Sample drop detection for asynchronous devices distributed in space

Tina Raissi
Human Language Technology and Pattern Recognition, RWTH Aachen University
Aachen, Germany
raissi@i6.informatik.rwth-aachen.de

Santiago Pascual*
Universitat Politècnica de Catalunya
Barcelona, Spain
santi.pascual@upc.edu

Maurizio Omologo
Center for Information and Communication Technology (ICT) Fondazione Bruno Kessler (FBK)
Trento, Italy
omologo@fbk.eu

***Abstract*—In many applications of multi-microphone multi-device processing, the synchronization among different input channels can be affected by the lack of a common clock and isolated drops of samples. In this work, we address the issue of sample drop detection in the context of a conversational speech scenario, recorded by a set of microphones distributed in space. The goal is to design a neural-based model that given a short window in the time domain, detects whether one or more devices have been subjected to a sample drop event. The candidate time windows are selected from a set of large time intervals, possibly including a sample drop, and by using a preprocessing step. The latter is based on the application of normalized cross-correlation between signals acquired by different devices. The architecture of the neural network relies on a CNN-LSTM encoder, followed by multi-head attention. The experiments are conducted using both artificial and real data. Our proposed approach obtained F1 score of 88% on an evaluation set extracted from the CHiME-5 corpus. A comparable performance was found in a larger set of experiments conducted on a set of multi-channel artificial scenes.**

***Index Terms*—Far-field speech recognition, conversational speech, microphone array synchronization, sample drop detection, CHiME-5 challenge**

I. Introduction

Distant-talking Automatic Speech Recognition (ASR) has attracted a considerable interest in the research community during the past years. There is a large number of applications related to this field which still require robustness to different sources of degradation of the input signal. Moreover, distant-ASR introduces many new complex challenges to solve, mainly due to the environmental acoustics and unpredictable noisy conditions, often represented by interfering speakers [1]. In order to improve the recognition performance, it is very common to adopt a multi-microphone multi-device setting which allows to observe the scene from different locations in space. For this purpose, the challenges and corpora such as CHiME, DIRHA, and AMI/AMIDA [2]–[4] are created to address a wide spectrum of research topics related to scenarios such as automatic transcription of speech in office and home environments, with spontaneous speech input, different noisy and reverberant conditions as well as different microphone and device placement configurations [5], [6]. In the specific case of the CHiME challenge, we are in the presence of simultaneous recordings of different real conversational scenarios from multiple microphone arrays, distributed in rather large spaces. The far-field recording sessions are done by using 6 Kinect devices. The signals are sample-synchronous within each device. However, the different devices can be subjected to asynchrony, due to both small clock speed variations and sample drop events [7]. The latter aspect causes misalignment between different signals, creating additional hurdles for tasks such as voice activity detection, beamforming, and any other multi-channel processing that would require synchrony at sample or frame level [8], [9]. It is also worth noting that such misalignment is additional to an intrinsic one that characterizes audio signals acquired by microphones largely spaced one to another, i.e., a shift due to the propagation time delays related to the geometry of the problem. This can strongly depend on the spatial location of each sound source, which is typically represented by both active speakers and coherent noise sources, and is continuously changing in experimental scenarios such as CHiME. In principle, clock drift, sample drop, and sound source localization could be jointly addressed with the aim of obtaining a rather accurate re-alignment between all signals [10]. Under controlled conditions, this joint goal can be addressed by using artificial multi-microphone audio datasets, where, differently to a real context, the fully ground-truth information is available.

Concerning the CHiME-6 challenge, which relies on the same data used for CHiME-5, the information regarding the time instant and the duration of each drop event, together with an array synchronization tool to mitigate the time misalignment problem, were made available very recently [11]. However, this ground-truth information derives from non-publicly distributed Audio/Video Interleaved (AVI) sequences, which were obtained by the organizers by means of additional devices during the data collection. Hence, they are affected by the clock rate mismatch between different devices. Based on these ground-truth instants and durations, it is possible to see that the overall shift among different devices due to the sample drop exceeds one second in many sessions. Moreover, the duration of a single drop can range from a few milliseconds to more than a half second, with an average value of 70

* Santiago Pascual is currently at Dolby Laboratories, Barcelona, Spain.

milliseconds. This large range represents the main challenge for the development of an automatic drop detection system.

