An Open Source Corpus and Recording Software for Distant Speech Recognition with the Microsoft Kinect

Dirk Schnelle-Walka, Stephan Radeck-Arneth, Chris Biemann, Stefan Radomski

Telecooperation - Language Technology, Technische Universität Darmstadt, 64289 Darmstadt, Germany
Email: {dirk, stephan.radeck-arneth, radomski}@tk.informatik.tu-darmstadt.de
Email: biem@cs.tu-darmstadt.de
Web: www.{tk, lt}.informatik.tu-darmstadt.de

Abstract

A basic requirement for improvements in distant speech recognition is the availability of a respective corpus of recorded utterances. With microphone arrays now available off-the-shelf as part of the Microsoft Kinect, a common recording device for such a corpus is wide-spread. In this paper, we introduce KiSRecord, an open source recording tool that can be used to alleviate this situation for data collection. Thus, we provide the first steps towards a *community sourcing* effort for a speech corpus recorded with de-facto standard microphone arrays.

1 Introduction

The usage of voice-based interfaces in home environments requires robust speech recognition. One of the "greatest barriers to the uptake of ASR is the lack of robustness to interfering noise sources" [1]. That is why in current applications, users have to pick up devices to explicitly enter their command. A hands&eyes-free access still bears some challenges that hamper the deployment of actual applications, including the isolation of the audio source [2] and the detection of the onset of speech [3]. The first challenge includes distortions from echo and cross talk, while the latter includes communication channels in general [4]. For the first problem, there are already off-the-shelf solutions like the Microsoft Kinect. It uses a microphone array of four microphones to implement beamforming, echo cancellation and sound source localization [5]. It is also shipped with the Microsoft Speech Server that is used for speech recognition. However, this closed solution has some limitations. Besides the fact that dictation is not directly possible since it can currently only be used with grammars, it also limits the possibilities of developers and researchers to investigate other problems to improve accuracy and usability of the targeted applications. A viable solution would be to use other speech recognizers with the Kinect. This is possible since the beamformed audio stream can be captured and fed into the recognizer, but it will perform badly since the employed acoustic model has not been trained for this sort of incoming audio.

In Section 3 we will have a closer look at the specifics of the beamformed audio recorded with the Kinect. Our overall goal is to make the first steps towards an open source corpus that can be used for distant speech recognition with off-the-shelf devices, described in Section 5. This freely available corpus can be employed to train open source speech recognizers, e.g. Sphinx [6], Julius [7] or the more recent Kaldi [8]. Further, we provide a software to enable community contributions to this corpus with the **Ki**nect **S**peech **Record**er (KiSRecord), described in Section 4. Our hope is that other researchers could reuse it and share their recordings to lower the barriers to arrive at a larger corpus that gives rise to acceptable recognition results. We conclude our description in Section 6.

2 Related Work

Some open source speech recognizers are already shipped with acoustic models that help developers to get started. Unfortunately, these models do not perform well. Arthur Chan, a former developer of the CMU Sphinx speech recognizer states that "the default model (trained by Broadcast News) is there but it is definitely not for usage for every speech applications" [9].

Most of the speech recognizers also feature a training environment that can be used to train custom models. However, this does not alleviate users from conducting recording sessions. Sphinxtrain suggests to have at least 50 hours of recordings of 200 speakers for multiple speaker dictation models[1].

One approach to overcome this need is the VoxForge [10] speech collection portal. Other approaches with web based interfaces are those from Gruenstein et al. [11] and Schultz et al. [12]. VoxForge provides a Java applet that participants can use to provide speech samples of the ten major European languages. A major drawback of this approach, however, is the large variety in microphones: some users own high quality headsets, while others use the built-in microphones of their laptops. Hence, the models have only very limited use. Also, they aim at desktop settings, while we target distant speech recognition in the scope of this paper. To our knowledge, there are no efforts to provide open source acoustic models for distant speech recognition.

