DNN-HMM based Automatic Speech Recognition for HRI Scenarios

José Novoa

Speech Processing and Transmission
Laboratory, University of Chile
Av. Tupper 2007, Santiago, Chile
jose.novoa@ing.uchile.cl

Jorge Wuth

Speech Processing and Transmission
Laboratory, University of Chile
Av. Tupper 2007, Santiago, Chile
jwuth@ing.uchile.cl

Juan Pablo Escudero

Speech Processing and Transmission
Laboratory, University of Chile
Av. Tupper 2007, Santiago, Chile
jescudero@ing.uchile.cl

Josué Fredes

Speech Processing and Transmission
Laboratory, University of Chile
Av. Tupper 2007, Santiago, Chile
jfredes@ing.uchile.cl

Rodrigo Mahu

Speech Processing and Transmission
Laboratory, University of Chile
Av. Tupper 2007, Santiago, Chile
rmahu@ing.uchile.cl

Néstor Becerra Yoma

Speech Processing and Transmission
Laboratory, University of Chile
Av. Tupper 2007, Santiago, Chile
nbecerra@ing.uchile.cl

ABSTRACT

In this paper, we propose to replace the classical black box integration of automatic speech recognition technology in HRI applications with the incorporation of the HRI environment representation and modeling, and the robot and user states and contexts. Accordingly, this paper focuses on the environment representation and modeling by training a deep neural network-hidden Markov model based automatic speech recognition engine combining clean utterances with the acoustic-channel responses and noise that were obtained from an HRI testbed built with a PR2 mobile manipulation robot. This method avoids recording a training database in all the possible acoustic environments given an HRI scenario. Moreover, different speech recognition testing conditions were produced by recording two types of acoustics sources, i.e. a loudspeaker and human speakers, using a Microsoft Kinect mounted on top of the PR2 robot, while performing head rotations and movements towards and away from the fixed sources. In this generic HRI scenario, the resulting automatic speech recognition engine provided a word error rate that is at least 26% and 38% lower than publicly available speech recognition APIs with the playback (i.e. loudspeaker) and human testing databases, respectively, with a limited amount of training data.

ACM Reference Format:

J. Novoa, J. Wuth, J. P. Escudero, J. Fredes, R. Mahu and N. B. Yoma. 2018. DNN-HMM based Automatic Speech Recognition for HRI Scenarios. In *HRI '18: 2018 ACM/IEEE International Conference on Human-Robot Interaction*, March 5–8, 2018, Chicago, IL, USA. ACM, NY, NY, USA, 10 pages. <https://doi.org/10.1145/3171221.3171280>

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).

HRI '18, March 5-8, 2018, Chicago, IL, USA

© 2018 Copyright is held by the owner/author(s).

ACM ISBN 978-1-4503-4953-6/18/03.

<https://doi.org/10.1145/3171221.3171280>

KEYWORDS

DNN-HMM, time-varying acoustic channel, speech recognition.

1 INTRODUCTION

If social robotics is a reality, then the appropriate social integration between humans and robots could greatly improve the cooperation between users and machines. There are several applications in defense, hostile environments, mining, industry, forestry, education and natural disasters where some integration and collaboration between humans and robots will be required [1]. Human Robot Interaction (HRI) is especially relevant in those situations when robots are not fully autonomous and require interaction with humans to receive instructions or information in decision-making applications [2–5]. In this context, human like communication between people and robots is essential for a successful human-robot collaborative symbiosis [6,7]. Additionally, speech is the most straightforward and natural way that humans employ to communicate [8–10]. As a consequence, voice-based HRI should be the most natural way to facilitate a collaborative human-robot synergy. Hence, speech technology, especially automatic speech recognition (ASR), should play an important role in social robotics.

Furthermore, it is well known that computer vision is an important research topic in robotics. Recent challenges such as DARPA Robotics Challenge [11] and Robocup [12] have led to great improvements in computer vision [13–16]. On the other hand, there has also been a significant progress in ASR, but this advancement has taken place outside the HRI field. ASR has gained relevance in robotics in the last years, but its status is still far from the one enjoyed by computer vision in the robotic research. This is still somehow surprising, considering that both technologies make use of similar signal processing and deep learning methods, and may explain partly the lower penetration of ASR in the robotic community.

In this paper, we propose that ASR technology should also be investigated, designed and developed to address HRI

applications. Subsequently, the ASR engine should take into consideration the environment, and robot and user states and contexts. Following this strategy, this paper focuses on the environment representation and modeling by training our ASR engine with the combination of clean utterances with the acoustic-channel responses and noise that were estimated and recorded, respectively, with an HRI testbed. This testbed represents the generic problem of HRI in mobile robotics and the resulting ASR accuracy outperforms publicly available ASR APIs with a limited amount of training data.

2 RELATED WORK

2.1 An introduction to ASR technology

Automatic speech recognition (ASR) is the process and related technology for transcribing human speech into words. By using Bayes's rule, the ASR problem can be formulated as follows [17]:

$$\hat{W} = \underset{W}{\operatorname{argmax}} P(W|X) = \underset{W}{\operatorname{argmax}} P(X|W) \cdot P(W) \quad (1)$$

where $\hat{W}$ is the optimal label (word or phone) sequence; X is the input speech observation sequence that represents a given speech utterance; $P(W)$ denotes the language model describing the probabilities of word combinations; and, $P(X|W)$ indicates the acoustic model. Consequently, the task of an ASR system is to find (by means of a process called decoding, performed with the Viterbi algorithm [18]) the most likely label sequence $\hat{W}$ given an observed sequence of feature vectors that corresponds to the speech utterance. The language model can be represented with [19]: statistical models; stochastic context-free grammars (SCFG); or, stochastic finite-state models. In the case of statistical models, which are widely employed in research, the prior probability of a word sequence $W = w_1, \dots, w_L$ in (1) can be approximated with N-grams:

$$p(W) \cong \prod_{l=1}^L p(w_l | w_{l-1}, w_{l-2}, \dots, w_{l-N+1}) \quad (2)$$

where N is typically between 2 and 4. The language model defines the transition probability from one N-gram to the next word to guide the search for an interpretation of the acoustic input. Additionally, the size of the vocabulary and perplexity [20] are critical for the ASR accuracy. Basically, perplexity measures the uncertainty about the words that may follow a given N-gram. A low-perplexity language model defined by a given task or context will constrain the decoding and perform better than a high-perplexity one.

