MULTICHANNEL AUDIO FRONT-END FOR FAR-FIELD AUTOMATIC SPEECH RECOGNITION

Amit Chhetri, Philip Hilmes, Trausti Kristjansson,
Wai Chu, Mohamed Mansour, Xiaoxue Li, Xianxian Zhang

Amazon Inc., Sunnyvale, CA, USA

ABSTRACT

Far-field automatic speech recognition (ASR) is a key enabling technology that allows untethered and natural voice interaction between users and Amazon Echo family of products. A key component in realizing far-field ASR on these products is the suite of audio front-end (AFE) algorithms that helps in mitigating acoustic environmental challenges and thereby improving the ASR performance. In this paper, we discuss the key algorithms within the AFE, and we provide insights into how these algorithms help in mitigating the various acoustical challenges for far-field processing. We also provide insights into the audio algorithm architecture adopted for the AFE, and we discuss ongoing and future research.

***Index Terms*—** Beamforming, far-field, AFE, deep neural networks, ASR, Amazon Echo.

1. INTRODUCTION

The launch of Amazon Echo propelled the use of far-field ASR in the consumer electronics space, as it enabled an untethered and natural voice interaction by allowing users to interact with the device from several meters away. The first version of Echo device allowed users to ask questions related to weather, traffic, news, and to stream audio content from the device. Since its launch, the Echo family of devices and their functionalities have grown considerably; users can now request the device to stream videos, make voice calls, pair Echo devices to their existing home audio systems, and so on.

A user query for Amazon Echo is typically phrased as: "Alexa, what is the time?", where the first word *Alexa* is called the *wake-word* (WW) (to get the device's attention), and the remaining part of the utterance is termed as the *voice command*. One of the primary challenges for Echo devices to scale to millions of households was to cope up with the unknown acoustical conditions in users homes, which include varying levels of acoustic echo, noise and reverberation; the acoustic interference in the room can significantly impair the spoken utterance. While significant progress has recently been made in the ASR and WW recognition performance by using deep neural networks (DNNs) in acoustic modeling (AM) [1–3], their performance can be further improved with a well-designed AFE [4–6]. Echo devices use a highly specialized multi-channel (or multi-microphone) AFE, which significantly improves the ASR and WW performance under a variety of acoustic conditions. Note that for the rest of the paper we will use the term *smart-speaker* instead of Amazon Echo to avoid confusion with the term 'acoustic echo'.

In addition to the acoustical challenges, we also need to be cognizant of real-time and practical constraints that impact the customer experience. For example, the latency constraint dictates that the device should respond almost instantly when the WW is spoken, or that bandwidth constraints may limit us from transmitting large amounts of data from the device to cloud infrastructure. In this paper, we provide details on the key design tenets to combat the acoustic and practical challenges and constraints.

This paper is organized as follows. In Section 2, we present the far-field acoustic environment for smart speakers. In Section 3, we present the overall system model and its key metrics. In Section 4, we provide insights into the various approaches (signal processing and DNN) employed to mitigate the acoustic challenges. Section 5 provides experimental results, and we provide conclusions in Section 6.

2. FAR-FIELD ACOUSTIC ENVIRONMENT

Figure 1 depicts the far-field acoustic environment for a smart-speaker system. Here, the audio content is played out of an L-loudspeaker system. The user's speech is captured by an array of M microphones; the room reflections cause the recorded speech to be reverberant. The various noise sources in the room generate the acoustic ambient noise. Lastly, the audio content played out of the device's loudspeakers is also recorded at the microphones as multi-channel acoustic echo (MAE). The ambient noise, reverberation, and MAE components are lumped together as *acoustic interference*. The signal acquired at the mth microphone can be expressed as:

$$x_m(n) = g_m(n) * s(n) + \sum_{l=1}^{L} h_{l,m}(n) * u_l(n) + v_m(n), \quad (1)$$

where n denotes the discrete time index, $*$ denotes convolution, $s(n)$ denotes the clean speech signal, $u_l(n)$ denotes the

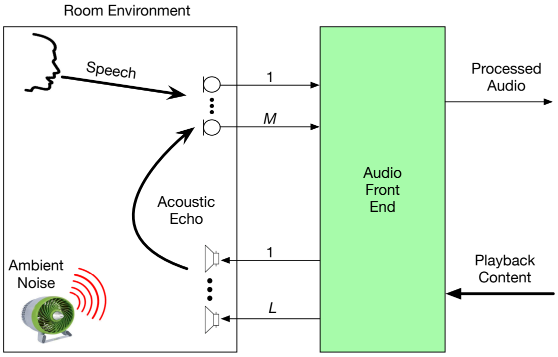

Fig. 1. Acoustic environment for a smart-speaker system.

playback signal for the lth channel, $v_m(n)$ denotes the noise at the mth microphone, and g_m and $h_{l,m}$ denote the acoustic impulse responses between the mth microphone and the speech source and the lth loudspeaker, respectively.

