Deep Learning-based Acoustic-echo Cancellation

Amir Ivry Mark

Deep Learning-based Acoustic-echo Cancellation

Research Thesis

Submitted in partial fulfillment of the requirements

for the degree of Doctor of Philosophy

Amir Ivry Mark

Submitted to the Senate

of the Technion — Israel Institute of Technology

Kislev 5784 Haifa November 2023

This research was carried out under the supervision of Prof. Israel Cohen and Dr. Baruch Berdugo in the Faculty of Electrical and Computer Engineering.

The author of this thesis states that the research, including the collection, processing and presentation of data, addressing and comparing to previous research, etc., was done entirely in an honest way, as expected from scientific research that is conducted according to the ethical standards of the academic world. Also, reporting the research and its results in this thesis was done in an honest and complete manner, according to the same standards.

List of Publications

Journal Papers

- A. Ivry, I. Cohen, and B. Berdugo. Voice activity detection for transient noisy environment based on diffusion nets. *IEEE Journal of Selected Topics in Signal Processing*, 13(2):254–264, 2019.

- A. Ivry, B. Berdugo, and I. Cohen. A user-centric approach for deep residual-echo suppression in double-talk. submitted to *IEEE Transactions of Acoustic, Speech, and Language Processing*, 2023.

Conference Papers

- A. Ivry, B. Berdugo, and I. Cohen. Evaluation of deep-learning-based voice activity detectors and room impulse response models in reverberant environments. In *Proc. International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 406–410, 2020.

- A. Ivry, B. Berdugo, and I. Cohen. Deep residual echo suppression with a tunable tradeoff between signal distortion and echo suppression. In *Proc. International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 126–130, 2021.

- A. Ivry, B. Berdugo, and I. Cohen. Nonlinear acoustic echo cancellation with deep learning. In *Proc. Interspeech*, pages 4773–-4777, 2021.

- A. Ivry, B. Berdugo, and I. Cohen. Objective metrics to evaluate residual-echo suppression during double-talk. In *Proc. Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, pages 101–105, 2021.

- A. Ivry, B. Berdugo, and I. Cohen. Off-the-Shelf deep integration for residual-Echo suppression. In *Proc. International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 746–750, 2022.

- A. Ivry, B. Berdugo, and I. Cohen. Deep adaptation control for acoustic echo cancellation. In *Proc. International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 741–745, 2022.

- A. Ivry, B. Berdugo, and I. Cohen. Objective metrics to evaluate residual-Echo suppression during double-talk in the Stereophonic Case. In *Proc. Interspeech*, pages 5348–5352, 2022.

- A. Ivry, B. Berdugo, and I. Cohen. Deep adaptation control for stereophonic acoustic echo cancellation. accepted to *Proc. Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, 2023.

Acknowledgements

I would like to thank Prof. Israel Cohen, my principal advisor, for always believing in me and helping me overcome many obstacles during my journey. His expertise, guidance and support have made me a significantly better researcher.

I would also like to thank Dr. Baruch Berdugo, my secondary advisor, for providing me with technical aim and expert insights with constant availability, especially during the early stage of my journey.

To my late father Reuven, who recognized my passion for math and science from a very early age - and tirelessly dedicated his time and care to provide me with all the conditions to succeed. Without you, I would not be who I am today.

To my mother Eva, for celebrating with me during happy times, and for being present during hardships. Thanks to your support, I knew how to cross many crossroads during my journey.

To my big sisters Yael and Michal, for giving me room to express who I am while growing up, increasing my self-confidence, and being there in big moments.

Above all, I thank my dear wife Noy and baby boy Nitay. During good times and hard times, my strength to push through the biggest challenges came from you. This achievement is much more meaningful because you are here by my side.

Contents

List of Tables

List of Figures

Abstract

This dissertation focuses on deep learning-based systems for acoustic-echo cancellation in monophonic setups, with emphasis on lean implementations that meet the computational and timing standards of modern hands-free communication platforms. A complementary part of this thesis concerns stereophonic acoustic-echo cancellation and voice-activity detection.

In recent years, face-to-face meetings have been often replaced by virtual conferencing, especially in office environments. One representative scenario is a standard conference-call between two ends of a virtual conversation; a near-end conference room and a far-end home environment. In this setup, the far-end participants often suffer both of deterioration of the near-end speech intelligibility and of hearing their own echoing voices. This may cause lost of information, fatigue, and decrease in work productivity in an era that heavily relies on remote communication.

This thesis introduces deep learning-based solutions that handle non-linear acoustic-echo cancellation, linear acoustic-echo cancellation, residual-echo suppression, and objective measurements of acoustic echo.

As miniaturization of electrical components becomes more dominant, the more non-linear effects occur in acoustic-echo cancellation systems. Specifically, the relation between the far-end speech and its reverberant echo as perceived in the near-end microphone is often non-linear. We developed a system that tracks and estimates the non-linearities that modern hardware applies to the far-end signal, and fed the non-linear estimate of the far-end to the linear acoustic-echo cancellation filter as reference.

The linear acoustic-echo cancellation system experiences misalignment between the real and estimated near-end linear echo-path from the loudspeaker to the microphone,

mainly in double-talk scenarios and during echo-path changes. We offer a solution that mitigates both these gaps and offer on-the-fly adaptation control of the step-size that governs the behavior of the linear acoustic-echo cancellation filter.

Residual-echo components frequently remain after the non-linear and linear acoustic-echo cancellation stages due to imperfect algorithms and challenging acoustic scenarios. We proposed a lean residual-echo suppressor with a novel design parameter that allows for a desirable and practical trait; a dynamic control between the echo suppression and the speech distortion levels of the residual-echo suppression system.

Existing objective metrics for acoustic-echo cancellation have inherent ambiguity, especially in double-talk, and may present a similar value if echo levels were low and speech distortion was high, and vice versa. We have shown that these metrics experience very low correlation with subjective human ratings, and introduced two alternative objective performance metrics to assess the echo suppression level and the desired speech distortion level.

We also regard stereophonic acoustic-echo cancellation scenarios that include two loudspeakers and two microphones in both the near-end and far-end. We successfully projected two concepts we introduced in the monophonic case; an adaptation control framework that allows to better handle double-talk scenarios and echo-path changes, and an extended version of our monophonic objective evaluation metrics.

Voice activity detection is an integral part in many speech-based systems, including ones that feature acoustic-echo cancellation. We achieved state-of-the-art performance in real acoustic environments of reverberation, noises, and transients by using inherent geometric structures that distinguish speech from non-speech segments and by utilizing a deep encoder-decoder for speech classification.

Abbreviations and Notations

Abbreviations

AEC	:	Acoustic-echo cancellation
AECMOS	:	Acoustic-echo cancellation mean-opinion score
AUC	:	Area under curve
CPU	:	Centeral processing unit
dB	:	Decibels
DED	:	Deep encoder-decoder
DNSMOS	:	Deep noise-suppression mean-opinion score
DVSS	:	Deep VSS
DSML	:	Desired-speech maintained level
DM	:	Diffusion maps
D/A	:	Digital-to-analog converter
ERLE	:	Echo-return loss enhancement
ENR	:	Echo-to-noise-ratio
ESR	:	Echo-to-speech-ratio
FLOPS	:	Float-point operations per second
GRU	:	Gated recurrent unit
GFLOPS	:	Giga float-point operations per second
GHz	:	Giga-hertz
GPU	:	Graphic processing-unit
Hz	:	Hertz
KB	:	Kilo-bytes
MB	:	Mega-bytes

MFLOPS	:	Mega float-point operations per second
MHz	:	Mega-hertz
MFCC	:	Mel-frequency cepstral coefficient
ms	:	Milliseconds
NN	:	Neural network
NNVSS	:	Neural-network VSS
NL	:	Non-linear
NLM	:	Non-linear model
NPVSS	:	Non-parametric VSS
NLMS	:	Normalized least-mean squares
PCC	:	Pearson correlation coefficient
PESQ	:	Perceptual evaluation of speech quality
PLU	:	Piece-wise linear unit
PSLT	:	Positive saturating linear transfer
ROC	:	Receiver operating characteristic
RNN	:	Recurrent neural network
ReLU	:	Rectified linear unit
RES	:	Residual-echo suppression
RESL	:	Residual-echo suppression level
RDE	:	RESL-DSML Estimator
RIR	:	Room impulse-response
s	:	Seconds
STFT	:	Short-time Fourier Transform
SVSS	:	Sigmoid VSS
SNLMS	:	Sign NLMS
SAR	:	Signal-to-artifacts ratio
SDR	:	Signal-to-distortion ratio
SER	:	Signal-to-echo-ratio
SNR	:	Signal-to-noise-ratio
SRCC	:	Spearman's rank correlation coefficient

SE	:	Speech enhancement
SS	:	Speech separation
std	:	Standard deviation
SDSML	:	Stereophonic DSML
SENR	:	Stereophonic ENR
SESR	:	Stereophonic ESR
SRESL	:	Stereophonic RESL
SSDR	:	Stereophonic SDR
SSER	:	Stereophonic SER
SSNR	:	Stereophonic SNR
SVM	:	Support vector machines
TN	:	True negative
TP	:	True positive
URES	:	User-centric RES
UOP	:	User operating-point
VSS	:	Variable step-size
VAD	:	Voice activity detection

Notations

$\lvert \cdot \rvert$	:	absolute value operator
$(\cdot)^H$	:	adjoint operator
ϵ	:	A priori adaptation error
e	:	A posteriori adaptation error
$\tilde{s}$	:	Bias-free compensated version of s
$(\cdot)^*$	:	conjugate operator
$*$	:	Convolution operator
$\widehat{s}_i$	:	Desired-speech estimate by RES_i
Δ_{D}	:	Deviation of $\widehat{\text{D}}_{\widehat{i}}$ from D
Δ_{R}	:	Deviation of $\widehat{\text{R}}_{\widehat{i}}$ from R
$\widehat{\text{D}}_i$	:	DSML estimate by RDE_i
$\widehat{\text{D}}_{\widehat{i}}$	:	DSML estimate by the chosen RDE model instance $\widehat{i}$
D	:	DSML value
I	:	Electrical signal received by the power amplifier
$\hat{(\cdot)}$	:	Estimation of the value inside the parenthesis
$E(\cdot)$	:	expectation operator
x	:	Far-end signal
x_L	:	Far-end signal as captured by left far-end microphone in the stereophonic case
x_R	:	Far-end signal as captured by right far-end microphone in the stereophonic case
f	:	Frequency bin
$\mathbb{I}_{(\cdot)}$	:	Indicator function of the term inside the parenthesis
$\langle \cdot, \cdot \rangle$	:	Internal product between two vectors
$\hat{g}$	:	Gain factor to create $\tilde{s}$ from s
g	:	Gain of neural network
m_L	:	Left channel of the near-end microphone signal in the stereophonic case
H	:	Magnetic signal strength around the loudspeaker voice coil
m	:	Meters
m	:	Near-end microphone signal
w	:	Near-end noise signal as captured by m

y	:	Near-end reverberant echo signal as captured by m
h	:	Near-end room impulse response
s	:	Near-end speech signal as captured by m
s_L	:	Near-end speech signal as captured by m_L in the stereophonic case
s_R	:	Near-end speech signal as captured by m_R in the stereophonic case
r	:	Noisy residual-echo estimate
x^{NL}	:	Non-linearly distorted far-end signal
$\mathbf{x}_{\mathrm{L}}^{\mathrm{NL}}$	:	Non-linearly distorted version of x_L
$\mathbf{x}_{\mathrm{R}}^{\mathrm{NL}}$	:	Non-linearly distorted version of x_R
$\mathcal{D}$	:	Normalized misalignment
$(\,\cdot\,)^*$	:	Optimal value to the one inside the parenthesis
RDE_i	:	RDE model instance with index i
$\widehat{\mathrm{R}}_i$	:	RESL estimate by RDE_i
$\widehat{\mathrm{R}}_{\widehat{i}}$	:	RESL estimate by the chosen RDE model instance $\widehat{i}$
R	:	RESL value
RES_i	:	RES model instance with index i
y_L	:	Reverberant echo signal as captured by m_L in the stereophonic case
y_R	:	Reverberant echo signal as captured by m_R in the stereophonic case
m_R	:	Right channel of the near-end microphone signal in the stereophonic case
h_{LL}	:	RIR from the left loudspeaker to m_L in the stereophonic case
h_{LR}	:	RIR from the left loudspeaker to m_R in the stereophonic case
h_{RL}	:	RIR from the right loudspeaker to m_L in the stereophonic case
h_{RR}	:	RIR from the right loudspeaker to m_R in the stereophonic case
sgn	:	sign operator
δ	:	Small regularization parameter
μ	:	Step-size of the adaptation process
S	:	STFT representation of s
n	:	Time index
TH_{R}	:	Tolerance of deviation of $\widehat{\mathrm{R}}_i$ from R
TH_{D}	:	Tolerance of deviation of $\widehat{\mathrm{D}}_i$ from D

$(\cdot)^T$	:	Transpose of the value inside the parenthesis
α	:	Tunable design-parameter
$\sigma^2_{(\cdot)}$	:	Variance of the value inside the parenthesis
Δ	:	Voice coil displacement
$\|\cdot\|_2$	:	ℓ_2-norm operator
j	:	$\sqrt{-1}$

Chapter 1

Introduction

1.1 Background and Motivation

This thesis focuses on offering deep learning-based acoustic-echo cancellation solutions for remote conversations between two ends; a near-end and a far-end. In the most basic scenario, a near-end microphone captures three types of acoustic signals; speech from the near-end speaker, reverberant version of a non-linearly distorted far-end speech that is played by a near-end loudspeaker, and environmental and system noises. The acoustic coupling between the near-end loudspeaker and microphone lead to two undesired phenomena in the far-end; deterioration in the intelligibility of the near-end speech, and remaining presence of the far-end echo [SMH95], [BGM+01]. This problem has sprouted numerous studies on acoustic-echo cancellation (AEC) that try to address this challenge by cancelling the far-end echo and preserve the near-end speech undistorted, both with classic signal processing approaches, deep learning methods, and hybrid solutions. However, these systems under-perform in practice due to four main reasons. First is the false assumption of linearity between the far-end signal and the near-end echo. The second would be the suboptimal assessment of the linear near-end echo-path from the loudspeaker to the microphone. Third is the characterization and removal of residual echo after the linear AEC stage has surpassed. The fourth and final challenge is the ambiguous objective performance assessment of AEC systems, especially in double-talk periods.

In recent years, miniaturization of electronic components in hands-free devices,

e.g., smart phones, smart speakers, and wearable devices, caused non-negligible non-linear distortions in the echo path between the far-end signal and the loudspeaker output [BG95a]. Consequently, AEC systems that assume an echo path that is linear often fail in practice [MEB10]. To mitigate this mismatch, various non-linear AEC approaches were proposed to identify the non-linear echo path. The Volterra series showed success in modeling systems with weak non-linearities and memory using non-linear basis functions, while often requiring high computational complexity [GFLBJ03]. A simplified version is given by the block-oriented Hammerstein and Wiener models, which describe non-linear systems without memory and linear systems with memory [SCPU11]. Also, adaptive functional link filters [CSAR$^+$13], Bayesian state-space modeling [ME12], and kernel-based methods [VVARC16] are commonly used for non-linear AEC. Avargel and Cohen considered this problem from a time-frequency point-of-view and applied multiplicative function approximation [AC08], sub-band adaptive filtering [AC09a], and an efficient Volttera series modeling using cross-band terms [AC09b], [AC10]. Neural networks provide an alternative framework for a more accurate non-linear modeling compared to classic approaches [BG95b], [RT98], [Jan04], [ZZ17]. For instance, Malek and Koldovsky [MK16a] estimated the non-linear echo path with a fully-connected neural network that assumes the Hammerstein model, followed by an adaptive linear filter to track the acoustic path. Recently, Halimeh et al. [HHK19] constructed an fully-connected neural network that assumes the Wiener-Hammerstein model and captures both the non-linear and linear echo paths. Despite showing promising results, the performance of these methods is still challenging in real-life scenarios, which may be associated with two of their attributes. First, these models are not accurately designed according to the physical behavior of distortions that modern hands-free devices apply to the far-end signal. Second, they are mostly parametric, i.e., they require that memory lengths and non-linear basis functions are predetermined. E.g., in [GFLBJ03], [SCPU11], the presented models assume a given number of memory taps, and in [MK16a], [HHK19], fixed non-linear activation functions are employed inside the neural network. These drawbacks may produce sub-optimal solutions in real setups.