Acoustic modeling defines the statistical representations for the sequence of acoustic feature vectors X obtained from the speech waveform. The utterances are divided into 20 or 30 ms windows with overlap (e.g. 50%). The set of acoustic features are usually obtained from the short-term fast Fourier transform (FFT) within each window [18,21,22]. Speed and acceleration coefficients (also called delta and delta-delta coefficients) are also typically used, and the final feature vector is composed of the static features plus the delta and delta-delta coefficients [23]. Mean and variance normalization of the coefficients can also be

employed. Until a few years ago, most speech recognition systems adopted hidden Markov models (HMMs), to deal with the temporal variability of speech, and Gaussian mixture models (GMMs) to represent $P(X|W)$. Given a set of speech feature vectors $\mathbf{X} = \{\mathbf{x}_t\}_{t=1}^T$, the state observation probability density function of feature vector $\mathbf{x}_t$ at frame t in state s_i is expressed by [18]:

$$p(\mathbf{x}_t | s_i) = \sum_{m=1}^M c_{i,m} \cdot \mathcal{N}(\mathbf{x}_t; \mu_{i,m}, \Sigma_{i,m}) \quad (3)$$

where $c_{i,m}$, $\mu_{i,m}$, and $\Sigma_{i,m}$ correspond to the mixture weights, mean vectors, and covariance matrices, respectively, for M Gaussian mixture components. In the last few years, artificial neural networks (ANNs), e.g. deep neural networks (DNNs), have shown significant performance improvement over GMM based models. In a DNN-HMM system, the DNN provides a pseudo-log-likelihood defined as:

$$\log[p(\mathbf{x}_t | s_j)] = \log[p(s_j | \mathbf{x}_t)] - \log[p(s_j)] \quad (4)$$

where s_j denotes one of the states or senones; and the state priors $\log[p(s_j)]$ can be trained using the state alignments obtained with the training speech data. The final decoded word string, $\hat{W}$, is determined by:

$$\hat{W} = \underset{W}{\operatorname{argmax}} \{\log[P(X|W)] + \lambda \cdot \log[P(W)]\} \quad (5)$$

where the acoustic model probability $P(X|W)$ depends on the pseudo log-likelihood $\log[p(\mathbf{x}_t | s)]$ delivered by the DNN, and λ is the constant that is employed to balance the acoustic model and language model scores [24]. The results reported in [25] showed that the DNN-HMM ASR can lead to a word error rate reduction of 32% relative when compared to the ordinary GMM-HMM system with the Switchboard task [26]. However, training a DNN is not an easy task. The objective function can be highly non-convex and the training algorithm can easily converge to a suboptimal local minimum. This problem can be minimized by making use of a pre-training strategy [27]. Also, ANNs need more training data than GMM-HMM systems [28]. It is worth mentioning that public ANN based ASR APIs employ at least tens of thousands of hours of speech for training, if not millions of hours. Other ANN architectures have also been applied to ASR: LSTM [29]; CNN [30]; and, RNN [31]. The results obtained using DNN-HMM systems are competitive when compared to those reported with others ANN architectures [32–36]. In some cases, systems employing combinations of ANN architectures, very deep CNN [37] or fCNN [38] have outperformed DNN, LSTM, or the ordinary CNN approaches. However, the higher the number of the ANN parameters, the higher the required amount of training data.

In matched conditions between training and testing data, ASR shows large performance gain. In contrast, models will have difficulties recognizing test samples if they differ from data used in training. For this reason, noise robustness of ANN based systems can be achieved by using multi-style training. For instance, a DNN trained with several types of noise and SNR

levels can lead to a high accuracy improvement in real applications [39].

2.2 Black box based integration of ASR technology

Most of the research that considers ASR in HRI scenarios use ASR toolkits or APIs as black boxes. A non-exhaustive list of available options that support ASR includes systems such as HTK [40], SPHINX [41,42], JULIUS [43], KALDI [44] and BAVIECA [45], and general purpose ASR APIs provided by, for instance, Google, Microsoft and IBM. These toolkits and APIs have been employed in HRI applications to incorporate ASR capabilities to a robot on a plug-and-play fashion [46–50], i.e. a speech signal is input to the ASR to obtain a text transcription (see Fig. 1) without taking into consideration operation conditions such as noise, relative movement between the speaker and the robot, microphones directivity and response, or user or robot context.

Figure 1: Ordinary black box based ASR integration in HRI scenarios.

In [46], a project that integrates smart home technology and a socially assistive robot to extend independent living for elderly people is described. A Nao robot plays the role of communication interface between the elderly, the smart home, and the external world. The robot can recognize simple answers from the user such as “yes” and “no” by using Sphinx 4.0 from Carnegie-Mellon University. Despite the fact that the Nao robot has a built-in microphone, its quality is too low for practical indoor applications, and a ceiling-mounted microphone was used to capture user speech. CMU Sphinx engine was also employed in [49], as part of a voice control system for a robotic endoscope holder during minimally invasive surgery. In [48], a general framework for multimodal human-robot communication is proposed. This framework allows users to interact with robots using speech and gestures. The Google Speech API was chosen because it offered speaker and vocabulary independency, which in turn could allow a natural speech interaction with no constraints. Google Speech API was also employed in [50] to provide ASR capabilities to a robot that needed to understand the intentions of users without requiring specialized user training. It comprises a recognition model that combines language, gestures, and visual attributes. In [51], four ASR engines were compared by making use of different grammars: the Google Speech API; the Microsoft Speech APIM; Pocket Sphinx from CMU; and, the NAO-embedded Nuance VoCon 4.7 engine. Experimental results showed that the Google Speech API led to the highest accuracy.

The integration of ASR technology on a black box basis can lead to poor performance because the chosen ASR system is not

designed necessarily to comply with specific scenarios or tasks. In [47], an evaluation with children aged from 4 to 10 years old playing versions of a language-based game hosted by an animated character is described. Speech recognition results using Sphinx3 on children utterance showed a poor performance, partially due to the mismatch between the children’s voices and the adult acoustic model of the ASR engine. General purpose speech toolkits or APIs have been widely used as an easy solution to integrate ASR to some platforms. However, while those ASR engines provide good results in several scenarios, they may not provide an optimal solution to specific tasks because they are not considered in the training procedure, or the technology simply does not compensate for unexpected distortions. As an example, in [52], it was investigated whether the open-source speech recognizer Sphinx can be tuned to outperform Google cloud-based speech recognition API in a spoken dialog system task. By training a domain-specific language and making adjustments, Sphinx could outperform the Google API by 3.3%.