The AFE's goal is to condition the playback signal for optimal sound quality, and to mitigate the acoustic interference in order to provide the highest quality of audio stream to the ASR engine; an ideal AFE output signal is denoted by $y(n) = s(n - n_d)$, where n_d is AFE's processing delay.

3. OVERALL SYSTEM-LEVEL MODEL AND KEY METRICS

A smart-speaker's response time to the wake-word is a very important metric as it strongly ties to the user experience. In addition, users also want the device to respond to their queries with a high accuracy. In order to meet latency, bandwidth, and performance constraints, we adopt the system model depicted in Figure 2. The system processing is divided into two parts: (a) on-device processing, which comprises of the AFE algorithms and the WW (wake-word) engine, and (b) processing on Amazon's cloud infrastructure, which hosts the ASR, natural language understanding (NLU), and the text-to-speech (TTS) engines, along with other Alexa services. As noted, the microphone data is first processed by the AFE algorithms, and its output is sent to the WW engine. If Alexa keyword is detected, the WW engine streams the user utterance to the cloud where the ASR and NLU engines work in tandem to decode the spoken utterance. Thereafter, Alexa's response (to the decoded utterance) is played through the device's loudspeaker for the user. In the following sections, we will mainly focus on the AFE algorithms.

Although each algorithm within the AFE is driven by its own metrics, there are four global metrics on which all AFE algorithms have been optimized. These are: (a) Word-error-rate (WER), which is a key metric for the ASR engine and it is defined as the ratio of the decoding errors (insertion, deletion, and substitution) and the total number of valid words [4], (b) False-rejection-rate (FRR), which is a key metric for the WW engine and it measures the percentage of missed WW commands, (c) latency, which is defined as the processing delay that the AFE introduces in both the playback and capture paths, and (d) computational cost of the algorithms.

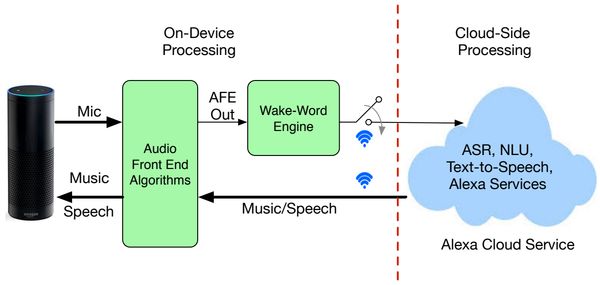

Fig. 2. Block diagram of the overall system.

4. AUDIO FRONT-END

Figure 3 depicts the block diagram for the AFE architecture. For the microphone capture path, we use the subband processing framework, which helps us to achieve a desirable algorithmic performance for a low computational cost [7]. Audio processing occurs on a frame-by-frame basis; the shaded blocks receive an input audio frame and provide an output audio frame, while the non-shaded blocks process an audio frame to generate a system-state variable. The microphone signals are first processed by an analysis filterbank to generate the subband samples. Next, we apply pre-processing on the subband samples that include pre-emphasis filters and delay correction. Next, the spatial processing and multi-channel AEC (MCAEC) block provides further suppression of ambient noise and MAE. Thereafter, we process the audio frames through a post filtering stage, which further helps in improving the signal-to-noise-ratio (SNR). Lastly, the audio frames are synthesized back into time-domain before being sent to the WW engine.

Spatial processing algorithms such as beamformers need an estimate of the user's bearings (i.e., *look-direction*) w.r.t. the device . For this, we use source localization algorithms to determine the likely direction of an active user. We also make use of a sophisticated system-state control (SSC) module, which takes into account the various system states (e.g. talker/playback is active, Alexa is responding, noise conditions, etc.), which are then used to control the various audio algorithms. Lastly, the AFE also receives audio content such as music and speech (e.g. Alexa response) from the cloud, which is processed by playback enhancement algorithms (for optimal sound experience) before being sent to the device's loudspeakers. In the following, we present the major AFE algorithms in more details.