The relevance of the sample drop detection in applications of multi-device multi-microphone processing is not restricted merely to the CHiME challenge. Nowadays, many ASR systems rely on multiple devices distributed in the environment, which transmit eventually the signals to the Cloud. Sometimes, these devices are low-cost and characterized by possible interrupts in their operational activity. The communication step itself can also be affected by packet loss, for instance in the case of limited internet access. According to the best of the authors knowledge, there is no similar prior work on sample drop detection in distributed microphone arrays for distant-ASR.

As discussed in the following, for the identification of the temporal intervals that are affected by a sample drop, one can apply standard cross-correlation method. In order to both accurately detect and eventually quantify the drop duration, the cross-correlation technique generally needs to process rather large context windows, which can range between 10 to 15 seconds. However, for the purpose of attaining a more efficient time accuracy, the analysis on a shorter context window is strongly required. The main focus of this paper is concerned with the sample drop detection within a short context window, which is tackled by adopting a neural classifier. The design of the architecture has been conceived to respond to a two-fold problem. First, the model has to learn a latent representation of the signal that allows for the comparison between the signal from a device affected by a sample drop event, and the signals coming from the other devices. Secondly, the system has to acquire robustness to the resulting time shift.

The paper is structured as follows. We first discuss the binary classification task, the cross-correlation based method and the neural solution in Sec. II. The description of the dataset, the conducted experiments and their results are reported in Sec. III, followed by our conclusions.

II. Classification Task and Methods

Let $x(t)$ be a source signal, and $y_{m,k}(t)$ the corresponding signal acquired at time sample t by the m-th microphone of the device k, with $m \in \{1, \cdots, M\}$ and $k \in \{1, \cdots, K\}$, where $M = 4$ and $K = 6$ in CHiME-5. Denote by $Y_{m,k}(\cdot)$ the Short Time Fourier Transform (STFT) of the acquired signal. Given a context window of length ℓ, within a time interval of length T, with $\ell \ll T$, our goal is to decide whether a device has been subjected to a sample drop event, as depicted in Fig. 1.

We denote by $\mathrm{C}(\cdot,\cdot)$ a function which correlates or compares $|Y_{m,hyp}(\cdot)|$ of a *hypothesis* device with the remaining *reference* devices $|Y_{m,k}(\cdot)|$, for all $k \neq hyp$. The binary classification task can be defined as follows:

$$\mathrm{Comb}_k \left(\mathcal{F} \left(\mathrm{C} \left(|Y_{m,hyp}(\cdot)| , |Y_{m,k}(\cdot)| \right) \right) \right) \rightarrow \{0, 1\}, \quad (1)$$

where $\mathcal{F}$ and Comb are a binary classifier and a combination strategy, e.g. averaging, respectively. Furthermore, we assume that the input to the function $\mathrm{C}(\cdot,\cdot)$ is the log-magnitude spectrum $20 \log_{10} |Y_{m,k}(\cdot)|$, in decibel.