2.3 Simulating ASR with WoZ evaluations

One of the challenges in HRI interaction that may require an ad-hoc solution instead of a multipurpose API, is the speech recognition with relative movements between the speaker and the robot. In scenarios where ASR is performed by moving robots, the corruption of speech produced by the additive noise of the robot’s motors should be taken into consideration. Speech recognition experiments with moving robots in [53] led the authors to recommend that the robot should pause its actions as soon as it realizes that it is being talked to, which in some applications is unacceptable. They also suggest that the only reliable speech recognition engine for HRI is another human being. Given the fragility of ASR technology that was unveiled in HRI environments, many researchers have adopted interaction mechanisms that do not rely on speech recognition technology and Wizard of Oz (WoZ) based approaches have been chosen by several authors [54–61].

2.4 Evaluation of optimal physical set up and operating conditions

There is an alternative strategy, which instead of making the ASR technology more suitable to target operating conditions or adopting WoZ schemes, attempts to find the optimal operating environment that maximizes the ASR accuracy. In [51], the following variables were evaluated: different noise scenarios; different distances and angles of the speaker with respect to the microphones; three types of microphones, i.e. desktop, studio and the robot-mounted microphone. According to the experimental results, the authors provide recommendations regarding how the speech-based HRI with children should be deployed so as to achieve a smoother interaction. Some of the recommendations are: using additional input/output devices, even replacing verbal language input with a touchscreen; and, to place the user in an optimal location with respect to the microphones. Although these recommendations are based on

evaluations with children, the authors suggest that they are applicable to HRI in general.

A speech recognition friendly artificial language (ROILA) was compared to English spoken language when talking to a Nao robot in [53]. The experiment considered: three microphone types (the ones built-in in the robot, a headset, and a desktop microphone); two conditions of head movement (static and moving) for the Nao robot; and, the two types of spoken languages (English and ROILA). The authors concluded that ROILA does not provide a significant improvement when compared to ordinary spoken English. However, the type of microphone and the robot's head movement are critical for the ASR accuracy.

If ideal operating conditions are not met, one strategy is to try to cancel the corrupting environments. For instance, in [62] and [63] the external noise sources or ego-noise caused by motors and fans of the robot are removed with enhancement methods.

3 GENERIC ASR TEST BED FOR HRI

In contrast to the ASR integration on a black box basis as discussed above, we propose to consider not only the acoustic signal but also the operation conditions such as the environment, and robot and user state and contexts (Fig. 2).

Figure 2: Proposed ASR integration in HRI scenarios.

By environment we mean basically the acoustic channel, reverberation conditions and the additive noise caused by the robot movement. Robot state and context denote all the information about current variables and operating conditions of the machine to generate a list of feasible or acceptable commands or information that could be input by the user. Finally, user state and context designate, among others, the user's attitude, emotional conditions, and task completion status that can also predict user's command and info input to the robot. The full accomplishment of this kind of integration is far beyond the scope of a single paper, and we focus here on the environment representation and modeling by training our ASR engine with clean utterances combined with the acoustic-channel responses and noise that were estimated and recorded, respectively, with an HRI testbed. This testbed attempts to represent the generic acoustic environment of HRI in mobile robotics from the ASR point of view.

First of all, for instance, consider some real human social scenarios where robots could be very useful: a museum guide giving a tour, a student in a classroom asking the teacher a

question, a rescue team helping a survivor and a team of chefs working in a restaurant. All these situations have something in common: a person talks to somebody else who is busy accomplishing a task and is not looking to who is talking to him/her. Also, the two individuals may be moving one with respect to the other.

As shown in Fig. 2, the proposed strategy considers the information related to the acoustic environment as one of the inputs of the ASR engine. In this paper we represent the acoustic environment with the impulse responses that characterize the time-varying acoustic channel (TVAC) and the additive noise generated by the robot movement. The main advantage of this strategy is the fact that it is much more efficient than recording the training database in all the possible operating conditions. To record the testing speech data in a real mobile robot scenario, to estimate the channel impulse responses and to record the robot noise, we implemented a testbed that employs a loudspeaker and human speakers as sources plus a moving robot as a receiver.

A preliminary version of this testbed was described in [64] where pilot experiments were reported. Because of the high relevance to the HRI community, a more complete version of this type of HRI scenario is proposed and described in the following subsections. Particularly, different types of robot noise were recorded and included in the training procedure to represent more accurately the robot movement-conditions and the acoustic environment. Also, additional test sets were recorded by replacing the loudspeaker with human speakers in the same context.

3.1 Robotic platform and database recording

Our experimental platform makes use of the PR2 (Personal Robot 2) shown in Fig. 3. Our PR2 is equipped with a Microsoft Xbox 360 Kinect sensor mounted on top of its head. We re-recorded 330 clean testing utterances of the Aurora-4 database with our HRI testbed located in a meeting room (Fig. 4) including different specifications of the relative motion between the robot and the sources. Note that when the source and the robot are static one with respect to the other is a special case in relation to the more general situation (see Fig. 3). The two audio sources corresponded to a studio loudspeaker and four native American English speakers (two males and two females). The recording was performed by the PR2 Microsoft Kinect sensor, which contains a four-microphone array. The four signals received were summed to obtain a single channel signal. The recording procedure considered the relative movements of the robot microphones with respect to the sources by simultaneously applying translational movement to the robot body and angular rotation to the robot head.

3.1.1 Robot displacement

The robot moved towards and away from the source (i.e. the loudspeaker or the human speakers) between positions P1 and P3 (see Fig. 4). Three maximum robot displacement velocities were defined: $V_{max_1} = 0.30 \text{ m/s}$, $V_{max_2} = 0.45 \text{ m/s}$ and $V_{max_3} = 0.60 \text{ m/s}$. Those velocities were inspired by the discussions in [65], where a robot approached to a seated person

Figure 3: PR2 robot equipped with a Microsoft Kinect that was used to record the database: a) the source corresponds to a studio loudspeaker that was employed to reproduce clean utterances from a database; and, b) the source is a human speaker reading sentences from the same corpus.

at 0.2 m/s and 0.4 m/s . In those conditions, none of the human participants found these robot speeds were too fast. Then, the maximum velocities mentioned above were multiplied by an acceleration and deceleration function. Additionally, the

Figure 4: Meeting room where the HRI scenario was implemented. The robot moved towards and away from the source (i.e. the loudspeaker or the human speakers) between positions P1 and P3.

recording of the test database was also performed with the robot in a static condition with respect to the source at position P1.

Consequently, four robot displacement scenarios were considered for the test data recording: three translational movements between P1 and P3 with maximum velocities V_{max_1} , V_{max_2} and V_{max_3} ; and, a static position at P1.