The mismatch between the real and estimated near-end linear echo-path is rooted in several causes. Preliminary, the real echo-path may have a long reverberation time

and therefore a large number of coefficients are required to accurately express it, which is often a computational burden on classic adaptive filters that utilize smaller number of coefficients. Another reason is that the accuracy of the adaptation process may be subpar due to challenging acoustic conditions, e.g., double-talk, that impede the ability to characterize the echo-path. The last cause is related to the real-life frequent echo-path changes that are hard to track by conventional algorithms and cause re-convergence of the adaptation process. The normalized least mean-square (NLMS) filter is a popular adaptive filter since it is numerically stable and computationally efficient [PCBG15]. The NLMS integrates the normalized step-size parameter that governs the often conflicting fast convergence requirements and low misadjustment. Therefore, it is highly desirable to control the step-size during adaptation in practical scenarios of time-varying echo paths and double-talk. This problem has motivated numerous variable step-size (VSS) related studies. For example, Haubner et al. employed neural networks for near-end estimation [HHB+20], noise estimation[HBEK21], and minimizing the error using adaptation control in the frequency domain [HBK21]. Meier and Kellermann [MK16b] employed a deep neural network that maps statistical features of the far-end and a priori error signals to an analytically derived VSS. A batch of classic approaches includes the NPVSS that adjusts the step-size by reducing the squared error at each instant [BRVT06], the mean error SVSS that applies decomposition of the error into sub-blocks [HA16], and HVSS that estimates the system noise power to control the step-size update [HL11]. However, existing approaches make restricting assumptions in real-life setups, e.g., assuming a linear relationship between the echo and the far-end signals [HHB+20]–[HL11], and adopting a time-invariant echo-path [BRVT06]. In practice, these assumptions result in filter misadjustment and slow convergence rates during echo-path changes [ICB21b]. Also, such methods require tuning parameters that are difficult to control in real-life scenarios. For example, the NPVSS [BRVT06] involves estimating the noise power, which is challenging during double-talk.

Conventional AEC systems do not model non-linearities in the echo path, and generally introduce a mismatch between true and estimated echo paths during convergence and re-convergence [BMS98]. This results in residual echo that must be suppressed by a dedicated residual-echo suppression (RES) system. Deep learning has occupied a

major role in RES studies and showed enhanced performance compared to traditional methods [HK20], [FEKL20]. A recent study exploited long short-term memory (LSTM) networks to jointly obtain echo cancellation and to suppress noises and reverberations [CSVH19]. Lee et al. [LSK15] cascaded a fully-connected neural network after a linear AEC system and evaluated the objective gain between the spectra amplitudes of the near-end and canceler output signals. Lei et al. [LCH$^+$19] exploited past and future temporal context to map the microphone and reference far-end signals to the desired speaker via a fully-connected neural network. Lately, deep learning and classic methods were jointly utilized in [MHZS20] and [ZTW19], where the latter activated convolutional recurrent networks to evaluate the real and imaginary parts of the near-end signal spectrogram.

Despite numerous efforts to enhance the various AEC pipeline components, objective performance metrics have remained ambiguous and biased. Human perception of speech quality is optimally evaluated using human subjective evaluation [RBP$^+$19]. Lately, the objective Deep noise-suppression mean-opinion score (DNSMOS) metric has been proposed to estimate human ratings and has shown great accuracy [RGC21]. Regarding the task of RES, speech quality during double-talk is traditionally evaluated using the objective signal-to-distortion ratio (SDR) metric [VGF06], e.g., in [CSVH18, DDBW19, PP20, CXCL20, Fan20b, Fan20a]. Unfortunately, the SDR is affected by both desired-speech distortion and residual-echo presence, which renders it unreliable in predicting the DNSMOS and unreliable in predicting human perception of speech quality [RGC21].

This thesis also concerns the case of stereophonic AEC (SAEC), in which the near-end microphones may capture three types of acoustic signals; the desired speech, additional noises, and reverberant echoes. The echoes are non-linearly distorted versions of the far-end signal played by loudspeakers and reverberate to the microphones via echo paths [SMH95]. These echoes may impede conversation intelligibility as perceived by the far-end participant. The SAEC task is two-fold; tracking the near-end echo-paths and subtracting them from the microphones signals, and communicating the undistorted desired-speech signal to the far-end [BMS98]. This thesis considers the specific case of two loudspeakers and two microphones both in the far-end and the near-end

parts of the conversation, and addresses two of the four challenges raised in the monophonic AEC case; the suboptimal linear echo-paths estimations, and the ambiguous and biased performance evaluation of SAEC systems.

In this SAEC case, the echo paths between a pair of loudspeakers and a pair of microphones are modeled by adaptive filtering. The echo paths are converted into acoustic-echo approximations that are subtracted from the microphones [SMH95, BMS98]. Double-talk segments are most challenging, since the echoes overlap with desired speech. Various studies tried to cope with it by preserving the speech and removing the echoes [SBP+13, PBC14, CRPP12, KS17, MHB01, WQW10, RCP+10, GT98]. In practice, however, echo paths are not estimated accurately, e.g., when the adaptive filter has not yet converged [BGM+01]. Therefore, a RES system must succeed the SAEC system to eliminate the echoes. Subjective human evaluation is currently the most accurate assessment of human perception for speech quality [RBP+19, CNL+21]. Recently, an objective metric called the acoustic-echo cancellation mean-opinion score (AECMOS) was introduced. In double-talk specifically, the AECMOS has obtained impressive accuracy in estimating human ratings [PSS+22]. In contrast, RES systems conventionally use the SDR metric [VGF06] to assess speech quality in double-talk, e.g., in [CSVH18, DDBW19, PP20, CXCL20, Fan20b, Fan20a, WJ11, KJS21]. It will be empirically shown that the stereophonic SDR (SSDR) is by definition influenced by both distortion of stereo speech and presence of stereo residual-echo. Thus, for the task of RES in the stereophonic case, the SSDR is not an adequate indicator of neither the human evaluation for quality of speech nor of the AECMOS.

Voice-activity detection (VAD) in hands-free speech communication setups serves as a preliminary block that affects the performance of various speech-based systems that succeed it and depend upon its performance. In our context, detecting voice activity allows for an efficient usage of AEC upon these active segments, and propels their potential classification into single and double-talk. Classifying segments as silent by the VAD has the huge benefit of saving computational resources by the AEC and help the end-to-end hands-free speech communication systems to comply with practical computational budgets. Outside the context of this thesis, our VAD can help in speech separation, speech recognition and translation, speaker identification, and

general speaker diarization.

VAD refers to a family of methods that perform segmentation of an audio signal into parts that contain speech and silent parts. In this thesis, audio signals are captured by a single microphone and contain clean sequences of speech and silence. These signals are mixed with stationary and non-stationary noises (transients), e.g., door knocks and keyboard tapping [DC14, DTC16]. Our objective is to correctly assign each captured audio frame into the category of speech presence or absence. In acoustic environments that contain neither stationary or non-stationary noise, speech is detected by using methods that rely on frequency and energy values in short time frames [KN91, JMR94, VGX97]. These methods show significant deterioration in performance when noise is present, even with mild levels of signal-to-noise-ratios (SNRs). To cope with this problem, several approaches assume statistical models of the noisy signal in order to estimate its parameters [CK11, CKM06, SKS99, RSB$^+$04, CB01, Coh03]. Nonetheless, these methods are incapable of properly modeling transient interferences, which constitute an essential part of this study. Ideas that involve dimensionality reduction through kernel-based methods are introduced in [DTC15], where both supervised and unsupervised approaches have been exploited. However, its main limitation is a significant low-dimensional overlap between speech and non-speech representations. Machine learning techniques have been employed for voice activity detection in recent studies [SCK10, WZ11]. In contrast to classic methods, these approaches learn to implicitly model data without assuming an explicit model of a noisy signal. In particular, deep learning based methods have gained popularity in recent years due to a substantial increase in both computational power and data resources. Mendelev et al. [MPP15] constructed a deep neural network for voice activity detection, and suggested to employ the dropout technique [SHK$^+$14b] for enhanced robustness. The main drawback of this method is that temporal information between adjacent audio frames is ignored, due to independent classification of each time frame. Studies presented in [LHB15, GMH13, HM13, HL13] used a recurrent neural-network (RNN) to integrate temporal context with the use of past frames. However, the rapid time variations and prominent energy values of non-stationary noises in comparison to speech are still the main cause of degraded performance in these methods. A recent study conducted by

Ariav et al. [ADC18a] proposed to use an auto-encoder to implicitly learn an audio signal embedded representation. To enhance temporal relations between frames, this auto-encoder feeds an RNN. Despite its leading performance, the reported results are still unsatisfactory. Our study found that the main limitation of this algorithm is the dense low-dimensional representation forced by the auto-encoder and into the RNN. This density occurs largely due to the joint training of speech and non-speech frames, which fails to enhance their unique features. Thus, their low-dimensional representations, which are the sole information that feeds the RNN, are embedded closely in terms of Euclidean distance. Eventually, this poses a difficulty in separation of speech from non-speech frames based merely on temporal information, which is the core advantage of using RNN architecture.

1.2 Main Contributions

This thesis addresses the existing gaps described above and mitigates those gaps via eight main contributions. Five contributions relate to monophonic AEC, two contributions regard SAEC, and one additional contribution addresses VAD:

- We introduced a non-linear AEC system inspired by the physical behavior of modern hands-free devices, which features a novel neural network architecture that is specifically designed to model the non-linear distortions these devices induce between receiving and playing the far-end signal.
- We presented the DVSS framework for adaptation control that is data-driven and makes no assumptions on the acoustic setup and is entirely non-parametric.
- We proposed an RES system that embeds a novel design parameter that allows a dynamic tradeoff between the desired-speech distortion and residual-echo suppression levels in the system output in double-talk scenarios.
- We developed two objective metrics that separately evaluate the desired-speech maintained level and the residual-echo suppression level of RES systems during double-talk with high correlation to human subjective evaluation, and offered a framework to balance between them using our design parameter.

- We introduced the first user-centric RES framework in double-talk, which provides an RES output that maximizes subjective human ratings while confining with a user-specific operating point that consists of their requested desired-speech maintained level and residual-echo suppression level.

- We presented a general and data-driven adaptation-control framework for SAEC, with a compact and efficient adaptation rule that is expressed with the widely-linear model in the complex time domain.

- We proposed a pair of objective metrics that distinctly assess the stereophonic desired-speech maintained level and stereophonic residual-echo suppression level in the output of the RES during double-talk, and offered a framework to balance between them using our design parameter.

- We developed a VAD system that firstly exploits unique spatial patterns of speech and non-speech audio frames by independently learning their underlying geometric structures.

1.3 Overview of the Thesis

In Chapter 3, we make two contributions that are inspired by the physical behavior of modern hands-free devices. We first introduce a novel neural network architecture that is specifically designed to model the distortions these devices induce between receiving and playing the far-end signal. Second, we construct this neural network with trainable memory length and non-linear activation functions that are not parameterized in advance, but are rather optimized during the training stage based on the training data. The neural network output is inserted into a standard adaptive linear filter that constantly tracks the acoustic path from the loudspeaker output to the microphone. The end-to-end system, from the input of the neural network to the output of the linear filter, forms the proposed non-linear AEC system. During training, the neural network and the linear filter are jointly optimized to learn the neural network parameters. In testing, the neural network is used for inference and is not updated, while the linear filter is adapted to the time-varying acoustic paths. This system requires 17 thousand

parameters that consume 500 Million float-point operations per second (FLOPS) and 40 Kilo-bytes (KB) of memory, which renders it applicable for embedding on hands-free communication devices. It also meets the timing requirements of the AEC challenge [CSP+21], and more generally the constraints of hands-free communication standards [ETS16] on a standard neural processor.

In Chapter 4, we present the deep variable step-size (DVSS) framework. First, we solve a constrained non-linear optimization problem that minimizes the normalized misalignment between the actual and estimated echo path. Second, we present a deep neural network that learns the relation between the far-end, microphone, and a priori error signals and the optimal step-size. Finally, the trained neural network produces the VSS estimate in real-time, which is fed to the NLMS filter for echo cancellation. This data-driven method makes no acoustic assumptions and is completely non-parametric. The end-to-end system, from the neural network input to the NLMS output, comprises the proposed DVSS-NLMS filter. Notably, the DVSS framework can be generalized and is not restricted to NLMS-type algorithms. For evaluation, we use 100 hours of recordings from the AEC challenge database [CSP+21] and compare the DVSS to five competing methods. Experiments show that the DVSS is advantageous in echo cancellation and speech distortion in double-talk, is more robust to high levels of speech and noise, and has a better generalization to various non-linearities. The DVSS also achieves the best re-convergence times and success rates following abrupt echo-path changes during single-talk and double-talk across different acoustic conditions.

In Chapter 5, we introduce an RES method with a dual-channel input and single-channel output UNet neural network that directly maps the outputs of a linear AEC to the desired near-end signal in the short-time Fourier transform (STFT) domain. By utilizing the depth-wise separable convolution in every convolution layer of the UNet [GLA+20], the system comprises 136 thousand parameters that consume 1.6 Giga FLOPS and 10 mega-bytes (MB) of memory, which makes it suitable for on-device integration, e.g., using existing neural processors. Also, the system meets the timing standards of the AEC challenge [SCS+21], and more generally the constraints of hands-free communication systems [ETS16]. Even though competing models [HK20]–[ZTW19], [ZW18], [CSVH18] have shown promising results, the performance in real

acoustic environments is still challenging. Furthermore, a tunable tradeoff between the level of residual-echo suppression and desired-signal distortion may benefit applications that vary in their specific tradeoff requirements. However, this feature is not enabled by design in existing approaches. We bridge these gaps as follows. First, we conduct experiments with over 160 hours of data that was acquired from the AEC challenge database [SCS+21] and from independent recordings in real conditions. Second, a design parameter that allows dynamic balance between echo reduction and signal distortion is embedded in the UNet objective function that is minimized during the training process. The performance of the proposed system is compared to two existing methods: Zhang and Wang [ZW18], where a bi-LSTM structure was utilized to model an ideal ratio mask for both AEC and RES, and Carbajal et al. [CSVH18], who introduced a multiple input fully-connected neural network RES system, fed with the linear AEC outputs and reference far-end signal to estimate a phase-sensitive mask. Experimental results show state-of-the-art performance in various real-life acoustic setups. Particularly, high generalization is demonstrated in a variety of environments, devices, speakers, and moving echo paths. High robustness is also achieved in extreme conditions of very low signal-to-echo-ratios (SERs), and the effect of the tunable design parameter is demonstrated.

In Chapter 6, we show that RES can also be addressed as a speech separation (SS) [WC18] or speech enhancement (SE) [XDDL14] problem, where the echo is considered an interfering speech signal. We first fine-tune three off-the-shelf deep-learning-based systems: Our recently introduced RES system [ICB21a], a convolutional time-domain audio separation network called Conv-TasNet [LM19], and a denoiser develop by Facebook™ for SE [DSA20]. We show that the best-performing system of the three varies depending on the speech, echo, and noise levels. Second, we propose a real-time data-driven integration of these systems using a deep neural network that continuously tracks the best system based on single-talk and double-talk performance measures. Experiments with 100 hours of real and synthetic data show that the integrated system achieves better performance than each system in terms of echo cancellation and speech distortion across various acoustic setups in both single-talk and double-talk.

In Chapter 7, we introduce two objective metrics that separately evaluate the

desired-speech maintained level (DSML) and the residual-echo suppression level (RESL) during double-talk. Considering the RES system as a time-varying gain, the DSML is obtained by applying that gain to the desired speech and substituting the outcome in the definition of the SDR. The RESL is obtained by subtracting the desired speech from the double-talk segment and calculating the ratio of the noisy residual-echo before and after the gain is applied to it. To evaluate these metrics, we employ a deep learning-based RES system that also embeds a design parameter [ICB21a]. Experiments are done with 280 hours of real and simulated recordings in various scenarios and in high and low levels of echo and noise. Results show that the DSML and RESL have high correlation with human perception according to the DNSMOS, and high generalization to various setups, which renders them more suitable for speech quality evaluation than the SDR. We further investigate the empirical relation between tuning the design parameter and the DSML-RESL tradeoff it creates. Based on this relation, we offer a practical scheme for tuning the design parameter during training to optimally cope with dynamic system requirements.