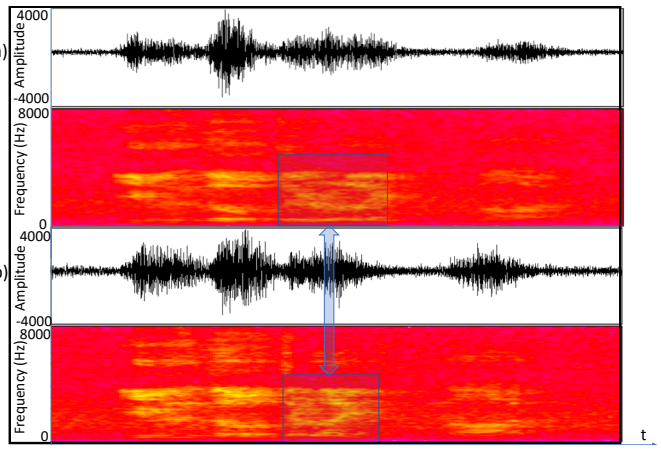

Fig. 1. Signals and related spectrograms referred to recordings acquired by two asynchronous devices for CHiME-5. The signal of case b) is affected by a sample drop, which is highlighted by a shorter vowel sequence in the middle of the sentence (see blue rectangles) and by a different spectrographic alignment after that time instant.

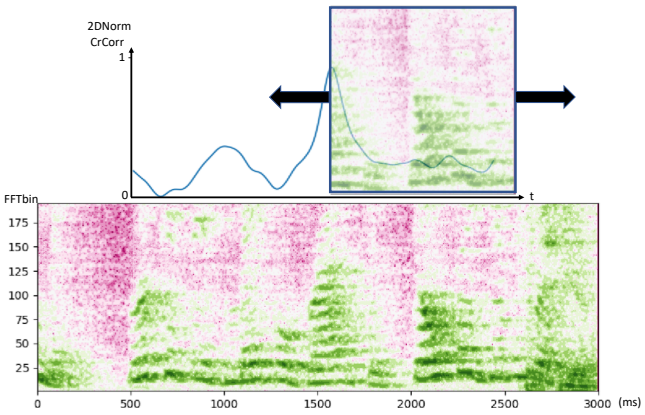

Fig. 2. Example of application of 2D normalized cross-correlation to spectrographic patterns extracted from two device microphones. For each time instant, a correlation value is computed between a different time-shifted pattern and a portion of the spectrogram (i.e., lower representation) representing the reference device signal. The resulting correlation function is reported in blue. The time location of its peak highlights the exact shift between the two signals.

A. Cross-correlation method

The normalized cross-correlation method adopted in this work derives from a standard approach used for pattern matching and feature detection in the image processing application field [12]–[14].

This method can be applied to microphone pairs referring to m-th microphone of the respective six devices, with $m = 1$ in our case. Given the log-magnitude spectrum $20 \log_{10} |Y_{1,k}(.)|$, each device signal is represented in terms of spectrogram, whose portions are then processed as images. The target is to find the best match shift between a pattern extracted from a device hyp, and a longer temporal sequence from a device ref, which is very likely covering the absolute time range of the former one, as shown in Fig. 2.

Due to the different propagation time delays from the

active sound sources to the microphones of each device, and minor drift effects caused by the clock mismatch, the two spectrographic representations are misaligned in time. Under noisy and reverberant conditions, they can differ substantially, especially when distant devices are involved. Nevertheless, a global similarity in terms of temporal relationships among most significant speech contents (e.g., formants and voice onsets) is often preserved. This similarity leads to a higher correlation value corresponding to the correct time shift. In the case of a sample drop that affects one sequence, this match drastically changes, on the grounds that an entire vertical slice of the spectrogram has been lost in one of the two channels. However, the cross-correlation function holds a peak at a time shift that differs from what was observed before the loss, by the exact duration of the loss. The resulting shift discontinuity represents a key aspect of the proposed method.

In order to increase the robustness of the method, we combine the analysis outputs obtained for each device by correlating the corresponding signal with each of the other $K-1$ device signals. In practice, a discontinuity corresponding to the same shift duration, that is observed for a device against each of the other ones, represents the cue for a highly probable sample drop. The described method is effective both to identify rather large intervals (10 seconds in this work), which are likely to include a possible drop, and to provide an estimate of its duration. However, in a preliminary study we observed some limitations when deriving a more accurate location for the drop with a time resolution of less than 2 to 3 seconds. For this reason, in this work we apply the normalized cross-correlation method as a preprocessing step. The intervention of the neural model is carried out over a short context window, once the former method has identified a set of candidate large time intervals.