3.1.2 Head rotation

The robot makes turns with the head as shown in Fig. 5 for each of the four displacement conditions described above. The robot head moves periodically from -150° to 150° and back at three angular velocities. The sources are located at 0° . The three angular velocities ω_i for the robot head were made equal to: 0.28 rad/s , 0.42 rad/s and 0.56 rad/s . The chosen angular velocities correspond to the angular speed of the head rotation necessary for the robot to follow a target with its head movement. The target is located two meters away from the robot and it is moving with tangential velocities of 2 km/h , 3 km/h and 4 km/h , respectively, as shown in Fig. 5. A fourth angular motion condition was zero, fixing the robot's head at 0° (i.e., oriented towards the source) for each robot displacement described above.

Figure 5: Movement of the PR2 robot head during the utterances recording. The head moves periodically from -150° to 150° and back at angular velocities equal to 0.28 rad/s , 0.42 rad/s and 0.56 rad/s . Recordings with static head are performed at 0° . The selected angular velocities for the robot head emulates the situations where the robot follows with the head a target located two meters away and moving with linear velocities of 2 km/h , 3 km/h and 4 km/h , respectively. The acoustic sources can be a loudspeaker or a human speaker. In both cases the sources were located at 0° with respect to the robot.

3.1.3 HRI scenario testing databases

The combination of four conditions for robot displacement and four robot's head angular movements produces 16 test database recording conditions. Consequently, the total number of Aurora-4 clean testing utterances reproduced with the studio

loudspeaker is equal to 330 utterances/robot-movement-conditions $\times$ 16 robot-movement-conditions = 5280 utterances. On the other hand, each of the four native American English speakers pronounced ten sentences from the Aurora-4 corpus per robot-movement-conditions. Those sentences were the same for all the four speakers. As a result, the human speakers recorded 4 $\times$ 10 utterances/robot-movement-conditions $\times$ 16 robot-movement-conditions = 640 utterances. The average number of words per utterances is equal to 16.2 words. The vocabulary size in the testing data is 1270 words. It is important to mention that background noise was kept under control and measured before recording the test database at each robot movement condition. The equivalent sound pressure level over ten minutes was equal to 39 dBA. Instructions for requesting the HRI playback testing database are available at <http://www.lptv.cl/en/hri-asr/>. Further information about the testing database recording can be found in [66].

3.2 Representing time varying acoustic channel

TVAC in this HRI scenario can be modeled using a set of samples of the acoustic channel impulse responses. In this paper 33 four-channel impulse responses (IRs) were computed with the robot placed at P1, P2 and P3 (Fig. 4), and for each robot position the head was oriented at 11 different angles with respect to the source. The head angle was varied from -150° to 150° in steps of 30° . Angle 0° corresponds to the Microsoft Kinect microphones oriented towards the sources in Fig. 5. The impulse responses were estimated using the Farina's sine sweep method [67]. An exponential sine sweep signal was generated from 64 Hz to 8 kHz and reproduced with a studio loudspeaker. The sweep audio was recorded with the four channel Microsoft Kinect sensor. An impulse response was estimated for each channel by convolving the corresponding recorded signal with the time-reversal of the original exponential sine sweep.

3.3 Noise recording

To incorporate additional information about the acoustic environment in our HRI scenario, different robot noise levels were recorded by the Kinect microphone array in the 16 robot movement conditions. The recorded noise was included in the ASR training procedure. The robot noise is generated by its internal fans and electrical motors operating at different translational and angular velocities. Finally, the four Kinect channels were summed to obtain a single channel signal.

4 DNN-HMM BASED ASR TRAINING

The speech recognition experiments reported here made use of Aurora-4 database, which in turn was generated with the 5000-word closed-loop vocabulary task based on the DARPA Wall Street Journal (WSJ0) Corpus [68]. To generate the Environment-based Training (EbT) set, 25% of the clean training utterances of the Aurora-4 database, which consists of 7138 utterances (i.e. 15.2 hours) from 83 native English speakers and contains only data recorded with a high-quality microphone

(i.e. Sennheiser HMD-414), was convolved with the IRs, corresponding to the four Kinect channels, estimated when the robot-source distance was equal to 1 m and the angle between the robot head and source was 0° . Then, the four convolution results were summed to obtain a single channel signal. The remaining 75% of the clean training set was convolved with the remaining 32 four-channel IRs by employing the same procedure described above, in such a way that the IRs were evenly distributed across the signals. The recorded noise was added to this 75% of utterances using the Filtering and Noise Adding Tool FaNT [69] at SNR between 10 dB and 20 dB. It is worth highlighting that this training data is completely different from the testing databases described above, i.e. different speakers and different utterances.

In this paper, the experiments were performed with a DNN-HMM ASR using the Kaldi Speech Recognition Toolkit [44], which is a state of the art and competitive ASR technology as mentioned above. To build a DNN-HMM system with Kaldi, first a GMM-HMM is trained with the EbT training data, using the tri2b Kaldi recipe for the Aurora-4 database. In this recipe, a monophone system is trained; then, the alignments from that system are employed to generate an initial triphone system; finally, the triphone alignments are employed to train the final triphone system. Also, Mel-frequency cepstral coefficients (MFCC) parametrization of speech, linear discriminant analysis (LDA), and maximum likelihood linear transforms (MLLT) are part of the recipe. Once the GMM-HMM system is trained, the GMM is replaced with a DNN. The DNN is composed of seven hidden layers and 2048 units per layer each, and the input considers a context window of 11 frames. The number of units of the output DNN layer is equal to the number of Gaussians in the corresponding GMM-HMM system. The reference for the DNN training is the alignment obtained with the clean version of the whole training data and the GMM-HMM trained with the same clean data. This leads to a better reference for the DNN than using the noisy or corrupted speech data directly [70,71]. The DNN is trained firstly using the Cross-Entropy criterion. Then, the final system is obtained by re-training the DNN with the sMBR discriminative training [72]. Our final ASR system is referred as EbT. For comparison reasons, we also trained a DNN with the clean database without any information regarding the HRI testbed scenario.

For decoding, the standard 5K lexicon and trigram language model from WSJ were used [73]. As a result, the language model is tuned to the task, i.e. it is task dependent. The required files and scripts to generate the EbT training data and the detailed Kaldi recipe to train the DNN-HMM based ASR system employed here are available at <http://www.lptv.cl/en/hri-asr/>.