In Chapter 8, we introduced the user-centric RES (URES) framework in double-talk. The URES is initiated with a user-operating point (UOP) that consists of two performance metrics values; the RESL and DSML [ICB21c] that the user wishes to experience from the RES prediction. The URES system then undergoes three stages. Firstly, we utilize an existing deep RES model that we introduced in [ICB21a]. This model embeds a design parameter that controls the trade-off between the RESL and DSML of the RES prediction. We consider 101 pre-trained instances from this model, each with a different design parameter value. Feeding the same input to all instances results in different RESL and DSML values in the prediction of every instance, which covers a wide range of UOPs. Second, each prediction is fed to a separate pre-trained deep model, which maps this prediction to its RESL and DSML estimates. This is essential since these metrics depend on the desired-speech signal that is unavailable in double-talk in practice. Third, the estimates from all instances are compared with the UOP. The ones that match it, up to a given tolerance threshold that specifies the allowed deviation from the UOP, are narrowed down to the single prediction with the maximal AECMOS, which is transmitted to the far-end. The proposed URES system

has three unique advantages; the RESL and DSML of its output match or approach the UOP, real-time changes in the UOP are tracked, and the AECMOS of its output is maximized. Experiments employ 60 hours of noisy real and synthetic data that include realistic acoustic scenarios with extremely high levels of echo and frequent echo-path changes. Average results can achieve an AECMOS of 4.4 out of 5 with RESL and DSML deviations of 1.95 dB and 2.1 dB from the UOP, where dB stands for decibles. Any user adjustment can be tracked in 38.4 ms, where ms stands for milliseconds. E.g., tightening the tolerance threshold can transition the above output to a lower AECMOS of 3.6 while the RESL and DSML deviations improve to 1.25 dB and 1.45 dB from the UOP, on average.

In Chapter 9, inspired by [ICB22a], we focus on the problem of SAEC and introduce a data-driven framework for DVSS that avoids heuristics and does not require acoustic setup hypotheses. First, the update rule of the adaptation process, governed by the step size, integrates the widely-linear model in the complex time domain. The mismatch between the actual echo paths and their filtered estimate is quantified by the normalized misalignment, which is then minimized with respect to the step size. A neural network (NN) relates acoustic signals to the optimal step-size in training, and the predicted step-size feeds the sign-NLMS (SNLMS) filter in real time for tracking the echo paths. We compare our approach with the competition by considering a pair of near-end loudspeakers and microphones, although this framework generalizes to any number of channels. Experimenting with 100 hours from the AEC-challenge corpus [CSP+21] reveals the consistent advantage of the DVSS in single and double-talk periods across various acoustic setups. The DVSS-SNLMS system also re-converges more rapidly and accurately after abrupt echo-path changes and is more robust to single-to-double-talk transitions.

In Chapter 10, we continue to address SAEC and introduce a pair of objective metrics to distinctly assess the stereophonic DSML (SDSML) and the stereophonic RESL (SRESL) in double-talk. We first consider an RES system that acts as a time-dependent gain, with a pair of input and output channels. To calculate the SDSML, this gain is projected into the stereo desired-speech and the result is substituted inside the SSDR expression. The SRESL requires an estimate of the noisy stereo residual-

echo, achieved by subtracting the stereo desired-speech from the double-talk frame. The ratio between this estimate without and with the gain applied to it generates the SRESL. The SDSML and SRESL metrics are evaluated with an RES system, based on deep learning, which incorporates a tunable design parameter. This study employs 100 hours of recordings that comprise of real signals and of simulations in various acoustic setups, with a range of echo and noise levels. Results reveal the AECMOS is well correlated with the SDSML and SRESL with high generalization to various scenarios. An additional empirical study investigates how the design parameter affects the tradeoff between the SDSML and SRESL. We then show how varying the design parameter during training can benefit interchangeable user demands of the RES system, which often occur in real-life. This study extends our previous work, which address the monophonic AEC case [ICB21c].

In Chapter 11, we concern the problem of VAD and propose an algorithm that addresses the limitations found in the methods proposed in [DTC15] and [ADC18a]. We independently learn the low-dimensional spatial patterns of speech and non-speech audio frames through the diffusion maps (DM) method. DM is a method that performs non-linear dimensionality reduction by mapping high-dimensional data points to a manifold, embedded in a low-dimensional space [TCGC13]. The mapped coordinates that lay on this manifold are referred to as DM coordinates. Since this method preserves locality, frames with similar contents in the original high dimension are mapped closely in the low, embedded dimension, with respect to their Euclidean distance. We separately apply DM for speech and non-speech frames through a pair of independent deep encoder-decoder structures. Inspired by the Diffusion nets architecture [MSCC17], the end of each encoder is forced to coincide with the embedded DM coordinates of its high-dimensional input. This approach allows us to differ the intrinsic structure of speech from the ones of transients and background noises based on the Euclidean metric. We suggest two variations for the voice activity detection algorithm, one for real-time applications and one for batch processes. We test both approaches on five comparative experiments conducted in [DTC15, ADC18a, TIH$^+$10]. Results show enhanced voice activity detection performance, that surpasses the known state-of-the-art speech detection results. Furthermore, our proposed architecture is more robust and

has better generalization ability than competing methods, as demonstrated through experiments.

In Chapter 12, we extend our VAD-related effort and consider the aforementioned five deep-learning-based VAD systems [ADC18b, WSSA18, KH18, AC19, IBC19] and five room impulse-response (RIR) models [Bor84, Vor89, Rin93, Lam05, VPH06]. First, we show that training these detectors with solely anechoic corpus and testing them in real reverberant rooms and spaces leads to a significantly impeded detection capability. To include unique acoustic patterns of reverberant data during training, we generated an augmented training set of nearly five million utterances. This extended corpus comprises of anechoic and reverberant signals, where the latter is generated by convolving the anechoic signals with a variety of RIRs, generated using a fixed RIR model. Then, all five VAD systems are independently trained with this augmented training set. This experiment is repeated for each of the five RIR models. All trained detection systems are tested in three real reverberant spaces of a classroom, a large concert hall, and an octagon shaped library. Experimental results demonstrate that the performance of all detectors is enhanced in each of the tested reverberant environments, regardless of the RIR model employed during training. Evaluation measures such as accuracy, precision and recall increase by 20% on average, compared to non-reverberant training. An interesting outcome shows that the leading accuracy of each detector was consistently achieved by the Valeaua RIR model [VPH06]. In a similar manner, the detector introduced by us [IBC19] prevailed competing VAD systems across all experiments.

1.4 Organization

This thesis is organized as follows. Chapter 2 aims to baseline required scientific background for the following chapters. The main contribution of this research thesis is laid out in detail in Chapters 3 to 12; Chapters 3 to 8 focus on monophonic AEC, Chapters 9 and 10 regard SAEC, and Chapters 11 and 12 discuss VAD. Finally, Chapter 13 concludes this thesis and offers future research directions.

Chapter 2

Scientific Background

2.1 Monophonic Acoustic-Echo Cancellation

Let $s(n)$ be the near-end speech signal and let $x(n)$ be the far-end speech signal. The microphone signal $m(n)$ is given by:

$$m(n) = s(n) + y(n) + w(n), \tag{2.1}$$

where $w(n)$ represents additive environmental and system noises and $y(n)$ is a nonlinear reverberant echo that is generated from $x(n)$.

The far-end signal, $x(n)$, is first nonlinearly distorted by electrical components such as a power amplifier and a loudspeaker that produce $x^{\mathrm{NL}}(n)$. Then, $x^{\mathrm{NL}}(n)$ is played by the loudspeaker and propagates via the linear acoustic path which is the room impulse response $h(n)$. Ultimately, $y(n)$ can be modeled as:

$$y(n) = \left(x^{\mathrm{NL}} * h\right)(n), \tag{2.2}$$

where $*$ is the convolution operator. The purpose of the AEC system is to suppress the echo $y(n)$ without distorting the desired signal $s(n)$, and transmit the system output to the far-end.

Traditionally, a linear AEC system is applied to reduce the linear component of $y(n)$ from the microphone signal. The AEC receives $m(n)$ as input and $x(n)$ as reference,

and aims to estimate the linear echo components, given by $\hat{y}(n)$:

$$\hat{y}(n) = \left(x * \hat{h}\right)(n), \tag{2.3}$$

where $\hat{h}(n)$ tracks the estimation of the near-end echo path $h(n)$. We assume that $h(n)$ and $\hat{h}(n)$ have the same length, and we represent the misalignment between them with $\tilde{h}(n)$:

$$\tilde{h}(n) = h(n) - \hat{h}(n). \tag{2.4}$$

The adaptation error of the linear AEC system is given by $e(n)$:

$$e(n) = m(n) - \hat{y}(n), \tag{2.5}$$

which can be reformulated using eqs. (2.1)–(2.4) as:

$$\begin{aligned} e(n) &= s(n) + (y(n) - \hat{y}(n)) + w(n) && (2.6)\\ &= s(n) + \left(x^{\mathrm{NL}} * h\right)(n) - \left(x * \hat{h}\right)(n) + w(n) && (2.7)\\ &= s(n) + \left(\left(x^{\mathrm{NL}} - x\right) * h\right)(n) - \left(x * \tilde{h}\right)(n) + w(n). && (2.8) \end{aligned}$$

Several observations can be made on the challenges of nullifying the echo components in the adaptation error expression.

First, the nonlinearity caused by nonideal hardware in hands-free communication systems impose $x^{\mathrm{NL}}(n) \neq x(n)$. Observing eq. (2.8), this projects that the $\left(\left(x^{\mathrm{NL}} - x\right) * h\right)(n)$ ingredient inside the adaptation error is not nullified. Chapter 3 in this thesis addresses this challenge.

Second, the echo component $\left(x * \tilde{h}\right)(n)$ inside the adaptation error, represents the mismatch between the real and estimated echo paths that is quantified by $\tilde{h}(n)$. In Chapter 4 we address this challenge and minimize the influence of $\tilde{h}(n)$.

The third challenge addresses the residual-echo components that remain in the adaptation error, even after the nonlinear and linear stages above. In Chapter 5, we employ deep learning to characterize and remove all these remaining echo components.

2.2 Stereophonic Acoustic-Echo Cancellation

In this thesis, the SAEC setup describes a scenario in which the near-end contains a pair of loudspeakers and a pair of microphones. The left and right near-end microphones $m_{\mathrm{L}}(n)$ and $m_{\mathrm{R}}(n)$ at time index n are, respectively,

$$m_{\mathrm{L}}(n) = s_{\mathrm{L}}(n) + y_{\mathrm{L}}(n) + w_{\mathrm{L}}(n), \tag{2.9}$$

$$m_{\mathrm{R}}(n) = s_{\mathrm{R}}(n) + y_{\mathrm{R}}(n) + w_{\mathrm{R}}(n), \tag{2.10}$$

where $s_{\mathrm{L}}(n)$ and $s_{\mathrm{R}}(n)$ are the near-end speech signals, $w_{\mathrm{L}}(n)$ and $w_{\mathrm{R}}(n)$ represent environmental and system noises, and $y_{\mathrm{L}}(n)$ and $y_{\mathrm{R}}(n)$ are the nonlinear reverberant echo signals, as correspondingly captured by the left and right microphones:

$$y_{\mathrm{L}}(n) = \left(x_{\mathrm{L}}^{\mathrm{NL}} * h_{\mathrm{LL}}\right)(n) + \left(x_{\mathrm{R}}^{\mathrm{NL}} * h_{\mathrm{RL}}\right)(n), \tag{2.11}$$

$$y_{\mathrm{R}}(n) = \left(x_{\mathrm{L}}^{\mathrm{NL}} * h_{\mathrm{LR}}\right)(n) + \left(x_{\mathrm{R}}^{\mathrm{NL}} * h_{\mathrm{RR}}\right)(n). \tag{2.12}$$

Here, $x_{\mathrm{L}}^{\mathrm{NL}}(n)$ and $x_{\mathrm{R}}^{\mathrm{NL}}(n)$ respectively denote the left and right far-end signals, i.e., $\mathbf{x}_{\mathrm{L}}(n)$ and $\mathbf{x}_{\mathrm{R}}(n)$, subsequent to nonlinear distortions by nonideal hardware [ICB21b]. All of $h_{\mathrm{LL}}(n)$, $h_{\mathrm{RL}}(n)$, $h_{\mathrm{LR}}(n)$, and $h_{\mathrm{RR}}(n)$ represents a linear echo-path from one of the loudspeakers to one of the microphones.

Traditionally, the echo components in each microphone are estimated using adaptive linear filtering:

$$\hat{y}_{\mathrm{L}}(n) = \left(x_{\mathrm{L}} * \hat{h}_{\mathrm{LL}}\right)(n) + \left(x_{\mathrm{R}} * \hat{h}_{\mathrm{RL}}\right)(n), \tag{2.13}$$

$$\hat{y}_{\mathrm{R}}(n) = \left(x_{\mathrm{L}} * \hat{h}_{\mathrm{LR}}\right)(n) + \left(x_{\mathrm{R}} * \hat{h}_{\mathrm{RR}}\right)(n), \tag{2.14}$$

where $\hat{h}_{\mathrm{LL}}(n)$, $\hat{h}_{\mathrm{RL}}(n)$, $\hat{h}_{\mathrm{LR}}(n)$, and $\hat{h}_{\mathrm{RR}}(n)$ respectively track the estimation of the near-end echo paths $h_{\mathrm{LL}}(n)$, $h_{\mathrm{RL}}(n)$, $h_{\mathrm{LR}}(n)$, and $h_{\mathrm{RR}}(n)$.

We consider a pair of left and right loudspeakers in the far-end. Each of these channels respectively receives the left and right adaptation errors, i.e., $e_{\mathrm{L}}(n)$ and $e_{\mathrm{R}}(n)$.

Using eqs. (2.9)–(2.14):

$$
\begin{aligned}
e_{\mathrm{L}}(n) &= m_{\mathrm{L}}(n) - \hat{y}_{\mathrm{L}}(n) \qquad (2.15)\\
&= s_{\mathrm{L}}(n) + \left(x_{\mathrm{L}}^{\mathrm{NL}} * h_{\mathrm{LL}} - x_{\mathrm{L}} * \hat{h}_{\mathrm{LL}}\right)(n) + \left(x_{\mathrm{R}}^{\mathrm{NL}} * h_{\mathrm{RL}} - x_{\mathrm{R}} * \hat{h}_{\mathrm{RL}}\right)(n) + w_{\mathrm{L}}(n)\,.\\
e_{\mathrm{R}}(n) &= m_{\mathrm{R}}(n) - \hat{y}_{\mathrm{R}}(n) \qquad (2.16)\\
&= s_{\mathrm{R}}(n) + \left(x_{\mathrm{L}}^{\mathrm{NL}} * h_{\mathrm{LR}} - x_{\mathrm{L}} * \hat{h}_{\mathrm{LR}}\right)(n) + \left(x_{\mathrm{R}}^{\mathrm{NL}} * h_{\mathrm{RR}} - x_{\mathrm{R}} * \hat{h}_{\mathrm{RR}}\right)(n) + w_{\mathrm{R}}(n)\,.
\end{aligned}
$$

For each of the left and right channels, the purpose of the SAEC system is to suppress the echo components and leave the desired-speech undistorted, and transmit each of the system outputs to the far-end. In Chapter 9, we address the challenge of SAEC.

Chapter 3

Nonlinear Acoustic-Echo Cancellation

3.1 Introduction

Hands-free communication often involves a conversation between two speakers located at near-end and far-end points. The near-end microphone captures the desired-speech signal and two interfering signals: echo produced by a loudspeaker playing the far-end signal, and background noises. The acoustic coupling between the loudspeaker output and the microphone may lead to degraded speech intelligibility in the far-end due to echo presence [SMH95]. This problem prompted numerous studies regarding AEC systems that aim to remove echo and preserve the near-end speech [BGM$^+$01]. In recent years, however, miniaturization of electronic components in hands-free devices, e.g., smart phones, smart speakers, and wearable devices, caused non-negligible nonlinear distortions in the echo path between the far-end signal and the loudspeaker output [BG95a]. Consequently, AEC systems that assume an echo path that is linear often fail in practice [MEB10].