B. Neural-based method

The cross-correlation method works upon raw features like the log-magnitude spectrum, whereas deep learning approaches build intermediate abstractions of the features throughout a stack of neural layers [15]. These abstractions are usually more robust to low-level feature changes, such as temporal shifts, and also facilitate a learnable preprocessing to solve a task of a high-abstraction level like classification. Hence we leverage the capacity of neural networks to process two separate log-magnitude spectrum sources, the hypothesis and the reference, and to classify whether there is a sample loss in the hypothesis. To that end, we design a network which processes the spectral inputs through two distinct branches, and relates them subsequently with a multi-head attention component. The proposed model is depicted in Fig. 3. Given the hypothesis input and the reference input, both of length ℓ, we inject them into the two encoder branches. Each input branch is composed by a convolutional neural network (CNN) front-end, specially suitable to detect local temporal correlations in the spectral frames, and a long-short term memory (LSTM) [16] block that exploits specifically the long-term sequential dependencies of the input sequence. The

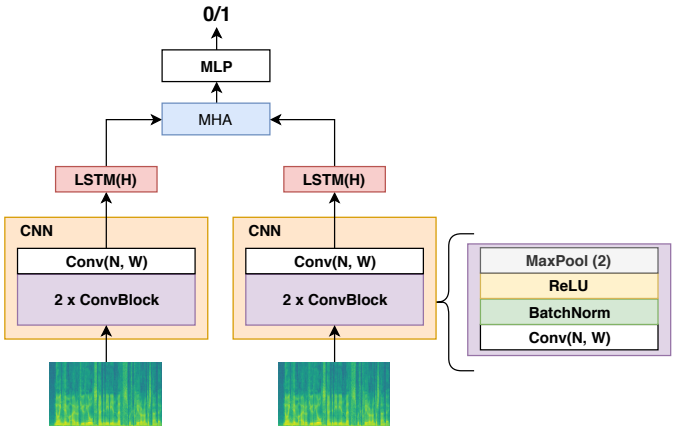

Fig. 3. Architecture of the proposed siamese network. MHA: multi-head attention. MLP: multi-layer perceptron. CNN: convolutional neural network. Conv(N, W) refers to a one-dimensional convolutional neural layer with N kernels and W kernel length. For the LSTM block, (H) refers to having H cells in the LSTM layer.

convolutional front-end is characterized by a couple of max-pooling layers that halve temporarily the feature maps after each convolutional block. This type of model is known as the siamese network [17], where both branches are constrained to share the same weights as they process the same type of input data. The optimization problem in the siamese network then leads to learning two separate embeddings which have positive inner products for sequences of the same class but negative inner products for those of different classes [18]. Once we obtained the two hidden representations of the signals, we forward them through the multi-head attention (MHA) component, which outputs a weighted sum of the reference computed by a compatibility function of the hypothesis to itself [19]. As a final step, we took the last time step values and forwarded it through a multi-layer perceptron (MLP), right before the application of a sigmoid function σ.

During training, the optimization procedure is not taking the multi-device scenario into account. The learning problem is reduced to the detection of the sample drop between pairs of hypotheses and references. Starting with Eq. (1) we define the function C to be a composition between a multi-head attention block and a linear layer. By setting $\mathcal{F} = \sigma$, we denote the detection of a sample drop event $\mathcal{D}$ as:

$$p(\mathcal{D}|h_{hyp}, h_{ref}) = \sigma\left(\text{MLP}\left(\text{MHA}\left(h_{hyp}, h_{ref}\right)\right)\right), \quad (2)$$

where h_{hyp} and h_{ref} are the embeddings extracted by the CNN-LSTM block. During test, for each device j we forwarded five segment pairs using all devices $k \neq j$. The distinct output values of the network are then combined together by taking the average or median value, and leading to

$$p(\mathcal{D}|h_j, h_k) = \text{Avg}_k\left(\sigma\left(\text{MLP}\left(\text{MHA}\left(h_j, h_k\right)\right)\right)\right). \quad (3)$$

III. Experiments

The CHiME-5 corpus does not include a set of data and related ground-truth for training and test of a sample drop detection system. Therefore, the overall training of the network has been carried out by using artificial data, and in two distinct stages. Concerning the evaluation of the performance of the model, we did not limit us to the mere usage of artificial data, and used also an evaluation set extracted from the real data.