ISBN 978-90-827970-1-5 © EURASIP 2018

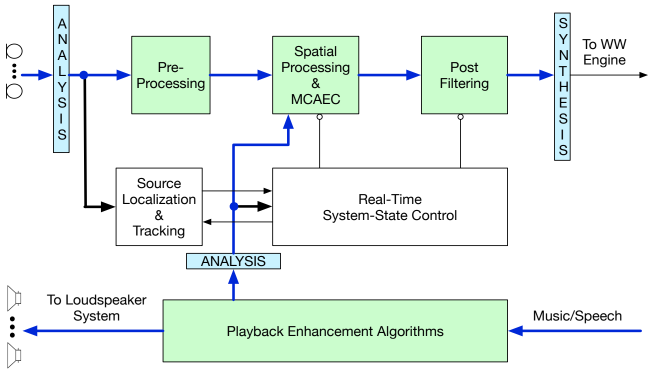
ANALYSIS
Pre-Processing
Spatial Processing & MCAEC
Post Filtering
SYNTHESIS
To WW Engine
Source Localization & Tracking
Real-Time System-State Control
ANALYSIS
To Loudspeaker System
Playback Enhancement Algorithms
Music/Speech

Fig. 3. Block diagram for the AFE architecture.

4.1. Spatial Processing with Microphone Arrays

The microphone array on a smart-speaker is a key enabler of far-field ASR. Array signal processing in itself is a well established field with a focus towards narrowband signal processing, for which it is relatively easier to design the array configuration. However, designing microphone arrays for speech processing is more challenging because speech is a wideband signal that spans several octaves [4]. For smart-speakers, the microphone arrays are designed through acoustic array principles and the associated algorithms, and they are optimized for the device's form-factor.

Beamforming with microphone arrays allows us to optimally combine the signals of multiple microphones in order to enhance the speech arriving from the look-direction while suppressing noise arriving from other directions. For AFE, we have investigated both signal-processing and DNN-based beamforming approaches, which are described next.

4.1.1. Signal Processing-Based Beamforming

One of the most generic forms of beamforming is the filter-and-sum (F&S) structure, where the microphone signals $x_m(n)$ are processed by the filters $w_m(n)$ and then summed together to generate the beamformer output $y(n)$:

$$y(n) = \sum_{m=1}^{M} w_m(n) * x_m(n). \qquad (2)$$

The F&S processing can be broadly divided into two categories: (a) fixed beamformer (FBF), where w_m's are usually optimized offline for a given look-direction, and they are signal and time-invariant, and (b) adaptive beamformer (ABF), where the w_m's are both time and signal dependent.

The FBF is typically designed by posing the beamformer design as a constrained optimization problem (e.g. super-directive beamformer and its variants [4, 8]). For ABF, the filters w_m's are optimized in real-time depending on the signal conditions. For example, in the well known minimum variance distortionless response (MVDR) beamformer, one needs to continuously update $w_m(n)$'s based on the estimated noise and signal statistics [8]. The constrained optimization formulation of MVDR problem can be converted into an unconstrained one by using the generalized sidelobe canceler (GSC) framework [8]. Variants of the standard GSC algorithm exist like the robust adaptive beamformer [5].

The smart-speaker uses beamforming algorithms that employ auxillary SSC and sound source localization (SSL) algorithms in order to adapt to, and mitigate a variety of real-life challenging noise conditions. These algorithms have been designed using several hundred hours of real-world noisy speech corpus, and their parameters have been tuned to achieve optimal WER and FRR performance (results provided in Section 5). Further, they have been deployed on a variety of microphone arrays, and they are being continuously improved with new speech corpora.

4.1.2. DNN-Based Beamforming

DNNs have recently been deployed successfully for a range of tasks including speech recognition and wake word modeling [1, 2]. They have also been gaining popularity on front-end algorithms like beamforming. For example, there have been two notable approaches on the application of DNN to beamforming. The first approach is to estimate the parameters of a beamformer directly [3]. The second approach focuses on using DNNs to classify the time-frequency tiles of the signal into speech or interference, which is then used to estimate the parameters of a traditional beamformer such as MVDR. For AFE processing, we are pursuing the application of DNN on SSL, intelligent system-state estimation and control, and spatial filtering algorithms.

For example, in [9] we present a neural network based approach to two-channel beamforming. Single and cross-channel spectral features were extracted to form a feature map for each utterance. A large neural network composed of a convolutional neural network (CNN), a long short-term memory (LSTM) network, and a fully-connected DNN was employed to estimate frame-level speech and noise spectral masks simultaneously. One mask is estimated for the speech and another for the interference. Based on these masks, cross-power spectral density (CPSD) matrices were estimated and the coefficients of the MVDR beamformer were computed. A second smaller DNN was used to tune the phase in the estimated steering vectors towards the target look direction. Our results show that the proposed methods leads to a 21% relative WER reduction over recent state-of-the-art systems in the literature (refer [9, 10] for more details).