Lane et al. [13] describe a set of tools to collect speech corpora remotely and in unsupervised fashion with mobile phones. They compare the quality of speech with a web based approach and found that "although prompts were generally read accurately, lack of training led to a significantly lower yield of high quality recordings". As we employ a community-based sourcing, wherein we encourage members of the speech community to contribute to the corpus, we avoid the unsupervised setting, thereby trading corpus size for quality.

3 Audio Signal Analysis of the Kinect

The Kinect is equipped with four microphones (see Figure 1). As indicated in [5], its suitability for more elaborate audio processing algorithms is limited. The main reason lies in its physical design, which brings the microphones very close to each other. E.g., the three microphones to the right merely span a distance of approximately 10 cm. This

[1] http://cmusphinx.sourceforge.net/wiki/tutorialam

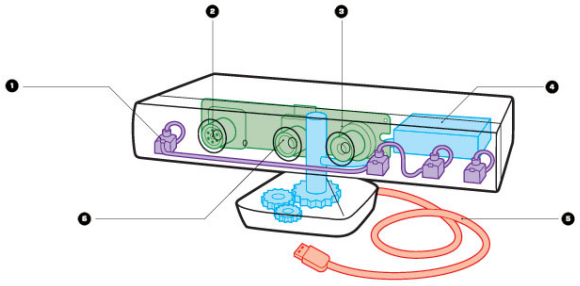

Figure 1: Microphones of the Kinect are labeled with 1, Source `http://www.openkinect.org/`

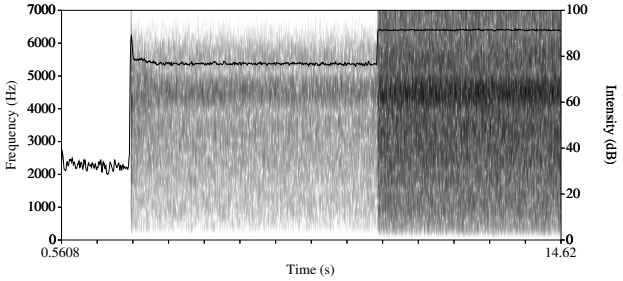

Figure 2: Spectrum and intensity of white noise recording with the Microsoft Kinect

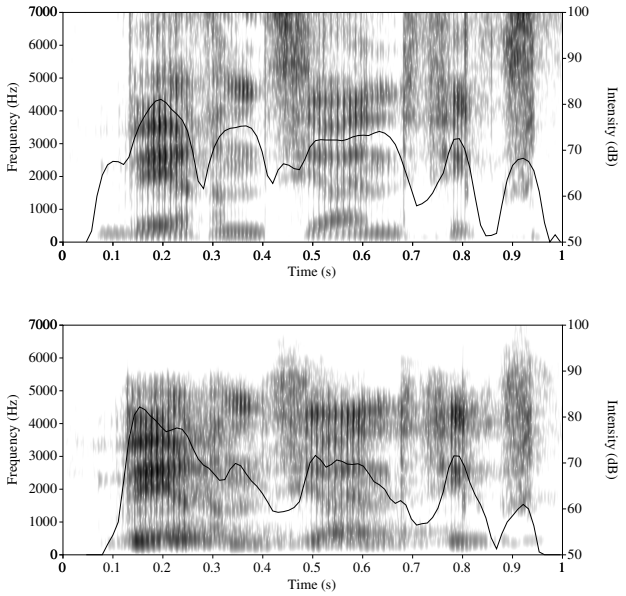

Figure 3: Spectrum and intensity of the human speech showing the first second of the sentence "*Während sie einen Fixbetrag an den Staat abführten, konnten sie Mehreinnahmen behalten*". Above the original recording and below the Kinect recording.

severely limits its capabilities to locate the audio source precisely and to suppress distortions.

We analyzed this behavior in a short test. White noise was played back with a Bose SoundLink speaker placed in front of the Kinect at a distance of 1.8 m, which is the recommended distance[2]. The spectrogram of the recorded audio signal is shown in Figure 2.