A. Data

1) Noisy-reverberated LibriSpeech: The first training round of the neural model was conducted using two different noisy and reverberated datasets, both derived from the LibriSpeech train-clean-100 corpus [20], containing 28539 speech utterances. Each utterance was filtered by using a different room Impulse Response (IR), characterized by a reverberation time typically ranging between 0.4 and 0.8 seconds. IRs were computed by using a modified version of the image method [21], [22]. Moreover, the two datasets differ both due to a different environmental noise, and to a different sound propagation time shift. Overall, SNRs are generally in the range between 5 and 25 dB. Finally, a random sample loss event was simulated on one of the two versions, and from each of them segment pairs of hypothesis and reference were extracted.

2) Artificial multi-device mini-scenes: The artificial dataset incorporates 1182 mini-scenes that were generated using clean Librispeech signals, and an accurate and realistic simulation of multi-microphone acquisition in a noisy and reverberant environment, as described in [22]. Each scene refers to a different distribution of the six Kinect devices in space, with varying room size, location of speakers, and noise sources, as well as directivity of source, microphone polar pattern, and reflection coefficient of each wall. The mini-scenes were created without any sample drops. Each scene was then postprocessed to simulate the introduction of none, one, or more drops, for a total number of 880 drops. The data set was split into 782, 100 and 300 mini-scenes for drop detection training, development and evaluation, respectively. The scene duration ranges from 5 to 30 seconds. This material has then been organized to expose the neural model to a balanced set of drop/no-drop examples.

3) Real dataset: In order to evaluate the proposed model in a more realistic context, it is necessary to test the performance also on a real dataset. In this regard, it is worth noting that there exists no real corpus for studies on sample drop detection. The evaluation set was created by extracting 65 segments, from which 31 with sample drop, from three sessions of the train portion of the CHiME-5 corpus, namely sessions 03, 07, and 08. Some of these segments are characterized by a loss that was found through a very careful visual inspection of the related spectrograms, as outlined in Sec. II-A, and thus further improving the accuracy of the automatically derived boundaries provided by CHiME-6 organizers. As reported in the following, this set of real drops was processed by both using the cross-correlation method and eventually applying the proposed neural model.

B. Setting

The duration of the loss for the two different contaminated versions of LibriSpeech, was drawn from a left-truncated normal distribution $\mathcal{N}(600, 150)$, with a cut value of 50. The samples belonging to that interval were then eliminated from one of the segments. A one second context window containing this loss was isolated, by positioning the loss point randomly within the context window. The parameters of the pre-trained model on this data were then used for a further training step on the mini-scenes. The performance of the final model was tested on two separate evaluation sets belonging to both artificial and real data. The labels of the one-second-long segment pairs are distributed equally in both training and evaluation sets. Regarding the network's architecture, two different experiments were carried out. First, we concatenated h_{hyp} and h_{ref} defined in Eq. (2) and applied the sigmoid function on the last time step. Then we added the attention block and operated in the same way for the binary output. We also experimented with the average activations over the time values, but it proved to be less effective. We trained the models for 20 epochs on minibatches of size 50 and 30 for the pre-trained and final models respectively. We used the Adam optimizer [23] with the default PyTorch [24] parameters and learning rate $5 \cdot 10^{-5}$. Regarding the network structure, fully connected and convolutional layers comprise 512 units, with kernel length $N = 5$ in the case of convolutions. The LSTM layer contains 1024 cells. The model structure hence amounts to approximately 15 M of parameters in total.

