Dumouchel, and Pierre Ouellet, "Front-end factor analysis for speaker verification," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, pp. 788–798, 2011.

[10] Yoshua Bengio, Aaron Courville, and Pascal Vincent, "Representation learning: A review and new perspectives," *IEEE transactions on pattern analysis and machine intelligence*, vol. 35, no. 8, pp. 1798–1828, 2013.

[11] David Lopez-Paz, Léon Bottou, Bernhard Schölkopf, and Vladimir Vapnik, "Unifying distillation and privileged information," in *International Conference on Learning Representations*, 2016.

[12] Vladimir Vapnik and Rauf Izmailov, "Learning using privileged information: similarity control and knowledge transfer.," *Journal of machine learning research*, vol. 16, pp. 2023–2049, 2015.

[13] Geoffrey Hinton, Oriol Vinyals, and Jeffrey Dean, "Distilling the knowledge in a neural network," 2015.

[14] Ryan Price, Ken-ichi Iso, and Koichi Shinoda, "Wise teachers train better dnn acoustic models," *EURASIP Journal on Audio, Speech, and Music Processing*, 2016.

[15] Konstantin Markov and Tomoko Matsui, "Robust speech recognition using generalized distillation framework.," in *INTERSPEECH*, 2016.

[16] Jinyu Li, Michael L Seltzer, Xi Wang, Rui Zhao, and Yifan Gong, "Large-scale domain adaptation via teacher-student learning," 2017.

[17] Shinji Watanabe, Takaaki Hori, Jonathan Le Roux, and John R Hershey, "Student-teacher network learning with enhanced features," in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2017.

[18] Taichi Asami, Ryo Masumura, Yoshikazu Yamaguchi, Hirokazu Masataki, and Yushi Aono, "Domain adaptation of dnn acoustic models using knowledge distillation," in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2017.

[19] Neethu Mariam Joy, Sandeep Reddy Kothinti, S Umesh, and Basil Abraham, "Generalized distillation framework for speaker normalization," in *INTERSPEECH*, 2017.

[20] Daniel Povey, Arnab Ghoshal, Gilles Boulianne, Lukas Burget, Ondrej Glembek, et al., "The kaldi speech recognition toolkit," in *IEEE 2011 workshop on automatic speech recognition and understanding*, 2011.

[21] Joon Son Chung, Arsha Nagrani, and Andrew Zisserman, "Voxceleb2: Deep speaker recognition," in *INTERSPEECH*, 2018.

[22] François Chollet et al., "Keras," `https://keras.io`, 2015.

[23] Xavier Glorot and Yoshua Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, 2010.