To mitigate this mismatch, various nonlinear AEC approaches were proposed to identify the nonlinear echo path. The Volterra series showed success in modeling systems with weak nonlinearities and memory using nonlinear basis functions, while often requiring high computational complexity [GFLBJ03]. A simplified version is given by

the block-oriented Hammerstein and Wiener models, which describe nonlinear systems without memory and linear systems with memory [SCPU11]. Also, adaptive functional link filters [CSAR+13], Bayesian state-space modeling [ME12], and kernel-based methods [VVARC16] are commonly used for nonlinear AEC. Avargel and Cohen considered this problem from a time-frequency point-of-view and applied multiplicative function approximation [AC08], sub-band adaptive filtering [AC09a], and an efficient Volttera series modeling using cross-band terms [AC09b], [AC10]. Neural networks provide an alternative framework for a more accurate nonlinear modeling compared to classic approaches [BG95b], [RT98], [Jan04], [ZZ17]. For instance, Malek and Koldovsky [MK16a] estimated the nonlinear echo path with a fully-coneural networkected neural network that assumes the Hammerstein model, followed by an adaptive linear filter to track the acoustic path. Recently, Halimeh et al. [HHK19] constructed an fully-coneural networkected neural network that assumes the Wiener-Hammerstein model and captures both the nonlinear and linear echo paths.

Despite showing promising results, the performance of these methods is still challenging in real-life scenarios, which may be associated with two of their attributes. First, these models are not accurately designed according to the physical behavior of distortions that modern hands-free devices apply to the far-end signal. Second, they are mostly parametric, i.e., they require that memory lengths and nonlinear basis functions are predetermined. E.g., in [GFLBJ03], [SCPU11], the presented models assume a given number of memory taps, and in [MK16a], [HHK19], fixed nonlinear activation functions are employed inside the neural network. These drawbacks may produce sub-optimal solutions in real setups.

To address these two gaps, we make two contributions that are inspired by the physical behavior of modern hands-free devices. We first introduce a novel neural network architecture that is specifically designed to model the distortions these devices induce between receiving and playing the far-end signal. Second, we construct this neural network with trainable memory length and nonlinear activation functions that are not parameterized in advance, but are rather optimized during the training stage based on the training data. The neural network output is inserted into a standard adaptive linear filter that constantly tracks the acoustic path from the loudspeaker

output to the microphone. The end-to-end system, from the input of the neural network to the output of the linear filter, forms the proposed nonlinear AEC system. During training, the neural network and the linear filter are jointly optimized to learn the neural network parameters. In testing, the neural network is used for inference and is not updated, while the linear filter is adapted to the time-varying acoustic paths.

This system requires 17 thousand parameters that consume 500 Million FLOPS and 40 Kilo-bytes (KB) of memory, which renders it applicable for embedding on hands-free communication devices. It also meets the timing requirements of the AEC challenge [CSP+21], and more generally the constraints of hands-free communication standards [ETS16] on a standard neural processor.

Performance is evaluated against two recent neural network-based nonlinear AEC methods in [MK16a] and [HHK19], and to a linear AEC method. Experiments are conducted with 280 hours of both synthetic and real data, which include half-duplex and full-duplex periods affiliated with various acoustic environments, devices, speakers, and noise and echo levels. Results show leading performance of the proposed nonlinear AEC system in terms of echo cancellation and speech distortion levels, generalization and stability to various setups, robustness to high levels of noise and echo, and convergence and re-convergence rates.

The remainder of this chapter is organized as follows. In Section 3.2, we formulate the problem. In Section 3.3, we describe the proposed solution. In Section 3.4, we lay out the experimental setup. In Section 3.5, we present the experimental results. Finally, in Section 3.6, we draw conclusions.

3.2 Problem Formulation

Figure 3.1 depicts the scenario and proposed system for nonlinear AEC. Let $s\left(n\right)$ be the near-end speech signal and let $x\left(n\right)$ be the far-end speech signal. The microphone signal $m\left(n\right)$ is given by

$$m\left(n\right) = s\left(n\right) + y\left(n\right) + w\left(n\right), \tag{3.1}$$

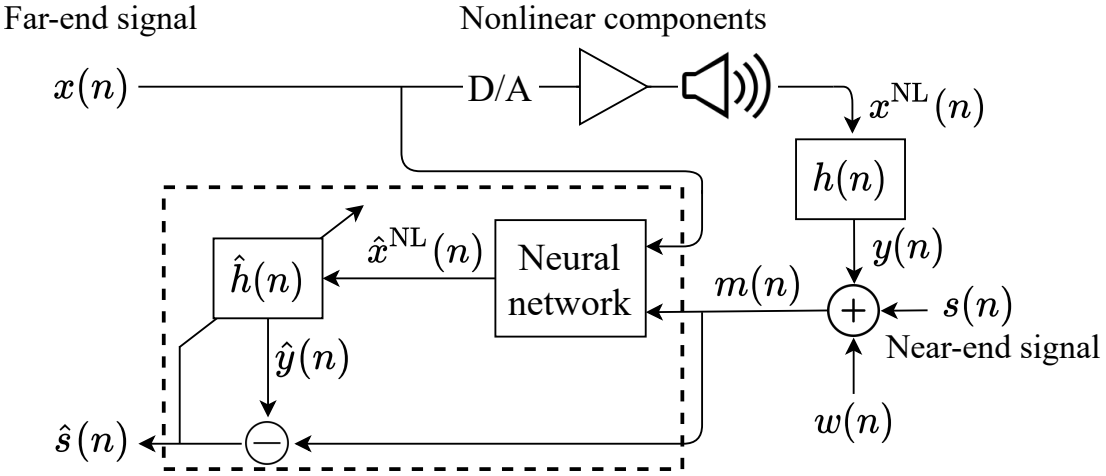

Figure 3.1: Nonlinear AEC scenario and proposed system (bordered). The nonlinear components are modeled with a neural network and the acoustic path with a standard adaptive linear filter.

where $w(n)$ represents additive environmental and system noises and $y(n)$ is a nonlinear reverberant echo that is generated from $x(n)$. The far-end signal, $x(n)$, is first distorted by electrical components that produce $x^{\mathrm{NL}}(n)$, and then $x^{\mathrm{NL}}(n)$ propagates via a linear acoustic path $h(n)$, namely $y(n) = \left(x^{\mathrm{NL}} * h\right)(n)$. The proposed nonlinear AEC system attempts to estimate $y(n)$ by using a neural network to find $\hat{x}^{\mathrm{NL}}(n)$, which is an estimate for $x^{\mathrm{NL}}(n)$, and filtering the result with an adaptive linear filter that tracks the acoustic path, denoted by $\hat{h}(n)$:

$$\hat{y}(n) = \left(\hat{x}^{\mathrm{NL}} * \hat{h}\right)(n). \tag{3.2}$$

The signal transmitted to the far-end is given by

$$\hat{s}(n) = m(n) - \hat{y}(n) = s(n) + (y(n) - \hat{y}(n)) + w(n). \tag{3.3}$$

Our goal is to cancel the echo $y(n)$ by eliminating the term $y(n) - \hat{y}(n)$, without distorting the speech signal $s(n)$.

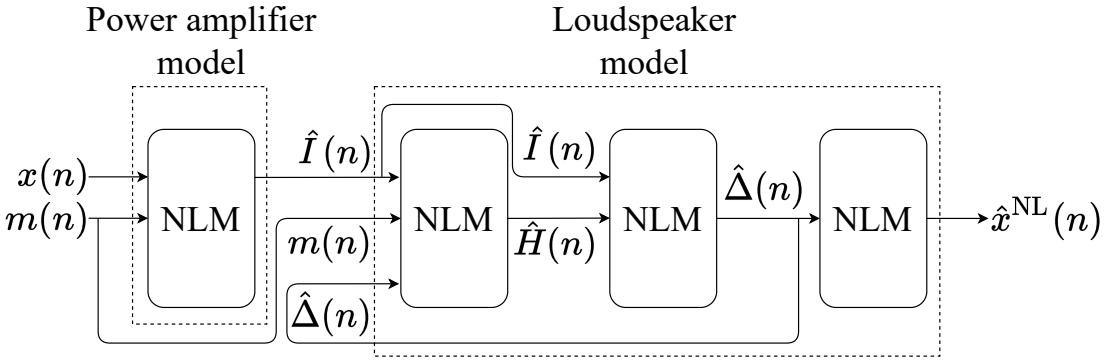

Figure 3.2: Proposed neural network architecture.

3.3 Nonlinear Acoustic-Echo Cancellation

The proposed nonlinear AEC system is comprised of two parts. First, a neural network models the physical behavior of distortions applied between the far-end signal and the loudspeaker output, caused by non-ideal electrical components in practical hands-free devices. Second, a standard adaptive linear filter tracks the acoustic-echo path from the loudspeaker output to the microphone.

In order to understand our system, it is helpful to understand how the above-mentioned electrical components behave. Modern hands-free devices often apply distortions between receiving the far-end signal and playing it in the near-end. These distortions are created by three different electrical components; a D/A, a power amplifier, and a loudspeaker [Dob11], [Kli05], [RLRL10], [SRGZ$^+$04]. This study uses a 16-bit data precision, so the signal-to-quantization-noise ratio is sufficiently high and the D/A distortions are numerically negligible [Dob11]. Thus, the D/A is not modeled. Ideally, the power amplifier should increase the energy of its input signal without distortions by using the power supply from the device battery. However, low-powered hands-free devices drive the amplifier to operate close to saturation, which yields distortions. The specific nonlinear behavior of each amplifier depends on its saturation curve, ranging from a soft-clipped sigmoid, to a hard-clipped rectified function, and in extreme cases, it may exhibit a square waveform behavior [Dob11].

The loudspeaker component is responsible for the majority of distortions. In this study, the widely-used electro-dynamic loudspeaker model is considered, which exhibits four major types of nonlinearities; electrical, magnetic, mechanical, and acousti-

cal [SRGZ+04]. The electrical signal, $I(n)$, is received from the amplifier output and creates a magnetic field signal of strength $H(n)$ around the voice coil, which renders it an electromagnet. The relation between $I(n)$ and $H(n)$ is nonlinear and depends on the coil displacement signal, $\Delta(n)$. Both $I(n)$ and $H(n)$ lead to polarity changes in the electromagnet that moves the coil back and forth with force that also has NL relations with $\Delta(n)$. This movement creates air pressure that is translated into acoustic sound waves that depend on $\Delta(n)$ and its temporal derivatives. This relation is nonlinear as well due to wave propagation and mechanical nonlinearities, caused by stiffness of the loudspeaker spider. Both the power amplifier and loudspeaker components may depend on previous observations.

The above nonlinear behavior is modeled using a neural network that is comprised of two cascaded parts: a power amplifier model, and a loudspeaker model, depicted in Figure 3.2. First, the amplifier is modeled with 3 identical GRUs that contain 16 cells each [CGCB14] and dropout [SHK+14a] in the recurrent layers, a fully-connected neural network with a one-neuron output, and a PLU activation function layer with trainable parameters [Nic18]. This entire NLM is fed with the far-end and microphone waveform signals, since the latter contains information about the distortions of the former. Second, the loudspeaker is modeled by a sequence of 3 consecutive NLMs. It receives the output of the amplifier, i.e., the estimated excitation current $\hat{I}(n)$ that drives the loudspeaker. Similarly to the amplifier model, $\hat{I}(n)$ is concatenated to the microphone signal, and the first NLM learns the electrical-to-magnetic nonlinear model from $\hat{I}(n)$ to $\hat{H}(n)$. Then, the predicted $\hat{H}(n)$ is concatenated to $\hat{I}(n)$ and inserted to the second NLM, which learns the magnetic-to-mechanical nonlinear model and predicts $\hat{\Delta}(n)$. Then, $\hat{\Delta}(n)$ is inserted to the third NLM, which learns the mechanical-to-acoustic nonlinear model and estimates the distorted far-end signal at the output of the loudspeaker, i.e., $\hat{x}^{\mathrm{NL}}(n)$. Since $\hat{\Delta}(n)$ also affects $\hat{H}(n)$, the first NLM is fed with the output of the second NLM using a skip-connection. The NLM unit is adjusted to receive between 1 to 3-dimensional input signals across the neural network model. Following this neural network, a linear adaptive filter models the acoustic path between the loudspeaker output and the microphone. This filter contains 150 samples and was developed by Phoenix Audio Technologies$^{\mathrm{TM}}$ using a filter bank approach. The neural

network and the linear filter construct the proposed end-to-end nonlinear AEC system.

To the best of our knowledge, the proposed neural network architecture is used in this study for the first time. The neural network is based on the GRU, whose internal gate-based mechanism is optimized for nonlinear sequence-to-sequence mapping in the waveform domain. Also, the GRU keeps relevant past information without discarding it through time, while neglecting irrelevant data. Thus, the optimal memory length is implicitly learned by the neural network during training and should not be set in advance. The trainable PLU parameters are also adjusted during training to optimally describe various saturation curves of the power amplifier and other nonlinear behaviors exhibited by the loudspeaker. Thus, the nonlinear behavior of the neural network is not restricted to a predetermined set of nonlinear basis functions. In addition, the GRU consumes low computational resources and requires short inference time.

The nonlinear AEC system contains 17 thousand parameters that consume 500 Million FLOPS and 40 KB of memory. Thus, its integration on hands-free devices is enabled, e.g., using the NDP120 neural processor by Syntiant$^{\text{TM}}$ [Syn21]. Timing constraints of hands-free communication on that processor are also met [ETS16].

3.4 Experimental Setup

3.4.1 Database Acquisition

Two data corpora are employed in this study; the AEC challenge database [CSP$^+$21], and a database recorded in our lab, both sampled at 16 kHz. These corpora include single-talk and double-talk periods both with and without echo-path change. In the case of no echo-path change, there is no movement in the room during the recording. In the other case, either the near-end speaker or the device are constantly moving during the recording. In [CSP$^+$21], two open sources of synthetic and real recordings are introduced. The synthetic data includes 100 hours, and the real data contains 140 hours of audio clips, generated from $5,000$ hands-free devices that are used in various acoustic environments. In both real and synthetic cases, SER and SNR levels were distributed on $[-10, 10]$ dB and $[0, 40]$ dB, respectively. Additional real recordings were conducted in our lab to test the generalization of the system to unseen setups and its

robustness to extremely low levels of SERs. This database is fully described in [ICB21a]. For completion, it contains 40 hours of recordings from the TIMIT [GLF+93b] and LibriSpeech [PCPK15] corpora with SNR levels of 32 ± 5 dB and SER levels distributed on $[-20, -10]$ dB.

Formally, the SER and SNR captured by the microphone are calculated with 50% overlapping time frames of 20 ms and are defined as $\text{SER} = 10 \log_{10} \left(\|s(n)\|_2^2 / \|y(n)\|_2^2 \right)$ and $\text{SNR}=10 \log_{10} \left(\|s(n)\|_2^2 / \|w(n)\|_2^2 \right)$, in dB.

3.4.2 Data Processing, Training, and Testing

The real and synthetic data from [CSP+21] is randomly split to create 185 hours of training set and 45 hours of validation set. The test set contains only real data that is comprised of the remaining 10 hours from [CSP+21] and all 40 hours from [ICB21a]. Each set is divided into 10 s segments that contain recordings in different setups. This leads to frequent re-convergence during transitions between segments, both without and with echo-path change. These sets are balanced to prevent bias in results, as detailed in [ICB21a].

During training, the neural network and the succeeding linear filter are jointly optimized to learn the neural network parameters. Optimization is done by minimizing the ℓ_2 distance between the output of the nonlinear AEC, $\hat{s}(n)$, and the desired-near-end speech $s(n)$.

To train the neural network, back-propagation through time is used with a learning rate of 0.0005, mini-batch size of 32 ms, and 20 epochs, using Adam optimizer [KB15]. Also, automatic differentiation [PGC+17] is applied, since the loudspeaker modeling involves temporal derivatives of its input signals. Training duration was typically 15 minutes per 10 hours of data on an Intel Core i7-8700K CPU @ 3.7 GHz with two GPUs of type Nvidia GeForce RTX 2080 Ti.

During testing, the neural network is used for inference only and is not updated. The linear filter receives the outputs of the neural network and is continuously adapted to account for time variations of the acoustic path. An artificial gain may be introduced by the neural network, which is compensated as shown in [VGF06].

Measure	Definition
ERLE	$10\log_{10}\left(\|m(n)\|_2^2/\|\hat{s}(n)\|_2^2\right)\Big\|_{\text{Far-end single-talk}}$
SDR	$10\log_{10}\left(\|s(n)\|_2^2/\|\hat{s}(n)-s(n)\|_2^2\right)\Big\|_{\text{Double-talk}}$

Table 3.1: Performance measures for nonlinear AEC estimation.