C. Results

The cross-correlation method is mainly proposed here as a first step of a two-step procedure, which includes the neural model in the second step. In order to show the effectiveness of this approach we report a preliminary result by applying only the cross-correlation method for the detection.

Experiments on the real drop dataset extracted from CHiME-5 show that in this case the detection performance is represented by an F1 score equal to 80%. Comparable results can be obtained by testing it on an artificial dataset. Concerning the proposed two-step detection approach, the results reported in Table I on a subset of CHiME-5 data show that thanks to the introduction of the neural model it is possible to obtain an improvement of 8.5% of F1 score. Moreover, the results confirm the importance of having a two-stage training procedure, since the pre-trained model on the contaminated LibriSpeech or the model trained directly on the mini-scenes without a pre-training phase have both a poor performance. Another important factor is concerned with the effect of the attention mechanism. The difference of F1 score of the final model over mini-scenes is much smaller than the one over CHiME-5, in case of application of multi-head attention. Furthermore, the big gap between the result of the pre-trained model and others indicates the importance of having an appropriate data for multi-device case.

During test, we combined the output of the network for different pair devices by using major voting, averaging and

the median value. The best choice resulted to be averaging and median for mini-scenes and CHiME-5, respectively.

TABLE I
PRECISION (P), RECALL (R) AND F1 SCORE [%] FOR THE NEURAL MODEL PRE-TRAINED ONLY ON CONTAMINATED LIBRISPEECH (PRE-NN), AND THE FINAL NEURAL MODEL (FINAL-NN) WITH AND WITHOUT ATTENTION AND PRE-TRAINING STEP, ON ARTIFICIAL MULTI-DEVICE MINI-SCENES AND CHIME-5 CORPUS.

Model	Pre-Trained	Attention	Mini-scenes			CHiME-5		
			P	R	F1	P	R	F1
Pre-NN	n/a	yes	57.1	64.5	60.6	45.6	67.7	54.5
Final-NN	no	yes	48.6	81.6	60.9	48.0	93.0	63.3
	yes	no	77.8	86.7	82.0	50.0	74.1	59.7
		yes	87.5	88.6	**88.0**	90.0	87.0	**88.5**

IV. CONCLUSIONS

In this paper, we investigated a possible approach for the detection of sample drops in the context of multi-microphone devices distributed in space, a very challenging technical issue recently emerged with reference to the CHiME-5 challenge.

The proposed approach consists in combining a normalized cross-correlation processing and a neural classifier, in order to detect short context windows that are characterized by a possible loss. Experimental results show that a classification performance of 88.5% F1 score is obtained both on simulated and on real evaluation datasets. This result shows the advantage of the proposed two-step procedure, over just using cross-correlation for detection, which would provide 8.5% F1 score lower performance, as reported above. In the near future, we plan to improve this performance along different research directions, such as cross-processing all the microphone signals acquired by the available devices.

Though our current main focus is on the CHiME-5 data set, the proposed solution can be applied to other similar contexts, which are affected by loss of segments in the audio input signals. Our work also represents a preliminary step towards a joint combination of sample drop detection and quantification of the duration of the lost segment, which is a research issue under study. Moreover, we envisage a third processing step that includes a possible reconstruction of the lost information, based on exploiting redundant information available from higher-quality input channels not affected by the sample drop. For all of these foreseen directions, we also plan to verify soon a possible improvement in terms of recognition performance on the CHiME-5 task, (and on the very recently launched CHiME-6) provided by the application of the proposed method and of the resulting array synchronization.

V. ACKNOWLEDGEMENTS

The work reported here was started at JSALT 2019, and supported by JHU with gifts from Amazon, Facebook, Google, and Microsoft. This work was also supported by the project TEC2015-69266-P (MINECO/FEDER, UE). We thank Jon Barker for providing us with a preliminary list of temporal instants related to possible sample drops in CHiME-5, and Tobias Menne and Ralf Schlüter for their valuable feedback and suggestions.

REFERENCES

[1] M. Wlfel and J. W. McDonough, *Distant Speech Recognition*. John Wiley and Sons, 2009.