5 ASR RESULTS AND COMPARISONS WITH APIs

The average word error rate (WER) obtained with the 5280 utterances recorded in our HRI scenario (subsection 3.1.3) with the loudspeaker, was equal to 65.0% using our ASR system trained with clean data. When only the IRs were incorporated in

the training procedure, the average WER was dramatically reduced to 31.4%. Moreover, our EbT system (i.e. that includes both IRs and robot noise) provided a much lower WER: 11.6%. This dramatic increase of the ASR accuracy strongly supports our proposed approach to model the acoustic environment of an HRI scenario with channel impulse responses and robot additive noise. Observe that this WER was achieved with only 15 hours of training data. This result was corroborated by making use of our testing data set that was recorded with the four native American English speakers: 73.5% and 20.1% with clean training and EbT, respectively. These WERs are higher than those obtained with the playback testing data. This must be due to the fact that the human speakers pronounced the utterances with a lower volume resulting in a lower signal-to-noise ratio. Actually, the average SNRs were equal to 11 dB and 18 dB for human speakers and loudspeaker data, respectively.

For comparison reasons, we also ran ASR experiments with three publicly available APIs by using the “Speech Recognition” Python library (Version 3.7) [74]: the Google Web Speech API (Google API); the IBM Speech to Text API (IBM API); and, the Bing Voice Recognition API (Bing API). Fig. 6 shows the WER obtained with our EbT system and the three API mentioned above with the 330 clean utterances from Aurora-4. As it can be seen in Fig. 6, the EbT system provided the lowest WER that is 34% lower than the second best, i.e. IBM API. This result suggests that adopting a better tuned language model, as done in our EbT ASR system, provides a clear advantage over a flatter or more general-purpose language model.

In our HRI test sets, we observed that in challenging scenarios the APIs evaluated here delivered empty strings as the result of the ASR queries. Given this situation, the WERs were estimated with the non-empty returned text strings. Table 1 presents the ASR results obtained with our EbT system, Google API and IBM API, in all the robot motion conditions, with the playback loudspeaker testing database (subsection 3.1.3). In the case of Bing API, the empty string rate (ESR) increased dramatically and prevented us from showing a

Figure 6: WERs obtained with the EbT system and the publicly available ASR APIs. The testing data corresponds to the original clean test set from Aurora-4 database (330 utterances).

Table 1: WERs obtained with the EbT system, Google API and IBM API. The testing sets correspond to the playback loudspeaker sub databases recorded at each combination of robot displacement and robot head angular velocities.

ASR System					
v m/s	ω rad/s	Clean training	EbT	Google API	IBM API
0	0	8.63	4.11	7.94	9.27
	0.28	52.79	7.73	9.94	27.88
	0.42	53.09	7.68	10.91	25.94
	0.56	54.06	7.83	11.32	26.57
0.3	0	36.04	5.66	9.32	22.71
	0.28	76.42	12.22	16.31	49.23
	0.42	76.59	12.33	17.39	51.68
	0.56	78.8	12.82	17.88	51.3
0.45	0	43.36	6.16	9.04	22.55
	0.28	82.94	16.74	19.30	53.03
	0.42	83.49	15.28	20.73	54.69
	0.56	83.73	15.45	22.31	55.93
0.6	0	53.3	7.49	10.78	29.57
	0.28	85.8	18.27	22.06	56.9
	0.42	85.07	17.47	22.91	56.37
	0.56	86.46	18.68	24.42	58.19
AVG		65.04	11.62	15.79	40.74

representative WER. All the ASR results with the APIs shown in Table 1 were carried out between September 6th and 12th, 2017. According to Table 1 the lowest and highest WERs were achieved with the static condition (i.e. translation and angular velocities equal to zero), and with the highest displacement and rotational velocities, respectively, with EbT, Google API and IBM API. Also, the lowest WER for each robot movement condition is achieved with EbT.

Fig. 7 summarizes the WERs obtained with EbT, Google API and IBM API, in all the robot movement conditions shown in Table 1, with the playback loudspeaker testing database (subsection 3.1.3). As can be seen in Table 1 and Fig. 7, the lowest WER correspond to our EbT system. The average WER achieved with the EbT system is 26% lower than the second best, i.e. Google API. According to Fig. 7, the EbT system and Google API provided the lowest WER dispersion. Also, the observed average empty string rates or ESRs were equal to 0%, 0.3% and 6.5% with EbT, Google API and IBM API, respectively. If we include the empty strings in the computation of the error rates, the WERs increased to 15.9% and 42.6% with Google API and IBM API, respectively. With EbT the WER was not modified because ESR is equal to zero in this case.

For validation purposes, Fig. 8 summarizes the WERs obtained with EbT, Google API and IBM API, in all the robot movement conditions shown in Table 1, with the native American English speaker testing database (subsection 3.1.3). According to Fig. 8, the lowest value and dispersion for WER also corresponds to our EbT system. The average WER achieved

Figure 7: WERs obtained with the EbT system, Google API and IBM API in all the robot movement conditions shown in Table 1, with the playback loudspeaker testing database (subsection 3.1.3).

with system EbT is 38% lower than the second best, i.e. Google API. The average ESRs with the human speaker testing data set are equal to 0%, 5.8% and 5.6%. If we include the empty strings in the computation of the error rates, the WERs increased to 35.0% and 57.1% with Google API and IBM API, respectively. The results with the ASR APIs using the native American English speaker testing database were obtained between September 25th and October 5th, 2017.

By comparing Fig. 6 with Fig. 7, we can observe that the lowest WER is achieved with our EbT system. However, the EbT system also provides the highest relative increase in average WER, i.e. 231%, from the clean testing data to the playback loudspeaker testing database (subsection 3.1.3) in the HRI

Figure 8: WERs obtained with the EbT system, Google API and IBM API in all the robot movement conditions shown in Table 1, with the native American English speakers testing database (subsection 3.1.3). The average WERs were 20.1%, 32.6% and 56.4% with EbT, Google API and IBM API, respectively.

scenario. In contrast, Google API, for instance, shows a relative increase in average WER equal to 117%. This result can be explained according to [75,76] and [77], where it is said that the ASR engines that support the APIs evaluated here could have been trained with at least thousands of hours of speech covering a wide diversity of acoustic conditions. In contrast, our EbT system was trained with only 15.2 hours of clean speech utterances that were convolved with channel impulse responses and had noise added (section 4). The proposed procedure is applicable to any HRI environment, being only necessary the capture of the robot noise and the estimation of the acoustic impulse responses to get a new EbT system. This procedure requires just a couple of days and a few hours of training data. At this point it is worth highlighting that the adequate use of user and robot states and contexts can reduce the language model perplexity, and lead to further improvements in recognition accuracy.