3.4.3 Performance Measures

To evaluate performance, the ERLE [ITU12] is used. It measures echo reduction between the degraded and enhanced signals when only a far-end signal and noise are present. For double-talk periods, we use the SDR [VGF06] that takes echo suppression and speech distortion into account, and the PESQ measure [ITU01], [ITU17]. The PESQ is calculated over an entire 10 s segment. The ERLE and SDR are calculated with 50% overlapping frames of 20 ms, and are defined in Table 3.4.2.

3.5 Experimental Results

The proposed nonlinear AEC system is compared against two competing neural network-based methods in [MK16a] and [HHK19], notated "Malek" and "Halimeh", respectively. For linear echo path approximation, the proposed system and "Malek" are implemented with an identical adaptive linear filter presented in Section 3.3, while "Halimeh" approximates the linear echo via a neural network. As benchmark, the linear filter is also applied alone, and this method is denoted by "Linear". Measures are reported by their mean and std values, with respect to the test set specified in each experiment. Unless stated otherwise, the format of the results is presented as mean±std. In this study, convergence was reached if the normalized misalignment between consecutive linear echo approximations was lower than -30 dB [PCBG15].

Results for segments with no echo-path change are given in Table 3.2 and for segments with echo-path change are given in Table 3.3, both after convergence. Compared to competition, the proposed method achieves enhanced echo cancellation in single-talk periods according to the ERLE measure. In double-talk periods, less speech distortion and better speech quality are obtained, as suggested by the SDR and PESQ scores,

respectively. Also, a lower std measure is achieved, which projects better stability of our method across various setups. Scenarios of echo-path change lead to overall decline in performance relative to no echo-path change, as expected. However, our method still prevails competition across all measures in terms of both higher mean and lower std. Based on the above, our method allows enhanced modeling of the nonlinear echo path, which improves both the estimation of acoustic paths with no echo-path change, and the tracking of acoustic paths with echo-path change.

In addition, we investigate the performance before convergence and during re-convergence for segments with no echo-path change. Due to the test set segmentation described in Section 3.4.2, re-convergence frequently occurs during transitions between segments. As shown in Table 3.4, performance is collectively impeded relative to the converged case in Table 3.2. However, our method still prevails across all measures in terms of both mean and std values. This indicates the high sensitivity of competing methods to converged echo approximation, while our model captures the behavior of the echo even from degraded measurements. We also examine the convergence time of each method. According to Table 3.5, our method achieves the shortest convergence time compared to competition. Again, it can be suggested that enhanced modeling of the nonlinear echo path is obtained by the proposed neural network, which allows the succeeding linear filter to be adjusted more accurately and rapidly.

Next, performance with no echo-path change is examined in various SNR and SER levels, after convergence. As shown in Figures 3.3–3.6, all methods suffer from decline in performance when acoustic conditions deteriorate. However, our method outperforms competition in both PESQ and SDR measures across all SNR and SER levels, which projects high generalization ability to various levels of noise and echo. The relatively stable behavior of the proposed method, especially in low levels of SNRs and SERs, indicates high robustness to high levels of noise and echo that often occur in practice. Interestingly, in severely degraded conditions of 0 dB SNR and of -20 dB SER, the proposed method achieves roughly 1 dB higher SDR and 0.5 higher PESQ score on average than the competition in second place.

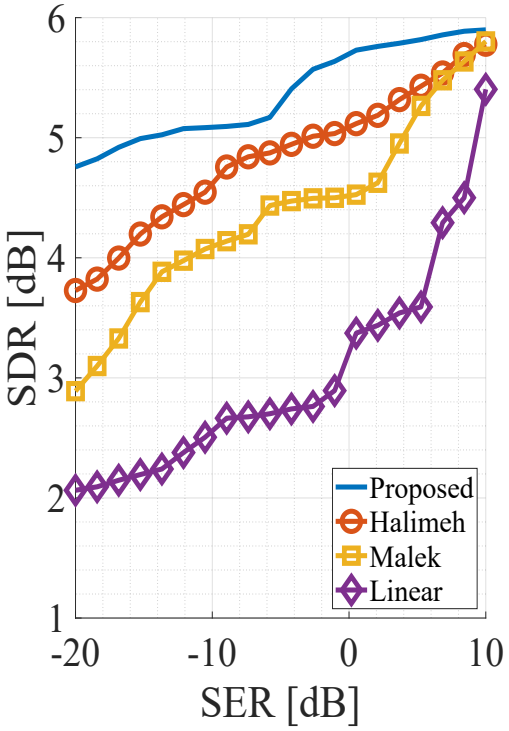

Figure 3.3: Average SDR in various SER levels for no echo-path change scenarios.

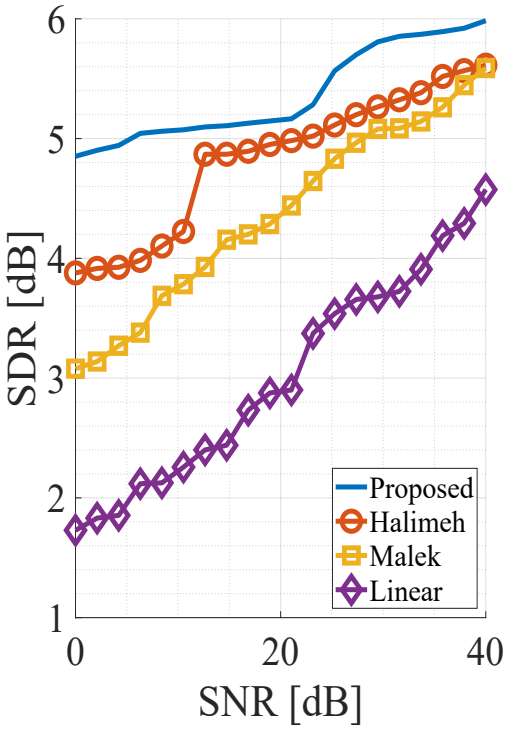

Figure 3.4: Average SDR in various SNR levels for no echo-path change scenarios.

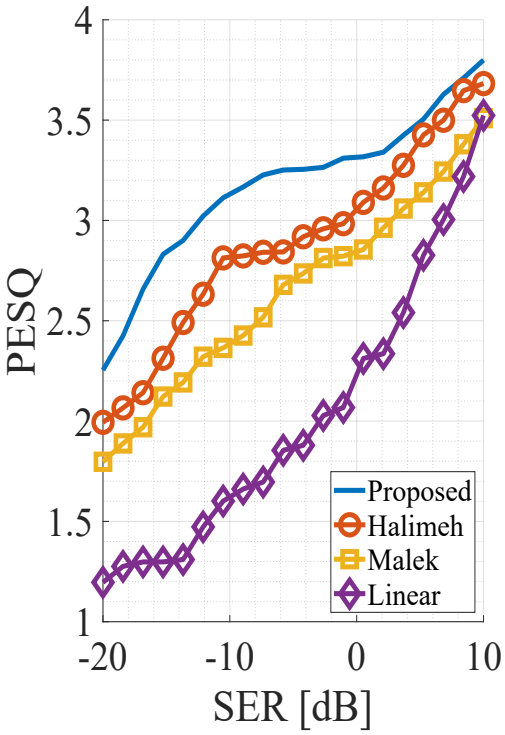

Figure 3.5: Average PESQ in various SER levels for no echo-path change scenarios.

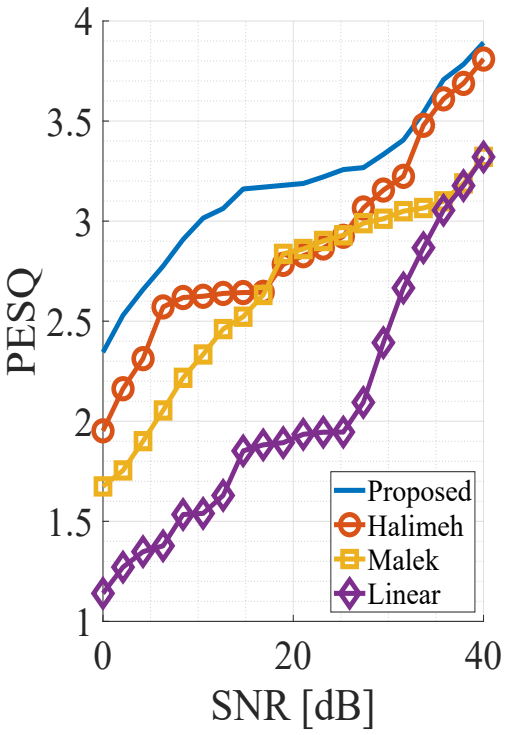

Figure 3.6: Average PESQ in various SNR levels for no echo-path change scenarios.

	Proposed	Halimeh	Malek	Linear
ERLE	**26.4±5.1**	23.1±5.9	22.6±6.7	21.3±7.2
PESQ	**3.17±0.4**	2.88±0.5	2.64±0.5	2.02±0.7
SDR	**5.37±0.4**	4.83±0.6	4.37±0.8	3.01±0.9

Table 3.2: Performance with no echo-path change.

	Proposed	Halimeh	Malek	Linear
ERLE	**23.2±6.0**	19.2±7.7	18.0±8.3	16.9±8.9
PESQ	**2.92±0.5**	2.54±0.7	2.31±0.6	1.91±0.6
SDR	**5.08±0.6**	4.25±0.9	3.82±0.9	2.52±1.0

Table 3.3: Performance with echo-path change.

	Proposed	Halimeh	Malek	Linear
ERLE	**19.7±7.5**	14.9±8.1	13.8±8.8	11.0±9.6
PESQ	**2.56±0.6**	1.98±0.7	1.91±0.7	1.75±0.6
SDR	**4.71±0.9**	3.58±1.2	3.04±1.3	1.54±1.3

Table 3.4: Performance before convergence.

Proposed	Halimeh	Malek	Linear
4.6±0.7	6.6±1.1	7.3±1.4	7.9±1.8

Table 3.5: Convergence times [sec].

3.6 Conclusions

We have presented a nonlinear AEC system that comprises a novel neural network architecture and a succeeding standard adaptive linear filter. To describe the distortions modern hands-free devices induce between receiving and playing the far-end signal, we constructed the neural network of a power amplifier model followed by a loudspeaker model. The adaptive filter is fed by the neural network and tracks the acoustic path from the loudspeaker output to the microphone. The neural network parameters are

updated during training using joint optimization of the neural network and the filter. The nonlinear AEC implementation is adequate for integration on hands-free devices, and can meet timing requirements of hands-free communication standards on a standard neural processor. Experiments with 280 hours of real and synthetic recordings demonstrate the improved performance of our method compared to competition in terms of echo suppression and desired-signal distortion, generalization and stability in various setups, robustness to high levels of noise and echo, and convergence and re-convergence times.

Chapter 4

Deep Adaptation Control for Acoustic-Echo Cancellation

4.1 Introduction

Hands-free speech communication often involves a conversation between two speakers located at near-end and far-end points. During double-talk, the near-end microphone captures the desired-speech signal in addition to an echo produced by a loudspeaker that nonlinearly distorts and plays the far-end signal. The acoustic coupling between the loudspeaker and the microphone may lead to degraded speech intelligibility in the far-end due to echo presence [SMH95]. AEC aims to identify the echo path with an adaptive filter and create a replica of the echo that is subtracted from the microphone signal[BGM$^+$01].

The NLMS filter is a popular adaptive filter since it is numerically stable and computationally efficient [PCBG15]. The NLMS integrates the normalized step-size parameter that governs the often conflicting fast convergence requirements and low misadjustment. Therefore, it is highly desirable to control the step-size during adaptation in practical scenarios of time-varying echo paths and double-talk. This problem has motivated numerous VSS related studies. For example, Haubner et al. employed neural networks for near-end estimation[HHB$^+$20], noise estimation[HBEK21], and minimizing the error using adaptation control in the frequency domain[HBK21]. Meier and

Kellermann[MK16b] employed a deep neural network that maps statistical features of the far-end and a priori error signals to an analytically derived VSS. A batch of classic approaches includes the NPVSS that adjusts the step-size by reducing the squared error at each instant [BRVT06], the mean error SVSS that applies decompoisition of the error into sub-blocks [HA16], and HVSS that estimates the system noise power to control the step-size update [HL11].

However, existing approaches make restricting assumptions in real-life setups, e.g., assuming a linear relationship between the echo and the far-end signals[HHB+20]–[HL11], and adopting a time-invariant echo-path[BRVT06]. In practice, these assumptions result in filter misadjustment and slow convergence rates during echo-path changes [ICB21b]. Also, such methods require tuning parameters that are difficult to control in real-life scenarios. For example, the NPVSS [BRVT06] involves estimating the noise power, which is challenging during double-talk.

We address these gaps by presenting a DVSS framework. First, we solve a constrained nonlinear optimization problem that minimizes the normalized misalignment between the actual and estimated echo path. Second, we present a deep neural network that learns the relation between the far-end, microphone, and a priori error signals and the optimal step-size. Finally, the trained neural network produces the VSS estimate in real-time, which is fed to the NLMS filter for echo cancellation. This data-driven method makes no acoustic assumptions and is completely non-parametric. The end-to-end system, from the neural network input to the NLMS output, comprises the proposed DVSS-NLMS filter. Notably, the DVSS framework can be generalized and is not restricted to NLMS-type algorithms.

For evaluation, we use 100 hours of recordings from the AEC challenge database [CSP+21] and compare the DVSS to five competing methods. Experiments show that the DVSS is advantageous in echo cancellation and speech distortion in double-talk, is more robust to high levels of speech and noise, and has a better generalization to various nonlinearities. The DVSS also achieves the best re-convergence times and success rates following abrupt echo-path changes during single-talk and double-talk across different acoustic conditions.

The remainder of this chapter is organized as follows. In Section 4.2, we formulate

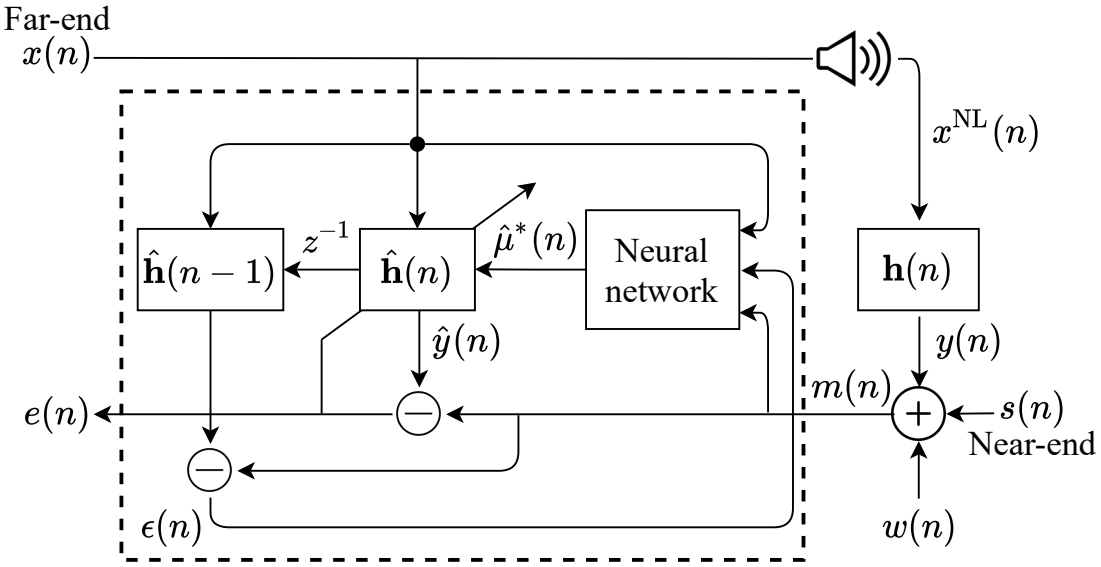

Figure 4.1: AEC scenario and proposed system (bordered). The neural network produces the DVSS estimate $\hat{\mu}^*(n)$, which is fed to an NLMS filter that generates the acoustic path estimation $\hat{\mathbf{h}}(n)$.

the problem. In Section 4.3, we describe the proposed solution. In Section 4.4, we lay out the experimental setup. In Section 4.5, we present the experimental results. Finally, in Section 4.6, we draw conclusions.