While the missing frequencies below 75 Hz are due to the frequency response of the speaker, the accentuation of frequencies at 4.4 kHz is due to the room acoustics and the microphone characteristics. There are some noteworthy conspicuities at 8.9 sec: It can clearly be seen that the beam is directed and signal strength increases by 15 dB.

Similar effects can be observed with human speech, though directing the beam is an order of magnitude faster (appr. 1 sec). Unfortunately, the beam seems to get misdirected after pauses, resulting in losses of input for the recognizer and different angles of the audio that are reported via the API. We verified this behavior by moving the speaker while human speech was played back. While we do not have any quantitative measurements yet, we are in the process of detailling and refining this observations.

Some researchers tried to improve the isolation of the audio source by combining it with information retrieved from the depth camera [14].

Figure 3 shows the different spectral characteristics of the original recording and the Kinect recording. While higher frequencies are hardly visible, the important frequencies in the range of human speech and the formants are still identifiable. This emphasizes the need for recordings of a specialized corpus.

We were also interested whether the spectra and this behavior differ in relation to the angle of the audio source. Therefore, we moved the speaker in intervals of 10° on a circle around the Kinect but found no peculiarity. Another observation was the fact that the angles of the Kinect that were reported by the API deviated by $6-7^\circ$ to the right from the actual position. This is, where the microphones of the Kinect are closer (refer to Figure 1). We verified deviation with another Kinect, leading us to the assumption that there must be some basic problems with the implementation of this feature. It seems that the Kinect does not combine information coming from the audio source localization with information coming from the depth camera as suggested in [14]. An experiment with the camera turned on and a person carrying the speaker yielded comparable results.

Audio source localization also works better for audio signals in the frequency range of human speech. It was faster, more stable and the angles' confidence values were always close to 1, while it was not reliable for white noise. Here, we observed confidence values in the range $[0,1]$.

Another observation that we made is that the recommended distance of 1.8 m seems to be solely based on the the Kinect's depth sensing capabilities. The audio signal was rather low at 1.8 m, but we got acceptable signal power at a distance of 1 m.

Based on our observations, we recommend the following for the recording sessions:

- The audio beam is not very stable after pauses, especially in the beginning. Since this will also be the case at runtime, it should be integrated into the training to learn this sort of variance in the speech signal.
- The distance between speaker and Kinect should be 1 m.

However, for a precise and reliable measurement, this experiment should be repeated with high quality speakers in an anechoic chamber.

[2] `http://support.xbox.com/en-US/xbox-360/kinect/sensor-placement`

Figure 4: KiSRecord screenshot

The new Kinect for the XBox One (est. 2014) will have some more audio processing capabilities that are better suited to capture human speech. However, it will still rely on beam forming[3]. At the point of writing the detailed specification is still unknown. We will investigate the capabilities of this new device, once the SDK is publicly available.

4 KiSRecord

Apart from the limitations mentioned in the previous section, the Kinect is still attractive to be used as a device for distant speech recognition. The huge advantage is that it is available as off-the-shelf device at a reasonable price and has already been deployed to many living rooms.

In an ongoing project, we plan to use it to explore improvements to speech recognition algorithms in home environments. Since the performance of speech recognizers depends on the acoustic model [15] and thereby also on the employed microphones [16], we need to train our own acoustic model. We believe that this de-facto standard microphone array is a suitable basis for a community-based approach to collect audio data for this purpose. As a first step we set up an open source project named KiSRecord at `http://kisrecord.sourceforge.net`.

Figure 4 shows a screenshot of the KiSRecord GUI. The GUI needs to be internationalized once we go for recordings in other languages than German. KiSRecord does not only record spoken utterances of the beamformed audio input with the Kinect, but also allows for recordings with multiple other microphones in parallel. Each microphone that is detected in the system settings is used for recordings. Users will have to disable the appropriate system setting in case they want to suppress recordings with a certain microphone. KiSRecord randomly selects a phrase from predefined databases and displays it. While the user is reading, the detected source angle and the recorded audio is shown to provide some feedback to the user about recording activities. Users may also repeat the recording with the same sentence at will, e.g. if they misread the sentence or were interrupted. It is also possible to abort/skip the current recording if the user has difficulties with a particular sentence. Each recording event generates a corresponding XML file with the meta data. Meta data contains

[3] `http://www.reddit.com/r/xboxone/comments/1iu8mr/audio_processing_improvements_to_the_new_kinect/`

Figure 5: Recording environment

the spoken sentence as well as statistics data of the user like age, dialect or gender that can optionally be entered via the GUI. For each detected input device, an individual wave file with the audio recording is created and stored accordingly.