6 CONCLUSIONS

First, we propose to replace the popular black box integration of automatic speech recognition technology in HRI applications with the addition of the HRI environment representation and modeling, and the robot and user states and contexts. Then, as a consequence of this strategy, this paper was focused on the environment representation and modeling by training a DNN-HMM model based automatic speech recognition engine with the combination of clean utterances with the acoustic-channel responses and noise that were estimated and recorded, respectively, with an HRI testbed built with a PR2 robot. The proposed procedure is much more effective and efficient than recording a training database in all the possible acoustic environments, given an HRI scenario. Also, different speech recognition testing conditions were generated by recording two types of acoustic sources, i.e. a loudspeaker and human speakers, using the PR2 robot, which has a Microsoft Kinect sensor mounted on top, while performing head rotations and movements towards and away from the fixed sources. This testbed models the generic problem of HRI in mobile robotics, and the resulting automatic speech recognition accuracy outperformed publicly available speech recognition APIs. The word error rate achieved by our system is at least 26% and 38% lower than the evaluated APIs with the loudspeaker and human testing databases, respectively, with a limited amount of training data. Other factor in HRI scenarios is that the user speech may be stressed in noisy conditions, i.e. Lombard effect. This problem, and the incorporation of user and robot states and contexts are proposed for future research.

ACKNOWLEDGMENTS

The research reported here was funded by Grants Conicyt-Fondecyt 1151306 and ONRG N°62909-17-1-2002. José Novoa was supported by Grant CONICYT-PCHA/Doctorado Nacional/2014-21140711. The authors would also like to thank Prof. Henny Admoni, CMU, for her valuable comments and suggestions to improve the quality of the manuscript.

REFERENCES

- [1] M. A. Goodrich and A. C. Schultz. 2007. Human-Robot Interaction: A Survey. *Foundations and Trends in Human-Computer Interaction*, vol. 1, no. 3, p. 203–275.
- [2] L. S. Lopes and A. Teixeira. 2000. Human-robot interaction through spoken language dialogue. In *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, Takamatsu, Japan.
- [3] G. Hoffman and K. Vanunu. 2013. Effects of robotic companionship on music enjoyment and agent perception. In *Proceedings of ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, Tokyo, Japan.
- [4] C. Y. Lin, K. T. Song, Y. W. Chen, S. C. Chien, S. H. Chen, C. Y. Chiang, J. H. Yang, Y. C. Wu and T. J. Liu. 2012. User identification design by fusion of face recognition and speaker recognition. In *Proceedings of 12th International Conference on Control, Automation and Systems*, JeJu Island, South Korea.
- [5] K. Zheng, D. F. Glas, T. Kanda, H. Ishiguro and N. Hagita. 2013. Designing and Implementing a Human-Robot Team for Social Interactions. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 13, no. 4, pp. 843–859.
- [6] Y. Kondo, K. Takemura, J. Takamatsu and T. Ogasawara. 2013. A gesture-centric android system for multi-party human-robot interaction. *Journal of Human-Robot Interaction*, vol. 2, no. 1, pp. 133–151.
- [7] D. Wang, H. Leung, A. P. Kurian, H. J. Kim and H. Yoon. 2010. A Deconvolutional Neural Network for Speech Classification With Applications to Home Service Robot. *IEEE Transactions on Instrumentation and Measurement*, vol. 59, no. 12, pp. 3237 – 3243.
- [8] E. L. Meszaros, M. Chandarana, A. Trujillo and B. D. Allen. 2018. Compensating for Limitations in Speech-Based Natural Language Processing with Multimodal Interfaces in UAV Operation. In *Advances in Human Factors in Robots and Unmanned Systems. AHFE 2017. Advances in Intelligent Systems and Computing*, California, LA, USA.
- [9] S. Han, J. Hong, S. Jeong and M. Hahn. 2010. Robust GSC-based speech enhancement for human machine interface. *IEEE Transactions on Consumer Electronics*, vol. 56, no. 2, pp. 965–970.
- [10] M. Staudte and M. W. Crocker. 2011. Investigating joint attention mechanisms through spoken human-robot interaction. *Cognition*, vol. 120, no. 2, pp. 268–291.
- [11] H. Polido. 2014. DARPA Robotics Challenge. Worcester Polytechnic Institute.
- [12] H. Kitano, M. Asada, Y. Kuniyoshi, I. Noda and E. Osawa. 1997. Robocup: The robot world cup initiative. In *Proceedings of the first international conference on Autonomous agents*, Marina del Rey, CA, USA.
- [13] L. Zhang, L. Du and Z. B. 2016. Deep Learning for Remote Sensing Data: A Technical Tutorial on the State of the Art. *IEEE Geoscience and Remote Sensing Magazine*, vol. 4, no. 2, pp. 22–40.
- [14] S. E. Umbaugh. 2010. Digital image processing and analysis: human and computer vision applications with CVPITools. CRC press.
- [15] W. Burger and M. J. Burge. 2016. Digital image processing: an algorithmic introduction using Java, Springer.
- [16] J. Nakamura. 2016. Image sensors and signal processing for digital still cameras, CRC press.
- [17] S. Young. 2008. HMMs and Related Speech Recognition Technologies. In *Springer Handbook of Speech Processing*, Springer, pp. 539–558.