4.2 Problem Formulation

Figure 4.1 illustrates the DVSS-NLMS configuration. The microphone signal $m(n)$ at time index n is given by:

$$m(n) = s(n) + y(n) + w(n), \tag{4.1}$$

where $s(n)$ is the near-end speech signal, $w(n)$ represents environmental and system noises, and $y(n) = \mathbf{x}^{\mathrm{NL}^T}(n)\,\mathbf{h}(n)$ is a nonlinear and reverberant echo. For sake of readability, in this chapter we define notations by using explicit vector representations. Namely, $\mathbf{x}^{\mathrm{NL}}(n)$ denotes the L most recent samples of the far-end signal, $\mathbf{x}(n)$, after undergoing nonlinear distortions by nonideal components, and the echo path $\mathbf{h}(n)$ is

modeled as a finite impulse response filter with L coefficients:

$$\mathbf{x}^{\mathrm{NL}}(n) = \left[x^{\mathrm{NL}}(n), \ldots, x^{\mathrm{NL}}(n-L+1)\right]^T, \tag{4.2}$$

$$\mathbf{h}(n) = \left[h_0(n), h_1(n), \ldots, h_{L-1}(n)\right]^T. \tag{4.3}$$

An NLMS adaptive filter with L coefficients tracks the echo path estimate $\hat{\mathbf{h}}(n)$ and echo estimate $\hat{y}(n) = \mathbf{x}^T(n)\,\hat{\mathbf{h}}(n)$:

$$\mathbf{x}(n) = \left[x(n), x(n-1), \ldots, x(n-L+1)\right]^T, \tag{4.4}$$

$$\hat{\mathbf{h}}(n) = \left[\hat{h}_0(n), \hat{h}_1(n), \ldots, \hat{h}_{L-1}(n)\right]^T. \tag{4.5}$$

Then, an estimate of the near-end speech signal is given by

$$e(n) = m(n) - \hat{y}(n) = (y(n) - \hat{y}(n)) + s(n) + w(n). \tag{4.6}$$

Our goal is to estimate $\hat{\mathbf{h}}(n)$ and to cancel the echo by eliminating $y(n) - \hat{y}(n)$, without distorting the speech $s(n)$.

4.3 Deep Variable Step-Size Algorithm

4.3.1 General NLMS Filter Model in Double-talk

The a priori and a posteriori error signals of the NLMS adaptation process are, respectively, given by [PCBG15]:

$$\epsilon(n) = \mathbf{x}^{\mathrm{NL}^T}(n)\,\mathbf{h}(n) - \mathbf{x}^T(n)\,\hat{\mathbf{h}}(n-1) + s(n) + w(n), \tag{4.7}$$

$$e(n) = \mathbf{x}^{\mathrm{NL}^T}(n)\,\mathbf{h}(n) - \mathbf{x}^T(n)\,\hat{\mathbf{h}}(n) + s(n) + w(n). \tag{4.8}$$

Also, NLMS-type adaptive filters follow the update rule:

$$\hat{\mathbf{h}}(n) = \hat{\mathbf{h}}(n-1) + \mu(n)\,\mathbf{x}(n)\,\epsilon(n), \quad \hat{\mathbf{h}}(0) = \mathbf{0}^T, \tag{4.9}$$

where the step-size $\mu(n)$ is a positive scalar that controls the trade-off between convergence rate and adaptation misalignment and $\hat{\mathbf{h}}(0)$ has L zeros. From (4.7)–(4.9), we have

$$e(n) = \epsilon(n)\left(1 - \mu(n)\,\mathbf{x}^T(n)\,\mathbf{x}(n)\right). \tag{4.10}$$

To derive the general expression for $\mu(n)$, we impose echo cancellation from the a posteriori error, namely:

$$e(n) = s(n) + w(n). \tag{4.11}$$

Substituting (4.11) into (4.10) yields:

$$s(n) + w(n) = \epsilon(n)\left(1 - \mu(n)\,\mathbf{x}^T(n)\,\mathbf{x}(n)\right). \tag{4.12}$$

4.3.2 Data-driven Generation of the Optimal Step-Size

The normalized misalignment $\mathcal{D}(n)$ quantifies the mismatch between the actual and estimated echo paths in dB:

$$\begin{aligned}\mathcal{D}(n) &= 20\log_{10}\left(\frac{\|\mathbf{h}(n)-\hat{\mathbf{h}}(n)\|_2}{\|\mathbf{h}(n)\|_2}\right) \\ &= 20\log_{10}\left(\frac{\|\mathbf{h}(n)-\hat{\mathbf{h}}(n-1)-\mu(n)\,\mathbf{x}(n)\,e(n)\|_2}{\|\mathbf{h}(n)\|_2}\right).\end{aligned} \tag{4.13}$$

The optimal step-size $\mu^*(n)$ is the solution of the constrained nonlinear optimization problem that minimizes $\mathcal{D}(n)$:

$$\mu^*(n) = \underset{0<\mu(n)<1}{\operatorname{argmin}}\,\mathcal{D}(n), \tag{4.14}$$

where the constraint complies with the stability condition of NLMS-type algorithms [PCBG15]. This optimization process is carried out using the active-set optimization algorithm [HZ06]. According to (4.13), merely the far-end and a priori error signals are required for $\mu^*(n)$. This allows a non-parametric and data-driven approach to estimate $\mu^*(n)$.

4.3.3 Optimal Step-Size Learning Using Neural Networks

Deriving $\mu^*(n)$ in practice is time-consuming and requires knowledge of the echo path. Thus, a deep neural network is built to learn the relation between available data measurements and $\mu^*(n)$ during training, and to produce an estimate $\hat{\mu}^*(n)$ in real-time. According to (4.12), the step-size involves information of the far-end, a priori error, and near-end speech and noise signals. Even though the near-end signals are not available in practice, they comprise the available microphone signal. Thus, we propose a deep neural network that receives the far-end, a priori error, and microphone signals as inputs and maps them to the corresponding optimal step-size.

We employ a convolutional neural network [AMAZ17] with three input channels, one for each input signal, and a single-neuron output for the step-size. Each input channel is fed with its corresponding waveform signal's STFT [GL84] amplitude. The

first convolution layer employs a 3×3 kernel size, stride of 3, dilation of 5, and padding of 1, followed by 2-D batch normalization and a ReLU activation layer, and has 3 input and 16 output channels. A second convolution layer follows the same filtering specifications, but has 16 input and 16 output channels. A fully-connected neural network unit receives the 16 filters and propagates their flatten version through a 1920×512 layer, followed by 1-D batch normalization, a ReLU activation function, and a dropout layer with a probability of 0.5. Finally, this outcome is concatenated to a second fully-connected layer with dimensions 512×1 that ends with a sigmoid activation function. The objective function is the ℓ_2 distance between the neural network prediction and the optimal step-size $\mu^*(n)$.

In real-time, the neural network produces $\hat{\mu}^*(n)$, which is fed to the succeeding NLMS. This end-to-end system contains 1 Million parameters that consume 4 Million FLOPS and 4.6 MB of memory. Thus, its integration on hands-free devices is enabled with hands-free communication timing constraints met [ETS16], e.g., using the NDP120 neural processor by Syntiant$^{\text{TM}}$ [Syn21].

4.4 Experimental Setup

4.4.1 Database Acquisition

The AEC challenge database [CSP+21] is employed in this study. This corpus is sampled at 16 kHz and includes single-talk and double-talk periods both with and without echo-path change. No echo-path change means no movement in the room during the recording, and echo-path change means either the near-end speaker or the device are moving during the recording. The corpus includes 25 hours of synthetic data and 75 hours of real clean and noisy data. To account for realistic acoustic environments, every far-end signal randomly undergoes one of 4500 simulated nonlinear modifications, generated according to the physical behavior of power amplifiers and loudspeakers in modern hands-free devices [ICB21b]. Also, every nonlinearly-distorted signal is randomly propagated via one of 4500 real room impulse responses that are taken from the corpus in [SSM+19] with their first L coefficients. The echo-to-speech ratio (ESR) and echo-to-noise ratio (ENR) levels were distributed on $[-10, 10]$ dB

Measure	Definition
ERLE	$10\log_{10}\left(\|m(n)\|_2^2/\|e(n)\|_2^2\right)\Big\|_{\text{Far-end single-talk}}$
SDR	$10\log_{10}\left(\|s(n)\|_2^2/\|e(n)-s(n)\|_2^2\right)\Big\|_{\text{Double-talk}}$

Table 4.1: Performance measures for the DVSS approach estimation.

and $[0, 40]$ dB, respectively, and are defined as ESR=$10\log_{10}\left(\|y(n)\|_2^2/\|s(n)\|_2^2\right)$ and ENR=$10\log_{10}\left(\|y(n)\|_2^2/\|w(n)\|_2^2\right)$ in dB, both calculated with 50% overlapping time frames of 20 ms.

4.4.2 Data Processing, Training, and Testing

Initially, the 100 hours of real and synthetic data are randomly split to create 80 hours of training, 10 hours of validation, and 10 hours of test sets. All sets are balanced to prevent biased results, as detailed in [ICB21a]. The training and validation sets are used for step-size generation via (4.14) with $\mu(0) = 3\times10^{-5}$, $L = 2400$, and $\hat{\mathbf{h}}(0) = \mathbf{0}^T$ being a vector of L zeros. The step-size is generated every 8 ms to avoid unnecessary heavy computations. An abrupt change in echo path reoccurs every t seconds, where $t \sim U[4.5, 5.5]$, resembling real-life scenarios. The signals are transformed by the STFT using 16 ms frames and 8 ms shifts. Past information of 96 ms is concatenated before entering the neural network. Training the neural network is done using back-propagation through time with a learning rate of 10^{-4} that decays by 10^{-6} every 5 epochs, mini-batch size of 32 ms, and 40 epochs, using Adam optimizer [KB15]. In real-time, the neural network infers the test set and is not updated. The NLMS receives the optimal step-size estimate from the neural network and continuously tracks the echo path. The neural network may introduce an artificial gain, which is compensated as in [ICB21c]. Training duration was 30 minutes per 1 hours of data, and the batch inference time of the end-to-end system, i.e., the neural network and adaptive filter, is 24 ms on an Intel Core i7-8700K CPU @ 3.7 GHz with two GPUs of Nvidia GeForce RTX 2080 Ti.

4.4.3 Performance Measures

To evaluate the performance, the ERLE [ITU12] is used. It measures echo reduction between the degraded and enhanced signals when only a far-end signal and noise are present. In double-talk, we use the SDR [VGF06] that takes echo suppression and speech distortion into account, and the PESQ [ITU17]. All measures are calculated with 50% overlapping frames of 20 ms, and the ERLE and SDR are defined in Table 4.4.2. Convergence times and success rates are also given. Convergence occurs when $\mathcal{D}(n)$ falls under -10 dB and is successful if that holds for the remaining echo path. We also report the value of $\mathcal{D}(n)$ as given in (4.13).

4.5 Experimental Results

Using the entire test set, the DVSS method is compared against four competing VSS-based methods in [MK16b]–[HL11], respectively notated "NNVSS", "NPVSS", "SVSS", and "HVSS". All methods are implemented with the NLMS filter, which is also implemented with a constant step-size of $\mu = 3 \times 10^{-5}$ as the benchmark, notated "NLMS". In Tables 4.2 and 4.3, measures are reported by their mean and std values in the format mean±std. In Table 4.4, the average convergence times and success rates are reported.

Results with no echo-path change are given in Table 4.2 and with echo-path change are shown in Table 4.3, both after convergence. According to the ERLE measure, the proposed method achieves leading echo cancellation in single-talk. The DVSS yields less speech distortion and better speech quality during double-talk, respectively deduced by the SDR and PESQ scores. A lower std value is also achieved, which implies better stability of the DVSS across various setups. Although scenarios of echo-path change lead to expected performance decline relative to no echo-path change, our method outperforms competing methods across all measures in terms of mean and std. Furthermore, by Table 4.4, our method achieves the fastest average re-convergence time and highest convergence success rate compared to the competition. Thus, the data-driven DVSS that requires no acoustic assumptions and is entirely non-parametric, can track the VSS in practical acoustic conditions with double-talk with high generalization and robustness, and adjust the VSS most accurately and rapidly.

	SDR [dB]	PESQ	ERLE [dB]	Norm. Mis. [dB]
DVSS	**3.51±0.4**	**2.52±0.3**	**21.3±4.6**	**-22.8±4.2**
NNVSS	2.48±0.9	1.78±0.4	15.5±5.7	-16.8±4.9
NPVSS	2.81±0.8	2.06±0.5	16.8±6.7	-18.1±5.7
SVSS	2.21±0.9	2.03±0.6	15.0±5.5	-16.3±5.0
HVSS	2.86±0.6	2.12±0.4	18.1±6.5	-19.9±6.2
NLMS	2.09±1.1	1.62±0.3	14.2±5.8	-15.5±4.9

Table 4.2: Performance with no echo-path change.

	SDR [dB]	PESQ	ERLE [dB]	Norm. Mis. [dB]
DVSS	**3.16±0.6**	**2.31±0.5**	**16.9±5.7**	**-18.3±5.2**
NNVSS	2.11±1.1	1.75±0.5	11.9±5.5	-11.9±4.9
NPVSS	2.57±1.0	1.99±0.6	15.9±7.7	-17.4±7.1
SVSS	2.03±1.2	1.80±0.7	15.0±6.1	-13.4±5.9
HVSS	2.62±0.9	2.03±0.5	12.7±5.7	-15.1±4.2
NLMS	1.95±1.4	1.56±0.3	10.2±4.1	-11.0±3.0

Table 4.3: Performance with echo-path change.

DVSS	NNVSS	NPVSS	SVSS	HVSS	NLMS
3.4s, 95%	5.9s, 77%	6.6s, 75%	5.6s, 83%	7.0s, 71%	7.9s, 58%

Table 4.4: Convergence times [sec] and success rates [%].

Convergence comparison is illustrated in Fig. 4.2, where the ESR and ENR continuously vary, and after 5 s, an abrupt echo-path change occurs. The DVSS-NLMS filter continues to converge during double-talk and is only disturbed by the abrupt echo-path change. Also, the DVSS rapid convergence and re-convergence are demonstrated. All VSS-based competing methods experience divergence due to double-talk, which degrades their adaptation process. This supports previous conclusions regarding the DVSS superiority in real conditions such as double-talk and echo-path changes.

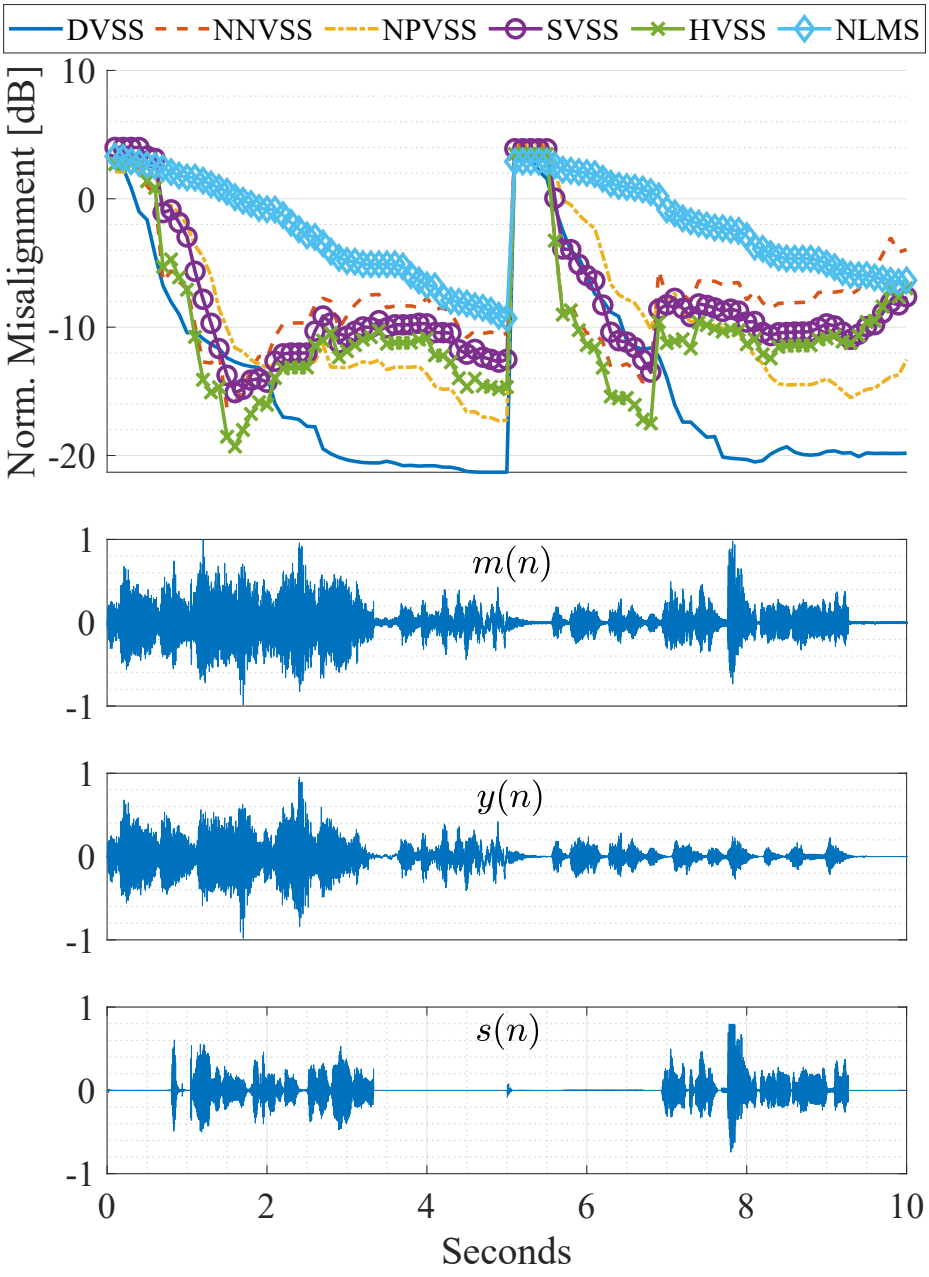

Figure 4.2: Convergence comparison to abrupt echo-path change that occurs after 5 s, while ESR and ENR values regularly change.