5 Corpus

We used KiSRecord to create an initial audio corpus of distant speech recordings. We started for German as the first language and will target other languages in the near future. The Kinect was placed at a distance of 1 m. The recording environment is shown in Figure 5. The sentences read from the participants stem from German Wikipedia[4] articles (2549 words) and from German European Parliament[5] transcriptions (13823 words).

Here are some typical example sentences:

Wikipedia "Im Flug wirkt der Steinadler trotz seiner Größe meist sehr leicht und elegant."

European Parliament "Wir müssen auch konsequent darauf achten, daß unsere Vorgaben fristgerecht durch die Mitgliedstaaten umgesetzt werden, und noch wichtiger, wir müssen darauf achten, daß sie anschließend auch angewendet werden."

The main reason for our choice of corpora lies in the aim to capture simple sentence structures as well as spontaneous speech as we suspect it to occur during daily usage. Since we are aiming at providing our trained acoustic model as open source, we were also looking for source material that would allow us to do so.

Currently, we have recordings from 58 native German speakers (16 female and 42 male). Each speaker spent a total of 30 minutes for the recordings, which provided us with about 700 minutes of utterances in about 5000 wave files for the Kinect. Each recording session was split equally to record sentences from the Wikipedia articles and those from the European Parliament. The initial five minutes were used for the introduction to the system. During the recording session, the supervisor was in charge to control the system so that the speaker could concentrate on reading aloud.

The Wikipedia sentences were perceived easier to read aloud by the participants than sentences from the European Parliament Corpus. For the latter, speakers made more errors and required more repetitions. In these cases, the supervisor and the participant decided ad hoc whether to re-

[4] `http://de.wikipedia.org`

[5] `http://statmt.org/europarl/v7/de-en.tgz`

peat or to abort the last sentence. If the participant did not feel comfortable with her recording, the supervisor recommended to proceed with the next sentence.

Finally, we used this corpus to train models in CMU Sphinx [6], an open source speech recognizer.

For the dictionary, we used a freely licensed one for the German language provided by VoxForge. Not all of the words in the recordings were transcribed, so we were also employing a recommended workaround[6] to use eSpeak, a freely available speech synthesizer, automatically generating phoneme transcription using German pronunciation rules. However, using eSpeak has some drawbacks: Transcribed text is not normalized (e.g. numbers or abbreviations) but spoken as it is given.

Our current model contains 4041 words, which we used to create an n-gram language model. A decoder run on the test set left us with 70.5% sentence errors and a word error rate (WER) of 18.0% with our German corpus. While training, the split was made into appr. 3000 recordings for the training and 800 recordings for test selected randomly. For a speaker-independent acoustic model we will need more data. Currently, we are collecting further data of more speakers to reach the recommended amount of 300 speakers.

6 Conclusion and Outlook

In this paper, we introduced KiSRecord, an open source tool that can be used to record audio for distant speech recognition. We used the audio from a recording session with 58 speakers to train an open source speech recognizer for speech recognition. Our hope is that our software can be used to lower the barriers for other researchers who are in the need to train their own acoustic models in a similar environment, as a variant of crowdsourcing audio in a semi-supervised setting.

Furthermore, we looked into specific details that have to be considered for the best practices of using the Kinect as a microphone array.

In the future, we will continue to add recordings and also start collaborating with other researchers who are also interested in creating their own models. Furthermore, we will have to tune the parameters used for the training and also improve the pronunciation rules to improve recognition accuracy. For the problem of converting text to phoneme-variants we plan to use OpenMary [17] from DFKI to automatically generate the phoneme sequences for the dictionary.