4.2. Multi-Channel Acoustic Echo Cancellation

The smart-speaker system can be used to play audio content over both internal loudspeakers and from external loudspeakers connected to the device over line-out (wired connection)

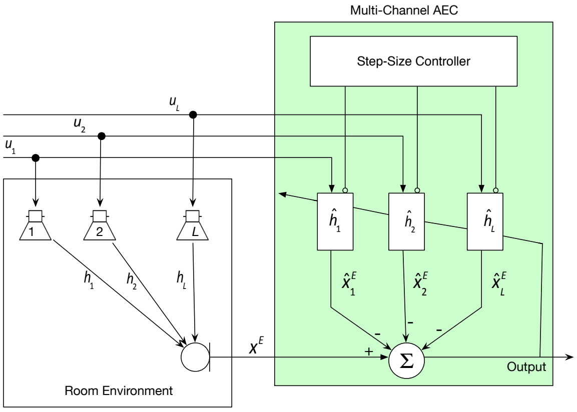

Fig. 4. Block diagram for the MCAEC algorithm.

or bluetooth interfaces. The loudspeaker output is captured by the microphones as MAE, which from (1) can be expressed as $x^E = \sum_{l=1}^{L} h_{l,m}(n) * u_l(n)$. The MAE component is the most dominant acoustic interference for smart-speakers, and we can make use of the MCAEC algorithm shown in Figure 4 to mitigate the MAE component [11]. Here, the playback signals u_l's are used to estimate the acoustic echo paths h_l's, and hence the MAE estimate, which is subtracted from the microphone signal to obtain the MCAEC output. This output signal is also used to update the filters $\hat{h}_l$'s with a time-varying step-size that is provided by a step-size controller.

At a system level, the interaction between the MCAEC and ABF algorithms poses some challenges. Firstly, we need to decide whether the MCAEC should precede or follow the ABF. If the MCAEC precedes the ABF, one can achieve better source localization performance and relieve the MCAEC adaptive filters from tracking the beamformer variations. However, a drawback is that one would need an instance of MCAEC for each microphone, which is computationally expensive. Another challenge is the decision logic to control the adaptive filters for the ABF and MCAEC algorithms depending on the signal conditions. This topic is discussed in details in [6]; for the AFE, we have optimized the system architecture by taking into consideration the computational resources, the microphone array, and the performance requirements.

4.3. Sound Source Localization

The knowledge of user's look-direction is very important for effective beamforming; for Echo products, we need to estimate the look-direction from the microphone array signals. One of the well-known and robust SSL algorithm is the steered response power (SRP) algorithm [12, 13]. Our analysis and experiments indicate that good accuracy is achievable for a moderate computational cost. The algorithm is based on computing the power of signals arriving from all directions of interest; by doing so, it is possible to identify the strongest sources surrounding the array, and thereby, knowing the directions of such sounds. Figure 5 shows the SRP response for the case where the microphone array is surrounded by two sound sources. We can clearly see the two well-formed peaks in the SRP response.

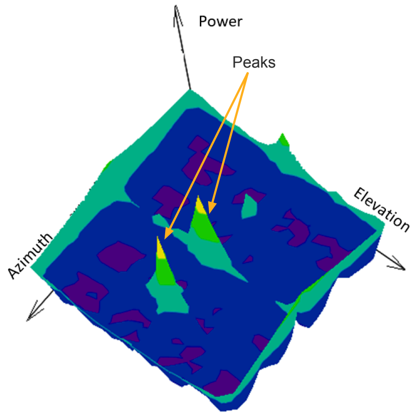

Fig. 5. The steered response power as a function of azimuth and elevation for an example where two sources are present.

By analyzing the SRPs as a function of time, it is possible to identify the sources with the highest power, and to derive tracks containing the directions of sound sources as a function of time. Numerous methods can be used to further smooth the track leading to more concise parameters of the sound source [14]. For multiple sound sources, the information of the tracks can be used to guide the beamformer so as to simultaneously listen to different directions surrounding the microphone array.

4.4. Post-Filtering

The output of the 'Spatial Processing & MCAEC' module may contain residual echo and noise. This usually happens because of loudspeaker nonlinearities or because the noise and speech sources are positioned close to each other. In this case, post filtering algorithms may be used to apply time-frequency processing to discriminate between desired speech and interference. In the past, the post-filtering algorithms have been designed by framing the problem as multi-channel Wiener filter [4, 8]; however, in the context of AFE, the post-filter should jointly consider both residual echo and ambient noise as interference. The parameters of the post-filter are optimized over large corpus of noisy speech signal, and a well-designed post-filter that works in tandem with the upstream algorithms can further improve the AFE performance.