4.6 Conclusions

We have introduced a general framework for real-time adaptation control using deep learning. We first performed optimal VSS generation that is entirely non-parametric

and makes no acoustic assumptions via minimization of the filter misalignment. Second, the relation of the data and the optimal VSS was learned via a deep neural network. Finally, in real-time, the neural network yields a VSS estimate that is fed into the adaptive filter that continuously tracks the echo path. Experiments using 100 hours of real and synthetic data showed superior performance of the DVSS over the competition in AEC using the NLMS filter. In particular, the DVSS is preferable during double-talk in terms of echo cancellation and speech distortion, and characterized by faster convergence following abrupt echo-path changes.

Chapter 5

Deep Residual-Echo Suppression with A Tunable Tradeoff Between Signal Distortion and Echo Suppression

5.1 Introduction

Real-life telecommunication scenarios involve a conversation between two speakers that are located at near-end and far-end points. The near-end includes a microphone that captures the near-end signal, the far-end signal played by a loudspeaker, and background noises [SMH95]. The presence of acoustic echo can lead to degradation in intelligibility and quality of conversation, since the far-end speaker can hear his own voice while speaking. Conventional AECs do not model non-linearities in the echo path, and generally introduce a mismatch between true and estimated echo paths during convergence and re-convergence [BMS98]. This results in residual echo that must be suppressed by a dedicated system.

Deep learning has occupied a major role in AEC studies and showed enhanced performance compared to traditional methods [HK20], [FEKL20]. A recent study exploited LSTM networks to jointly obtain echo cancellation and to suppress noises and

reverberations [CSVH19]. Lee et al. [LSK15] cascaded a fully-connected neural network after a linear AEC system and evaluated the objective gain between the spectra amplitudes of the near-end and canceler output signals. Lei et al. [LCH$^+$19] exploited past and future temporal context to map the microphone and reference far-end signals to the desired speaker via a fully-connected neural network. Lately, deep learning and classic methods were jointly utilized in [MHZS20] and [ZTW19], where the latter activated convolutional recurrent networks to evaluate the real and imaginary parts of the near-end signal spectrogram.

In this study, we introduce an RES method with a dual-channel input and single-channel output UNet neural network that directly maps the outputs of a linear AEC to the desired near-end signal in the STFT domain. By utilizing the depth-wise separable convolution in every convolution layer of the UNet [GLA$^+$20], the system comprises 136 thousand parameters that consume 1.6 Giga FLOPS and 10 MB of memory, which makes it suitable for on-device integration. Also, the system meets the timing standards of the AEC challenge [SCS$^+$21], and more generally the constraints of hands-free communication systems [ETS16].

Even though competing models [HK20]–[ZTW19], [ZW18], [CSVH18] have shown promising results, the performance in real acoustic environments is still challenging. Furthermore, a tunable tradeoff between the level of residual-echo suppression and desired-signal distortion may benefit applications that vary in their specific tradeoff requirements. However, this feature is not enabled by design in existing approaches. We bridge these gaps as follows. First, we conduct experiments with over 160 hours of data that was acquired from the AEC challenge database [SCS$^+$21] and from independent recordings in real conditions. Second, a design parameter that allows dynamic balance between echo reduction and signal distortion is embedded in the UNet objective function that is minimized during the training process.

The performance of the proposed system is compared to two existing methods: Zhang and Wang [ZW18], where a bi-LSTM structure was utilized to model an ideal ratio mask for both AEC and RES, and Carbajal et al. [CSVH18], who introduced a multiple input fully-connected neural network RES system, fed with the linear AEC outputs and reference far-end signal to estimate a phase-sensitive mask. Experimental

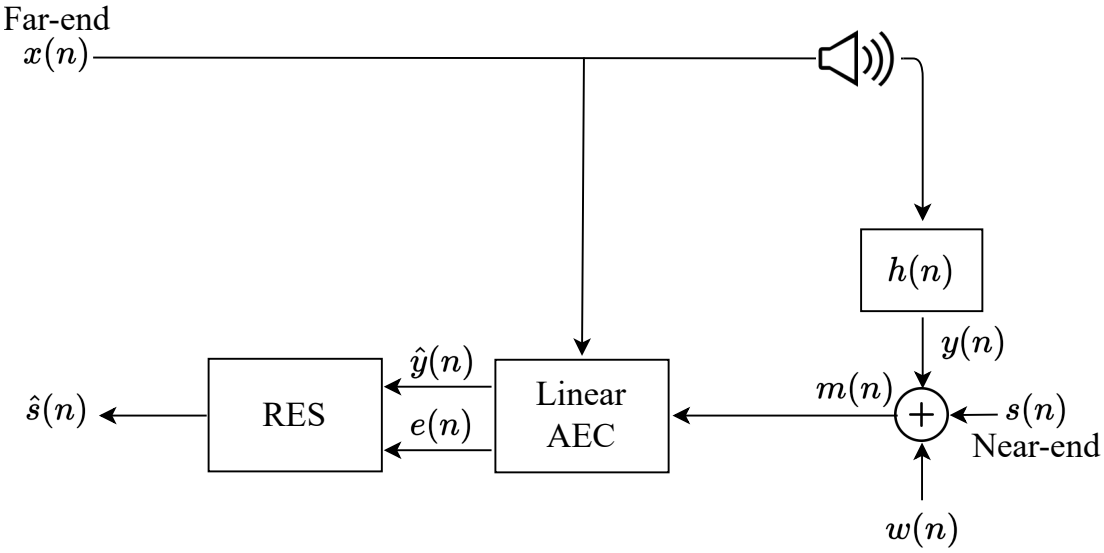

Figure 5.1: RES scenario and proposed system.

results show state-of-the-art performance in various real-life acoustic setups. Particularly, high generalization is demonstrated in a variety of environments, devices, speakers, and moving echo paths. High robustness is also achieved in extreme conditions of very low SERs, and the effect of the tunable design parameter is demonstrated.

The remainder of this chapter is organized as follows. In Section 5.2, we formulate the problem. In Section 5.3, we describe the proposed solution. In Section 5.4, we lay out the experimental setup. In Section 5.5, we present the experimental results. Finally, in Section 5.6, we draw conclusions.

5.2 Problem Formulation

Fig. 5.1 depicts the residual-echo suppression system. Let $x\left(n\right)$ denote the reference far-end signal and let $s\left(n\right)$ denote the desired near-end signal in time index n. The microphone signal, $m\left(n\right)$, is given by:

$$m\left(n\right) = s\left(n\right) + y\left(n\right) + w\left(n\right), \tag{5.1}$$

where the echo $y\left(n\right)$ is a reverberant non-linear modification of $x\left(n\right)$ and $w\left(n\right)$ represents the environmental and inherent system noises.

Before applying RES, a linear AEC system is applied to reduce the linear echo. The

AEC receives $m(n)$ as input and $x(n)$ as reference, and generates two output signals: $\hat{y}(n)$, the outcome of an adaptive filtering process that aims to estimate $y(n)$, and the adaptation error signal $e(n)$ that is given by:

$$e(n) = m(n) - \hat{y}(n). \tag{5.2}$$

From (5.1) and (5.2) we have:

$$e(n) = s(n) + (y(n) - \hat{y}(n)) + w(n). \tag{5.3}$$

The subsequent RES system aims to produce $\hat{s}(n)$ by suppressing the residual echo $y(n) - \hat{y}(n)$ without distorting the desired signal $s(n)$.

5.3 Deep Residual-echo Suppression with Tunable Trade-off

The proposed RES system comprises of a UNet neural network with two input channels and one output channel. The network is fed with the STFT amplitude of the linear AEC outputs and aims to recover the STFT amplitude of the desired near-end signal. The contracting and expansive paths of the UNet are each constructed of 5 convolution units. Every unit contains 2 concatenated and identical layers, where every layer consists of 2-D convolution, 2-D batch normalization, and a ReLU activation. Here, convolution is implemented in two parts; depth-wise convolution layer with a 3×3 kernel and padding of 1, followed by a separable convolution layer, to reduce computational load. During contraction, convolution units are followed by a max pooling layer, and during expansion, convolution units are preceded by an up-sampling layer, both of scaling factor 2. Skip connections are applied between matching pairs of contraction and expansion convolution units.

To exploit the powerful image segmentation abilities of the UNet [GLA+20], its channels are fed with a long temporal context of 300 ms that generates spectrogram images. During encoding, short filters jointly capture time-frequency local connec-

tions and produce numerous features that discriminate residual echo. During decoding, a similar convolution mechanism removes these echo signatures while preserving the desired signal. Long skip connections allow recovery of fine-grained details in the prediction, as features of the same dimension are reemployed from earlier layers, gradient flows directly via skip connections, which enhances optimization, and features are directly passed from encoder to decoder to recover spatial information lost during down-sampling.

A tunable design parameter $\alpha \geq 0$ is embedded in a custom loss function $J(\alpha)$ that is minimized during training:

$$J(\alpha) = \left\| \widehat{S}(f) - S(f) \right\|_2^2 + \alpha \cdot \left\| \widehat{S}(f) \right\|_2^2 + \sigma^2_{\widehat{S}(f)} \cdot \mathbb{I}_{\alpha>0} \ , \tag{5.4}$$

where $\hat{S}(f)$ and $S(f)$, respectively, represent the mini-batch predicted and desired spectra amplitudes in frequency bin f after normalization, as described in Section 5.4.2. During the training stage, $J(\alpha)$ is minimized while α penalizes $\left\| \widehat{S}(f) \right\|_2^2$, which allows a dynamic tradeoff between the levels of residual-echo suppression and desired-signal distortion of the system. When $\alpha = 0$, the error between the prediction and the near-end signal is minimized. However, when $\alpha > 0$, smaller prediction values are generated. This reduces the level of residual echo but compromises the level of desired-signal distortion. $\sigma^2_{\widehat{S}(f)}$ mitigates sub-band nullification that may occur when $\alpha \neq 0$.

The linear AEC system that precedes the UNet was made by Phoenix Audio Technologies. It employs a 150 ms filter length, converges after 5 s, and consumes 200 Kflops. Overall, the proposed end-to-end system, from the waveform input of the linear AEC to the waveform output of the RES, contains 136 thousand parameters that consume 1.6 Giga FLOPS and memory of 10 MB. The system meets timing constrains of hands-free communication [SCS+21], [ETS16] on the standard Intel Core i7-8700K CPU @ 3.7 GHz. Thus, on-device system integration is enabled, e.g., on the AM5749 processor by TI™ [TI19].

5.4 Experimental Setup

5.4.1 Database Acquisition

The SER and SNR captured by the microphone are SER=$10\log_{10}\left(\|s\left(n\right)\|_2^2/\|y\left(n\right)\|_2^2\right)$ and SNR=$10\log_{10}\left(\|s\left(n\right)\|_2^2/\|w\left(n\right)\|_2^2\right)$ in dB. Both measures are obtained using 50% overlapping time frames of 20 ms. Two data corpora were employed in this study; the AEC challenge database [SCS+21] used for training, and an independently recorded database used for both training and testing.

The AEC challenge database contains two new open sources of synthetic and real recordings. The synthetic data captures 100 hours of clean and noisy single talk and double talk periods. The real data was derived by a crowd sourcing effort that yielded 50 hours of audio clips, generated from $2,500$ real acoustic environments, audio devices, and human speaking in single and double talk scenarios that included changed and unchanged echo paths. SER levels were uniformly distributed between $[-10, 10]$ dB and SNR was randomly sampled between $[0, 40]$ dB in both data sources.

Also, independent recordings in real setups were acquired with sample frequency of 16 kHz to test the generalization of the system to unseen setups and its robustness to low levels of SER. The near-end signal was generated via a mouth simulator type 4227-A$^{\text{TM}}$ of Brüel&Kjaer, i.e., the near-end signal contained inherent system noise. The microphone and loudspeaker were either enclosed within a distance of 5 cm by speakerphones of type Spider$^{\text{TM}}$ MT503 or Quattro MT301$^{\text{TM}}$, or the echo was played externally by Logitech type Z120$^{\text{TM}}$ loudspeaker. The mouth simulator was placed in three positions located either 1, 1.5, or 2 m from the microphone, but was shifted only between recordings. Transitions in the echo path were generated by moving the external loudspeaker either 1, 1.5, or 2 m away from the microphone during recordings, producing 3 source-receiver positions. The data used for experiments was equally mixed between the TIMIT [GLF+93b] and the LibriSpeech [PCPK15] corpora. Recordings were performed in four different room sizes varied between a $3 \times 3 \times 2.5$ m^3 volume to a larger $5 \times 5 \times 4$ m^3 volume, and the reverberation time, i.e. RT_{60}, varied between $[0.3, 0.6]$ seconds. To create double talk utterances, near-end and far-end speakers were chosen randomly, then zero-padded to the same length and added in various SER

levels ranging in $[-20, -10]$ dB. The number of far-end single-talk, near-end single-talk, and double-talk utterances was identical. Technically, male and female speakers participated in a balanced manner, double-talk periods contained two different speakers, the training and test sets did not share the same speakers, and every speaker was both the far-end and near-end speaker. Overall, 11 hours of data were generated and equally split between the training and test databases. Both sets contained disjoint and balanced setups in terms of acoustic environments, devices, and speakers. SNR levels were distributed with mean values of 32 dB and std of 5 dB.

5.4.2 Data Processing, Training, and Testing

First, the linear AEC is applied to the microphone and reference signals. Then, 20 ms time frames are processed with 50% overlap. Each frame is represented by 161 frequency bins by taking the amplitude of a 320-point STFT. During training, the spectral data is normalized between 0 and 1, i.e., for every frequency bin between 1 and 161, the corresponding vector of time samples is subtracted by its minimum value and scaled by its dynamic range. These training statistics are reapplied to the test data. In training, disjoint batches of 30 frames, corresponding to 300 ms, are inserted to the dual input channel and single output channel of the UNet. Optimization is done by minimizing the loss function in eq. (5.4) with a learning rate of 0.0005, mini-batch size of 4, and 20 epochs using Adam optimizer [KB15]. Training duration was typically 1.5 hours per 10 hours of training data on an Intel Core i7-8700K CPU @ 3.7 GHz with two GPUs of type Nvidia GeForce RTX 2080 Ti. During testing, batches of 30 frames are inserted to the UNet with a step size of one. After the amplitude spectral prediction is generated, every frequency bin undergoes the inverse normalization described before using the training statistics. This result undergoes an inverse STFT using the error signal phase by employing the overlap-add method [GS97]. It should be noted that an artificial gain may be introduced by the RES, which is compensated as shown in [CSVH18].

Measure	Definition
ERLE	$10\log_{10}\left(\|e\left(n\right)\|_2^2/\|\hat{s}\left(n\right)\|_2^2\right)\Big\|_{\text{Far-end single-talk}}$
SAR	$10\log_{10}\left(\|s\left(n\right)\|_2^2/\|s\left(n\right)-\hat{s}\left(n\right)\|_2^2\right)\Big\|_{\text{Near-end single-talk}}$
SDR	$10\log_{10}\left(\|s\left(n\right)\|_2^2/\|s\left(n\right)-\hat{s}\left(n\right)\|_2^2\right)\Big\|_{\text{Double-talk}}$

Table 5.1: Performance measures for RES estimation.