[2] J. Barker, S. Watanabe, E. Vincent, and J. Trmal, "The fifth chime speech separation and recognition challenge: Dataset, task and baselines," *arXiv:1803.10609*, 2018.

[3] M. Ravanelli, L. Cristoforetti, R. Gretter, M. Pellin, A. Sosi, and M. Omologo, "The dirha-english corpus and related tasks for distant-speech recognition in domestic environments," in *IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, 2015, pp. 275–282.

[4] J. Carletta *et al.*, "The ami meeting corpus: A pre-announcement," in *International Workshop on Machine Learning for Multimodal Interaction*. Springer, 2005, pp. 28–39.

[5] S. Watanabe, M. Delcroix, F. Metze, and J. R. Hershey, *New era for robust speech recognition: exploiting deep learning*. Springer, 2017.

[6] J. Li, L. Deng, R. Haeb-Umbach, and Y. Gong, *Robust automatic speech recognition: a bridge to practical applications*. Academic Press, 2015.

[7] A. Plinge, F. Jacob, R. Haeb-Umbach, and G. A. Fink, "Acoustic microphone geometry calibration: An overview and experimental evaluation of state-of-the-art algorithms," *IEEE Signal Processing Magazine*, vol. 33, no. 4, pp. 14–29, 2016.

[8] I. Medennikov *et al.*, "The STC system for the CHiME 2018 challenge," in *Proc. CHiME-5 Workshop*, 2018.

[9] E. Vincent , T. Virtanen, and S. Gannot, *Audio Source Separation and Speech Enhancement*. John Wiley and Sons Ltd, 2018.

[10] M. Cobos, F. Antonacci, A. Alexandridis, A. Mouchtaris, and B. Lee, "A survey of sound source localization methods in wireless acoustic sensor networks," *Wireless Communications and Mobile Computing*, no. 3956282, pp. 1–24, 2017.

[11] S. Watanabe *et al.*, "CHiME-6 Challenge: Tackling multispeaker speech recognition for unsegmented recordings," in *6th International Workshop on Speech Processing in Everyday Environments (CHiME 2020)*, Barcelona, Spain, May 2020. [Online]. Available: https://hal.inria.fr/hal-02546993

[12] R. C. Gonzalez and R. E. Woods, *Digital image processing - 3rd edition*. Prentice Hall, 2008.

[13] R. Brunelli, *Template Matching Techniques in Computer Vision: Theory and Practice*. Wiley Publishing, 2009.

[14] J. Lewis, "Fast normalized cross-correlation," *Vision Interface*, pp. 120–123, 1995.

[15] I. Goodfellow, Y. Bengio, and A. Courville, *Deep learning*. MIT press, 2016.

[16] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[17] J. Bromley, I. Guyon, Y. LeCun, E. Säckinger, and R. Shah, "Signature verification using a" siamese" time delay neural network," in *Advances in neural information processing systems*, 1994, pp. 737–744.

[18] G. Kim, A. Okuno, K. Fukui, and H. Shimodaira, "Representation learning with weighted inner product for universal approximation of general similarities," *arXiv:1902.10409*, 2019.

[19] A. Vaswani *et al.*, "Attention is all you need," in *Advances in Neural Information Processing Systems (NIPS)*, 2017, pp. 5998–6008.

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

[21] J. Allen and D. Berkley, "Image method for efficiently simulating smallroom acoustics," in *J. Acoust. Soc. Am*, 1979, pp. 2425–2428.

[22] M. Ravanelli, P. Svaizer, and M. Omologo, "Realistic multi-microphone data simulation for distant speech recognition," in *Proceedings of Interspeech*, 2016, pp. 2786–2790.

[23] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv:1412.6980*, 2014.

[24] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer, "Automatic differentiation in PyTorch," in *Neural Information Processing Systems Workshop on The Future of Gradient-based Machine Learning Software & Techniques (NIPS-Autodiff)*, 2017.