5. RESULTS

The AFE is highly customizable and it fits within the computational budget of every Echo device. Furthermore, the latency introduced by the AFE is a small fraction of the time it takes to say 'Alexa'. For WER evaluation, we utilized an internal dataset with 39027 utterances that were collected from 49 participants under a variety of ambient noise conditions. Table 1 provides the relative WER reduction offered by the AFE algorithms over raw microphones for three different SNR conditions (low, medium, and high). For the FRR evaluation, we utilized an internal dataset with 92400 WW instances that were captured from 330 participants under a variety of device playback conditions. Table 2 provides the relative FRR reduction offered by the AFE algorithms over raw microphones for three different playback conditions (low, medium, high); the sound pressure levels (in dBC) at 1 m away from the device are provided in the table. From these results, we note that AFE significantly helps in improving the WER and FRR performance.

Table 1. Relative WER reduction through AFE processing.

SNR (dB)	Utterances	Relative WER Reduction (%)
[-20, 4) - Low	19348	45.5
[4, 8) - Medium	12982	27.5
[8, 30] - High	6697	17.5
Overall	39027	39.7

Table 2. Relative FRR reduction through AFE processing.

Playback Level 1 m away (dBC)	WW Instances	Relative FRR Reduction (%)
[60, 65] - Low	18480	92.4
[70, 75] - Medium	36960	83.9
[80, 85] - High	36960	60.2
Overall	92400	75.7

6. CONCLUSIONS

Smart-speakers are being used in increasingly challenging environments, and the AFE has to keep up with the new demands. While significant improvements have been achieved in ASR performance, we continue to pursue new ideas to improve the AFE's performance. For example, we are pursuing improvements in performance of DNN in beamforming, MCAEC, and SSL algorithms. Robustness is a key criterion for the AFE algorithms, and we continue to explore better models to improve the robustness of AFE by adaptively learning the environmental parameters.

REFERENCES

[1] Sri Garimella, Arindam Mandal, Nikko Strom, Björn Hoffmeister, Spyros Matsoukas, and Sree Hari Krishnan Parthasarathi, "Robust i-vector based adaptation of DNN acoustic model for speech recognition," in *Sixteenth Annual Conference of the International Speech Communication Association*, 2015.

[2] Sankaran Panchapagesan, Ming Sun, Aparna Khare, Spyros Matsoukas, Arindam Mandal, Björn Hoffmeister, and Shiv Vitaladevuni, "Multi-task learning and weighted cross-entropy for DNN-based keyword spotting.," in *INTERSPEECH*, 2016, pp. 760–764.

[3] Bo Li, Tara N Sainath, Ron J Weiss, Kevin W Wilson, and Michiel Bacchiani, "Neural network adaptive beamforming for robust multichannel speech recognition.," in *INTERSPEECH*, 2016, pp. 1976–1980.

[4] Matthias Wölfel and John McDonough, *Distant speech recognition*, John Wiley & Sons, 2009.

[5] Osamu Hoshuyama and Akihiko Sugiyama, "Robust adaptive beamforming," in *Microphone arrays*, pp. 87–109. Springer, 2001.

[6] Walter L Kellermann, "Acoustic echo cancellation for beamforming microphone arrays," in *Microphone Arrays*, pp. 281–306. Springer, 2001.

[7] Parishwad P Vaidyanathan, *Multirate systems and filter banks*, Pearson Education India, 1993.

[8] Wolfgang Herbordt, *Sound capture for human/machine interfaces*, vol. 315, Springer Science & Business Media, 2005.

[9] Yuzhou Liu, Anshuman Ganguly, Krishna Kamath, and Trausti Kristjansson, "Neural network based time frequency masking and steering vector estimation for two-channel mvdr beamforming.," in *Acoustics, Speech and Signal Processing (ICASSP), 2018 IEEE International Conference on*, 2018.

[10] Jahn Heymann, Lukas Drude, and Reinhold Haeb-Umbach, "Neural network based spectral mask estimation for acoustic beamforming," in *Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on*. IEEE, 2016, pp. 196–200.

[11] Jacob Benesty and Yiteng Huang, *Adaptive signal processing: applications to real-world problems*, Springer Science & Business Media, 2013.

[12] Joseph Hector DiBiase, *A high-accuracy, low-latency technique for talker localization in reverberant environments using microphone arrays*, Brown University Providence, 2000.

[13] Jacek P Dmochowski, Jacob Benesty, and Sofiene Affes, "A generalized steered response power method for computationally viable source localization," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 8, pp. 2510–2526, 2007.

[14] Subhash Challa and Mark R. Morelande, *Fundamentals of object tracking*, Cambridge University Press, 2011.