- [18] X. D. Huang, Y. Ariki, and M. A. Jack. 1990. Hidden Markov models for speech recognition. Edinburgh university press Edinburgh, vol. 2004.
- [19] R. Justo and M. I. Torres. 2015. Integration of complex language models in ASR and LU systems. *Pattern Analysis and Applications*, vol. 18, no. 3, pp. 493–505.
- [20] S. F. Chen, D. Beeferman and R. Rosenfeld. 1998. Evaluation metrics for language models. In *Proceedings of DARPA Broadcast News Transcription and Understanding Workshop*, pp. 275–280.
- [21] M. Chetouani, B. Gas and J. Zarader. 2002. Discriminative Training for Neural Predictive Coding Applied to Speech Features Extraction. In *Proceedings of the 2002 International Joint Conference on Neural Networks*, Honolulu, HI, USA.
- [22] N. Dave. 2013. Feature Extraction Methods LPC, PLP and MFCC in Speech Recognition. *International Journal for Advance Research in Engineering and Technology*, vol. 1, no. 6, pp. 1–5.
- [23] S. Furui. 1986. Speaker independent isolated word recognition using dynamic features of speech spectrum. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 34, no. 1, p. 52–59.
- [24] L. Bahl, R. Bakis, E. Jelinek and R. Mercer. 1980. Language-model/acoustic channel balance mechanism. *IBM Technical Disclosure Bulletin*, vol. 23, no. 7B, pp. 3464–3465.
- [25] G. Hinton, L. Deng, D. Yu, G. E. Dahl, A. R. Mohamed, N. Jaitly and B. Kingsbury. 2012. Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. *IEEE Signal Processing Magazine*, vol. 29, no. 6, pp. 82–97.
- [26] J. Godfrey and E. Holliman. 1997. Switchboard-1 Release 2. Linguistic Data Consortium, Philadelphia.
- [27] G. E. Hinton, S. Osindero and Y.-W. Teh. 2006. A fast learning algorithm for deep belief nets. *Neural computation*, vol. 18, no. 7, pp. 1527–1554.
- [28] J. Schröder, J. Anemüller and S. Goetze. 2016. Performance comparison of GMM, HMM and DNN based approaches for acoustic event detection within Task 3 of the DCASE 2016 challenge. In *Proceedings of Workshop on Detection and Classification of Acoustic Scenes and Events*, Budapest, Hungary.
- [29] S. Hochreiter and J. Schmidhuber. 1997. Long short-term memory. *Neural Computation*, vol. 9, no. 8, p. 1735–1780.
- [30] O. Abdel-Hamid, A. R. Mohamed, H. Jiang, L. Deng, G. Penn and D. Yu. 2014. Convolutional neural networks for speech recognition. *IEEE/ACM Transactions on audio, speech, and language processing*, vol. 22, no. 10, pp. 1533–1545.
- [31] A. Graves, A. R. Mohamed and G. Hinton. 2013. Speech recognition with deep recurrent neural networks. In *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing*, Vancouver, BC, Canada.
- [32] Z. Tang, D. Wang and Z. Zhang. 2016. Recurrent neural network training with dark knowledge transfer. In *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing*, Shanghai, China.
- [33] J. Li, A. Mohamed, G. Zweig and Y. Gong. 2015. LSTM time and frequency recurrence for automatic speech recognition. In *IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, Scottsdale, AZ, USA.
- [34] T. N. Sainath and B. Li. 2016. Modeling Time-Frequency Patterns with LSTM vs. Convolutional Architectures for LVCSR Tasks. In *Proceedings of INTERSPEECH*, San Francisco, USA.
- [35] Y. Liu and K. Kirchhoff. 2016. Novel Front-End Features Based on Neural Graph Embeddings for DNN-HMM and LSTM-CTC Acoustic Modeling. In *Proceedings of INTERSPEECH*, San Francisco, USA.
- [36] D. Yu, W. Xiong, J. Droppo, A. Stolcke, G. Ye, J. Li and G. Zweig. 2016. Deep convolutional neural networks with layer-wise context expansion and attention. In *Proceedings of INTERSPEECH*, San Francisco, USA.
- [37] Y. Qian, M. Bi, T. Tan and K. Yu. 2016. Very Deep Convolutional Neural Networks for Noise Robust Speech Recognition. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 24, no. 12, pp. 2263–2276.
- [38] V. Mitra and H. Franco. 2016. Coping with Unseen Data Conditions: Investigating Neural Net Architectures, Robust Features, and Information Fusion for Robust Speech Recognition. In *Proceedings of INTERSPEECH*, San Francisco, USA.
- [39] C. Weng, D. Yu, M. L. Seltzer and J. Droppo. 2014. Single-channel mixed speech recognition using deep neural networks. In *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing*, Florence, Italy.
- [40] S. Young, G. Evermann, M. Gales, T. Hain, D. Kershaw, X. Liu, G. Moore, J. Odell, D. Ollason, D. Povey and others. 2006. The HTK book. *Cambridge university engineering department*, vol. 3, p. 175.
- [41] K. F. Lee, H. W. Hon and R. Reddy. 1990. An overview of the SPHINX speech recognition system. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 38, no. 1, pp. 35–45.
- [42] W. Walker, P. Lamere, P. Kwok, B. Raj, R. Singh, E. Gouvea, P. Wolf and J. Woelfel. 2004. Sphinx-4: A flexible open source framework for speech recognition. Sun Microsystems, Inc.
- [43] A. Lee, T. Kawahara and K. Shikano. 2001. JULIUS - an open source real-time large vocabulary recognition engine. In *Proceeding of INTERSPEECH*, Aalborg, Denmark.
- [44] D. Povey, A. Ghoshal, G. Boulianne, N. Goel, M. Hannemann, Y. Qian, P. Schwarz and G. Stemmer. 2011. The Kaldi Speech Recognition Toolkit. In