Our corpus of recordings is available upon request. We encourage the community to join our efforts in creating a large-scale corpus of recordings for distant speech recognition.

Acknowledgements

This work was partly supported by the Bundesministerium für Bildung und Forschung (BMBF), Germany within the programme "KMU-innovativ: Mensch-Technik-Interaktion für den demografischen Wandel".

[6] http://www.dev.voxforge.org/projects/de/wiki/espeak2Phones

References

[1] H. Christensen, J. Barker, N. Ma, and P. D. Green, "The CHiME corpus: a resource and a challenge for computational hearing in multisource environments," in *INTERSPEECH*, pp. 1918–1921, 2010.

[2] J. Barker, E. Vincent, N. Ma, H. Christensen, and P. Green, "The PASCAL CHiME speech separation and recognition challenge," *Computer Speech & Language*, vol. 27, no. 3, pp. 621–633, 2013.

[3] A. Batliner, C. Hacker, and E. Nöth, "To Talk or not to Talk with a Computer: On-Talk vs. Off-Talk," *How People Talk to Computers, Robots, and Other Artificial Communication Partners*, p. 79, 2006.

[4] B. Gold, N. Morgan, and D. Ellis, *Speech and audio signal processing: processing and perception of speech and music*. John Wiley & Sons, 2011.

[5] N. Willson, C. Smith, D. Harris, and B. Richter, "Audio with kinect," tech. rep., Dept. Electrical and Computer Engineering, University of Victoria, Aug. 2012.

[6] K.-F. Lee, *Automatic Speech Recognition: The Development of the Sphinx Recognition System*. Springer, 1989.

[7] A. Lee, T. Kawahara, and K. Shikano, "Julius - an open source real-time large vocabulary recognition engine," in *Proc. European Conference on Speech Communication and Technology (EUROSPEECH)*, pp. 1691–1694, 2001.

[8] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz, *et al.*, "The kaldi speech recognition toolkit," in *Proc. ASRU*, pp. 1–4, 2011.

[9] A. Chan, "Do we have a true open source dictation machine?," blog, Carnegie Mellon University, CMU Sphinx Group, July 2005. last accessed on 04/09/2014.

[10] Voxforge.org, "Free speech... recognition (linux, windows and mac) - voxforge.org." http://www.voxforge.org/. accessed 06/25/2014.

[11] A. Gruenstein, I. McGraw, and A. Sutherland, "A self-transcribing speech corpus: collecting continuous speech with an online educational game," in *SLaTE Workshop*, 2009.

[12] T. Schultz, A. Black, S. Badaskar, M. Hornyak, and J. Kominek, "SPICE: Web-based Tools for Rapid Language Adaptation in Speech Processing Systems," in *Proceedings of INTERSPEECH*, 2007.

[13] I. Lane, A. Waibel, M. Eck, and K. Rottmann, "Tools for collecting speech corpora via mechanical-turk," in *Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk*, pp. 184–187, Association for Computational Linguistics, 2010.

[14] Y. Li, T. Banerjee, M. Popescu, and M. Skubic, "Improvement of acoustic fall detection using kinect depth sensing," in *Engineering in Medicine and Biology Society (EMBC), 2013 35th Annual International Conference of the IEEE*, pp. 6736–6739, IEEE, 2013.

[15] S. H. K. Parthasarathi, S.-Y. Chang, J. Cohen, N. Morgan, and S. Wegmann, "The blame game in meeting room ASR: An analysis of feature versus model errors in noisy and mismatched conditions," in *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*, pp. 6758–6762, IEEE, 2013.

[16] J. DeVeth, B. Cranen, and L. Boven, "Acoustic features and distance measure to reduce vulnerability of asr performance due to the presence of a communication channel and/or background noise," in *Robustness in language and speech technologies*, pp. 9–45, Springer, 2001.

[17] M. Schröder and J. Trouvain, "The German text-to-speech synthesis system MARY: A tool for research, development and teaching," *International Journal of Speech Technology*, vol. 6, no. 4, pp. 365–377, 2003.