Proceedings of ASRU, Hawaii, USA, December.

- [45] D. Bolaños. 2012. The Baviaca open-source speech recognition toolkit. In *Proceedings of IEEE Spoken Language Technology Workshop (SLT)*, Miami, FL, USA.
- [46] D. O. Johnson, R. H. Cuijpers, J. F. Juola, E. Torta, M. Simonov, A. Frisiello, M. Bazzani, W. Yan, C. Weber, S. Wermter, N. Meins, J. Oberzaucher, P. Panek, G. Edelmayer and P. Mayer. 2014. Socially Assistive Robots: A Comprehensive Approach to Extending Independent Living. *International Journal of Social Robotics*, vol. 6, no. 2, p. 195–211.
- [47] J. F. Lehman. 2014. Robo fashion world: a multimodal corpus of multi-child human-computer interaction. In *Proceedings of the 2014 Workshop on Understanding and Modeling Multiparty, Multimodal Interactions*, Istanbul, Turkey.
- [48] F. Cutugno, A. Finzi, M. Fiore, E. Leone and S. Rossi. 2013. Interacting with robots via speech and gestures, an integrated architecture. In *Proceedings of INTERSPEECH*, Lyon, France.
- [49] K. Zinchenko, C. Y. Wu and K. T. Song. 2017. A Study on Speech Recognition Control for a Surgical Robot. *IEEE Transactions on Industrial Informatics*, vol. 13, no. 2, pp. 607–615.
- [50] C. Matuszek, L. Bo, L. Zettlemoyer and D. Fox. 2014. Learning from Unscripted Deictic Gesture and Language for Human-Robot Interactions. In *Proceedings of the 28th National Conference on Artificial Intelligence*, Québec City, Quebec, Canada.
- [51] J. Kennedy, S. Lemaignan, C. Montassier, P. Lavalade, B. Irfan, F. Papadopoulos, E. Senft and T. Belpaeme. 2017. Child speech recognition in human-robot interaction: evaluations and recommendations. In *Proceedings of ACM/IEEE International Conference on Human-Robot Interaction*, Vienna, Austria.
- [52] P. Lange and D. Suendermann-Oeft. 2014. Tuning Sphinx to Outperform Google's Speech Recognition API. In *Proceedings of the Conference on Electronic Speech Signal Processing*, Dresden, Germany.
- [53] O. Mubin, J. Henderson and C. Bartneck. 2014. You just do not understand me! Speech Recognition in Human Robot Interaction. In *Proceedings of 23rd IEEE International Symposium on Robot and Human Interactive Communication*, Edinburgh, Scotland.
- [54] M. Marge, C. Bonial, B. Byrne, T. Cassidy, A. W. Evans, S. G. Hill and C. Voss. 2017. Applying the Wizard-of-Oz technique to multimodal human-robot dialogue. *arXiv preprint arXiv:1703.03714*.
- [55] P. Sequeira, P. Alves-Oliveira, T. Ribeiro, E. Di Tullio, S. Petisca, F. S. Melo, G. Castellano and A. Paiva. 2016. Discovering social interaction strategies for robots from restricted-perception Wizard-of-Oz studies. In *Proceedings of 11th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, Christchurch, New Zealand.
- [56] K. Hensby, J. Wiles, M. Boden, S. Heath, M. Nielsen, P. Pounds, J. Riddell, K. Rogers, N. Rybak, V. Slaughter, M. Smith, J. Taufatofua, P. Worthy and J. Weigel. 2016. Hand in hand: Tools and techniques for understanding children's touch with a social robot. In *Proceedings of 11th ACM/IEEE International Conference on Human-Robot Interaction*, Christchurch, New Zealand.
- [57] G. Hoffman. OpenWoZ: A Runtime-Configurable Wizard-of-Oz Framework for Human-Robot Interaction. 2016. In *Proceedings of AAAI Spring Symposium Series*, Palo Alto, CA, USA.
- [58] N. Martelaro. 2016. Wizard-of-Oz Interfaces as a Step Towards Autonomous HRI. In *Proceedings of AAAI Spring Symposium Series*, Palo Alto, CA, USA.
- [59] S. Pourmehr, J. Thomas and R. Vaughan. 2016. What untrained people do when asked "make the robot come to you". In *Proceedings of 11th ACM/IEEE International Conference on Human-Robot Interaction*, Christchurch, New Zealand.
- [60] E. Senft, P. Baxter, J. Kennedy, S. Lemaignan and T. Belpaeme. 2016. Providing a robot with learning abilities improves its perception by users. In *Proceedings of 11th ACM/IEEE International Conference on Human-Robot Interaction*, Christchurch, New Zealand.
- [61] J. M. K. Westlund and C. Breazeal. 2016. Transparency, teleoperation, and children's understanding of social robots. In *Proceedings of 11th ACM/IEEE International Conference on Human-Robot Interaction*, Christchurch, New Zealand.
- [62] H. W. Löllmann, A. Moore, P. A. Naylor, B. Rafaely, R. Horaud, A. Mazel and W. Kellermann. 2017. Microphone array signal processing for robot audition. In *Proceedings of Hands-free Speech Communications and Microphone Arrays*, San Francisco, CA, USA.
- [63] A. Deleforge and W. Kellermann. 2015. Phase-optimized K-SVD for signal extraction from underdetermined multichannel sparse mixtures. In *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing*, Brisbane, QLD, Australia.
- [64] J. Novoa, J. Wuth, J. P. Escudero, J. Fredes, R. Mahu, R. Stern and N. B. Yoma. 2017. Robustness over time-varying channels in DNN-HMM ASR based human-robot interaction. In *Proceedings of Interspeech*, Stockholm, Sweden.
- [65] K. Dautenhahn, M. Walters, S. Woods, K. L. Koay, C. L. Nehaniv, A. Sisbot, R. Alami and T. Siméon. 2006. How may I serve you?: a robot companion approaching a seated person in a helping context. In *Proceedings of ACM Conference on Human Robot Interaction*, Salt Lake City, UT, USA.
- [66] J. Novoa, J. Wuth, J. P. Escudero, J. Fredes, R. Mahu and N. Becerra Yoma. 2017. Multichannel Robot Speech Recognition Database: MChRSR. *arXiv preprint arXiv: 1801.00061*.
- [67] A. Farina. Simultaneous measurement of impulse response and distortion with a swept-sine technique. 2000. In *Proceedings of 108th Audio Engineering Society Convention*, Paris, France.
- [68] G. Hirsch. 2002. Experimental Framework for the Performance Evaluation of Speech Recognition Front-ends on a Large Vocabulary Task, Version 2.0, AU/417/02. *ETSI STQ Aurora DSR Working Group*.
- [69] G. Hirsch. 2005. FaNT filtering and noise adding tool. Niederrhein University of Applied Sciences.
- [70] S. Sivasankaran, E. Vincent and I. Illina. 2017. A combined evaluation of established and new approaches for speech recognition in varied reverberation conditions. *Computer Speech & Language*, vol. 46, no. Supplement C, pp. 444–460.
- [71] P. Lin, D.-C. Lyu, F. Chen, S.-S. Wang and Y. Tsao. 2017. Multi-style learning with denoising autoencoders for acoustic modeling in the internet of things (IoT). *Computer Speech & Language*, vol. 46, no. Supplement C, pp. 481–495.
- [72] K. Veselý, A. Ghoshal, L. Burget and D. Povey. 2013. Sequence-discriminative training of deep neural networks. In *Proceeding of INTERSPEECH*, Lyon, France.
- [73] J.-L. Gauvain, L. Lamel and M. Adda-Decker. 1995. Developments in continuous speech dictation using the ARPA WSJ task. In *Proceedings of International Conference on Acoustics, Speech, and Signal Processing*, Detroit, MI, USA.
- [74] A. Zhang. Speech Recognition (Version 3.7). 2017. [Online]. Available: https://github.com/Uberi/speech_recognition#readme. [Accessed 5th September 2017].
- [75] B. Li, T. Sainath, A. Narayanan, J. Caroselli, M. Bacchiani, A. Misra, I. Shafran, H. Sak, G. Pundak, K. Chin and others. 2017. Acoustic Modeling for Google Home. In *Proceedings of INTERSPEECH*, Stockholm, Sweden.
- [76] G. Saon, H.-K. J. Kuo, S. Rennie and M. Picheny. 2015. The IBM 2015 English Conversational Telephone Speech Recognition System. In *Proceedings of INTERSPEECH*, Dresden, Germany.
- [77] W. Xiong, J. Droppo, X. Huang, F. Seide, M. Seltzer, A. Stolcke, D. Yu and G. Zweig. 2017. The microsoft 2016 conversational speech recognition system. In *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing*, New Orleans, LA, USA.