5.4.3 Performance Measures

To evaluate performance we use the ERLE [ITU12] that measures the echo reduction between the noisy and enhanced signals when only far-end signal is present, and SAR that measures the distortion for near-end single-talk periods [VGF06]. For double-talk periods, we use the SDR [VGF06] that takes echo suppression and speech artifacts into account, and the PESQ [ITU01]. The performance measures are defined in Table 5.4.2. Besides the PESQ that is calculated over an entire utterance, these measures are calculated with 50% overlapping frames of 20 ms.

5.5 Experimental Results

We compare the performance of the proposed system with two competing RES methods in [ZW18], referring to its reported "AES+BLSTM" system, and [CSVH18]. All RES models are fed with the outputs of the same linear AEC discussed in this study. In all experiments, the linear AEC has converged and $\alpha = 0$ unless stated otherwise. Models are trained using both the entire AEC challenge data and independently recorded training data, which accumulates to over 155 hours. Performance measures are reported by their mean and std values across the entire 5.5 hours of the independently recorded test set, described in Section 5.4.1.

Results without change in echo path are given in Table 5.2 and with change in echo path are given in Table 5.3. It can be observed that our method outperforms competing methods in all the measures, while also attaining the lowest std. It is important to note that our method is least impeded by the changes in echo path,

	UNet	Zhang	Carbajal
PESQ	**3.61±0.24**	2.51±0.41	2.47±0.55
SDR	**7.1±0.8**	4.3±1.4	4.1±1.6
ERLE	**40.1±2.1**	35.7±3.3	21.5±3.6
SAR	**8.8±0.8**	4.8±1.1	4.5±1.1

Table 5.2: Performance with no echo-path change.

	UNet	Zhang	Carbajal
PESQ	**3.3±0.25**	2.35±0.45	2.05±0.7
SDR	**7±0.8**	2.71±1.9	2.8±1.65
ERLE	**39.7±1.9**	28.3±3.9	18±4
SAR	**8.8±0.95**	4.3±1.35	4.4±1.3

Table 5.3: Performance with echo-path change.

	UNet	Zhang	Carbajal
PESQ	**2.88±0.5**	2.02±0.8	1.91±0.95
SDR	**4.9±1.4**	2.6±2.1	1.1±1.7
ERLE	**31.8±2.9**	23.3±4.1	15.2±4.9
SAR	**8.5±1**	3.7±1.45	3.7±2.7

Table 5.4: Performance before convergence.

	$\alpha = 0$	$\alpha = 0.5$	$\alpha = 1$
PESQ	3.61±0.24	3.44±0.29	3.35±0.35
SDR	7.1±0.8	6.9±0.95	6.8±1.1
ERLE	40.1±2.1	41.9±2.2	43.5±2.2
SAR	8.8±0.8	8.4±0.8	8.2±0.9

Table 5.5: Performance for different values of α.

while the models in [ZW18] and [CSVH18] both deteriorate in this scenario. Thus, compared to the competing methods, the proposed system is characterized by better generalization ability to unseen real environments, devices, and speakers in extremely low SER levels between $[-20, -10]$ dB.

In the following, we investigate the performance before the linear AEC converges and during re-convergence, while addressing only unchanged echo paths. As shown in Table 5.4, performance is collectively impeded when the linear AEC has not converged. However, our method still shows leading performance that points out the high sensitivity of competing methods to converged echo approximation, while the UNet models the residual echo even from degraded measurements.

Next, we demonstrate the effect of α on the tradeoff between residual-echo suppression and desired-signal distortion levels. Again, only unchanged echo paths are considered. Results are presented in Table 5.5. It can be observed that increasing α leads to enhanced residual-echo suppression but at the expense of desired-signal distortion. However, an interesting conclusion is that for the given α values, the UNet does not severely degrade the quality of the desired signal, as suggested by the ERLE and SAR measures.

5.6 Conclusions

We have introduced an RES method based on a UNet neural network that receives the outputs of a linear AEC in the STFT domain. By using depth-wise separable convolution in the UNet layers, our system consists of 136 thousand parameters that require 1.6 Giga FLOPS and 10 MB of memory, which renders it adequate for on-device applications. This system satisfies hands-free communication timing standards on a standard CPU. In addition, we intergrate into the system a tunable tradeoff between echo suppression and signal distortion using a built-in design parameter. Experiments were conducted using 150 hours of synthetic and real recordings from the AEC challenge and 11 hours of real independent recordings. Results show state-of-the-art performance in terms of echo suppression and desired-signal distortion compared to competing methods, high generalization to various setups, and robustness to extremely low levels of

SER.

Chapter 6

Off-the-Shelf Deep Integration For Residual-Echo Suppression

6.1 Introduction

Hands-free speech communication often involves a conversation between two speakers located at near-end and far-end points. The near-end microphone captures the desired-speech signal, echo produced by a loudspeaker playing the far-end signal, and background noise. The acoustic coupling between the loudspeaker and the microphone may lead to degraded speech intelligibility in the far-end due to echo presence [SMH95]. Numerous AEC systems were proposed to reduce the echo of the far-end speaker's speech and preserve the near-end speaker[BGM$^+$01]. However, echos are often not eliminated by AEC systems and must be further reduced using RES systems.

RES can also be addressed as a speech separation (SS) [WC18] or speech enhancement (SE) [XDDL14] problem, where the echo is considered an interfering speech signal. We first fine-tune three off-the-shelf deep-learning-based systems: Our recently introduced RES system [ICB21a], a convolutional time-domain audio separation network called Conv-TasNet [LM19], and a denoiser develop by Facebook$^{\text{TM}}$ for SE [DSA20]. We show that the best-performing system of the three varies depending on the speech, echo, and noise levels. Second, we propose a real-time data-driven integration of these systems using a deep neural network that continuously tracks the best system based on

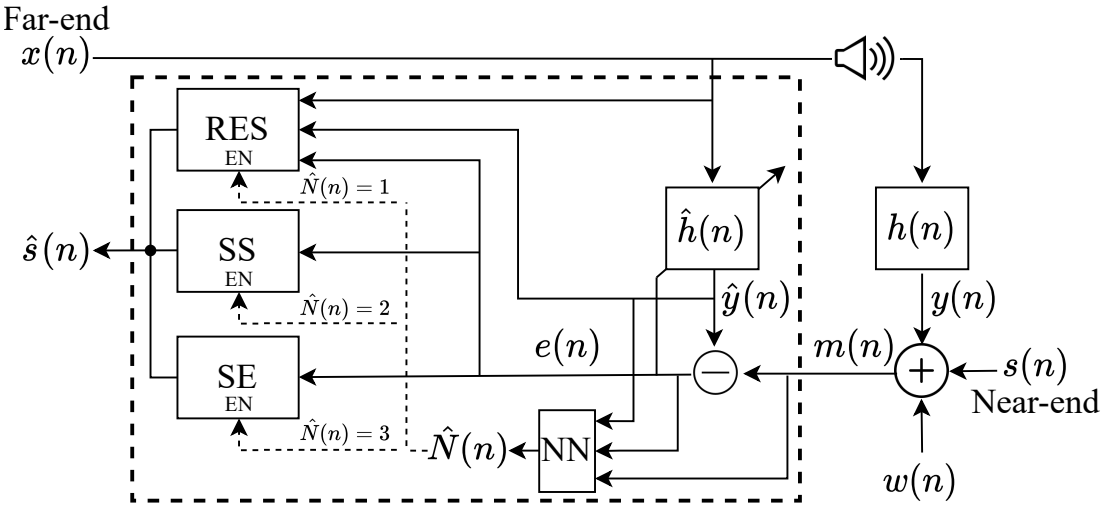

Figure 6.1: AEC scenario and proposed system integration.

single-talk and double-talk performance measures. Experiments with 100 hours of real and synthetic data show that the integrated system achieves better performance than each system in terms of echo cancellation and speech distortion across various acoustic setups in both single-talk and double-talk.

The remainder of this chapter is organized as follows. In Section 6.2, we formulate the problem. In Section 6.3, we describe the proposed solution. In Section 6.4, we lay out the experimental setup. In Section 6.5, we present the experimental results. Finally, in Section 6.6, we draw conclusions.

6.2 Problem Formulation

Figure 6.1 depicts the AEC scenario and proposed system. Let $s\,(n)$ be the near-end speech signal and let $x\,(n)$ be the far-end speech signal, where n is the time index. The microphone signal $m\,(n)$ is given by $m\,(n) = s\,(n) + y\,(n) + w\,(n)$, where $w\,(n)$ represents additive environmental and system noises and $y\,(n)$ is a nonlinear and reverberant echo that is generated from $x\,(n)$. First, an AEC system receives $m\,(n)$ as input and $x\,(n)$ as reference, and generates two signals: the echo estimate $\hat{y}\,(n)$, and the near-end signal estimate $e\,(n)$ given by:

$$e\,(n) = m\,(n) - \hat{y}\,(n) = s\,(n) + (y\,(n) - \hat{y}\,(n)) + w\,(n)\,. \tag{6.1}$$

A succeeding system aims to cancel the residual echo by eliminating $y(n) - \hat{y}(n)$, without distorting the speech $s(n)$. The neural network continuously selects and enables the best out of the RES, SS, and SE systems that interchangeably perform this task.

6.3 Off-the-Shelf Deep Integration For Residual-Echo Suppression

We consider three systems that were originally constructed and pre-trained for RES, SS, and SE. For RES, we employ an extension of the system in [ICB21a]. It comprises a U-net [RFB15] neural network that is fed with the STFT [GL84] amplitude of $e(n)$, $\hat{y}(n)$, and $x(n)$, and aims to recover the STFT amplitude of $s(n)$. The objective function that is minimized during training is the mean squared error between the neural network prediction and $s(n)$. The employed SS system is the waveform-based Conv-TasNet [LM19]. It comprises an encoder that maps the error mixture $e(n)$ to a high-dimensional representation, and a separation module that calculates a mask for each speech source in the mixture, i.e., the near-end speech and echo. Then, a decoder reconstructs the desired source from the masked features. A 1-dimensional convolutional autoencoder[GHD$^+$17] models the waveforms, and a temporal convolutional network separation module [LVRH16] estimates the masks based on the encoder output. The scale-invariant source-to-noise ratio [CDX20] is maximized during optimization, which is a modified version of the standard SDR [VGF06]. The SE system that is applied is the waveform-based neural network in [DSA20] that receives $e(n)$ and aims to cancel the residual echo and noise from it. The proposed model is based on an encoder-decoder architecture with skip-connections [MEG$^+$16]. It is optimized on both time and frequency domains using multiple loss functions. Namely, the ℓ_1 norm over the waveform together with a multi-resolution STFT loss over the spectrogram magnitudes are jointly minimized.

The proposed integrated system includes a deep neural network that receives the waveform representations of $e(n)$, $\hat{y}(n)$, and $m(n)$, and finds the best out of the RES, SS, and SE systems. The training stage of the neural network is done as follows. First, all three pre-trained systems are fine-tuned separately and independently with

an identical training database. Then, a validation set is propagated via each fine-tuned system, and two performance measures are extracted from each system. During single-talk periods, the ERLE [ITU12] is used. It measures echo reduction between the degraded and enhanced signals when only a far-end signal and noise are present. During double-talk, the objective DNSMOS metric is used [RGC21], which estimates objective human ratings. In [ICB21c], the DNSMOS has shown a strong correlation with echo suppression and speech preservation measures for the task of RES during double-talk. These measures are used to form a second training set as follows. Every time frame in the validation set is attached to a new categorical label, $N(n)$, from the set $\{1, 2, 3\}$, corresponding to the RES, SS, and SE systems. $N(n)$ is assigned to the index of the system with the highest ERLE during single-talk or highest DNSMOS during double-talk. This new dataset is used for training the neural network. The neural network architecture is waveform-based and follows the one in [ICB21b]. Still, its input layer is extended to three channels instead of two, and its final layer is concatenated to an additional softmax layer with three output neurons. In real-time, unseen data are propagated via the neural network that yields the index estimate of the best system, denoted by $\hat{N}(n)$, and the respective fine-tuned system is enabled to execute RES.

The proposed neural network contains 19 thousand parameters that consume 520 Million FLOPS and 42 KB of memory. Thus, its integration on hands-free devices is enabled, e.g., using the NDP120 neural processor by Syntiant$^{\text{TM}}$ [Syn21]. Timing constraints of hands-free communication on that processor are also met [ETS16]. The preceding AEC reduces linear echo with a standard NLMS [PCBG15] adaptive filter with a filter length of 150 ms and step size of 3×10^{-5}.

6.4 Experimental Setup

6.4.1 Database Acquisition

The AEC challenge database [CSP+21] is employed in this study. This corpus is sampled at 16 kHz and includes single-talk and double-talk periods, both with and without echo-path change. No echo-path change means no movement in the room during the recording, and echo-path change means either the near-end speaker or the device is

moving during the recording. The corpus includes 25 hours of synthetic data and 75 hours of real clean and noisy data. The SER, SNR, and ENR levels were distributed on $[-20, 10]$ dB, $[0, 40]$ dB, and $[0, 40]$ dB, respectively. These ratios are defined as:

$$\text{SER} = 10 \log_{10} \left(\|s\left(n\right)\|_2^2 / \|y\left(n\right)\|_2^2 \right), \tag{6.2}$$

$$\text{SNR} = 10 \log_{10} \left(\|s\left(n\right)\|_2^2 / \|w\left(n\right)\|_2^2 \right), \tag{6.3}$$

$$\text{ENR} = 10 \log_{10} \left(\|y\left(n\right)\|_2^2 / \|w\left(n\right)\|_2^2 \right), \tag{6.4}$$

and calculated with 50% overlapping time frames of 20 ms.

6.4.2 Data Processing, Traning, and Testing

The 100 hours of data is randomly split to create 80 hours of training, 10 hours of validation, and 10 hours of test sets. Each set is divided into 10 seconds segments that contain recordings in different setups. This leads to frequent re-convergence during transitions between segments, both without and with echo-path change. All sets are balanced to prevent a bias in the results, as described in [ICB21a]. During fine-tuning, each system maintains its original training configurations, but with an initial learning rate of 10 times smaller. For the NN, the data pre-processing follows [ICB21b], and the NN is trained with back-propagation through time with a learning rate of 5×10^{-4}, mini-batch of 40 ms, and 20 epochs, using Adam optimizer [KB15]. The minimized objective function is the categorical cross-entropy [ZM18] between the prediction and one-hot-vector encoding [KTC17] of the optimal-system index $N\left(n\right)$. Training duration was typically 15 minutes per 10 hours of data, and inference time was 12 ms per batch on an Intel Core i7-8700K CPU @ 3.7 GHz with two GPUs of type Nvidia GeForce RTX 2080 Ti.

6.4.3 Performance Measures

To separately measure echo suppression and speech distortion in double-talk, we respectively employ the recently introduced RESL and DSML [ICB21c]:

$$\text{RESL} = 10\log_{10}\left(\|r\left(n\right)\|_2^2/\|g\left(n\right)r\left(n\right)\|_2^2\right), \tag{6.5}$$

$$\text{DSML} = 10\log_{10}\left(\|\tilde{s}\left(n\right)\|_2^2/\|\tilde{s}\left(n\right)-g\left(n\right)s\left(n\right)\|_2^2\right), \tag{6.6}$$

where $g\left(n\right)=\widehat{s}\left(n\right)/e\left(n\right)$ is the time varying gain of the NN, $r\left(n\right)=e\left(n\right)-s\left(n\right)$ is the aligned noisy echo estimate, and $\tilde{s}\left(n\right)=\widehat{g}\left(n\right)s\left(n\right)$, where

$$\widehat{g}\left(n\right)=\left\langle g\left(n\right)s\left(n\right),s\left(n\right)\right\rangle/\|s\left(n\right)\|_2^2, \tag{6.7}$$

and $\left\langle\cdot,\cdot\right\rangle$ is the internal product between two vectors. The DNSMOS [RGC21] is used again during double-talk to assess speech quality for human perception. During single-talk, the echo suppression level is quantified using the ERLE [ITU12]. The DSML, RESL, and ERLE are calculated with 50% overlapping time frames of 20 ms, and the DNSMOS is applied with the API provided by Microsoft$^{\text{TM}}$.

6.5 Experimental Results

The integrated system, denoted as "INT", is compared against the particular RES, SS, and SE systems. In Tables 6.1–6.3, performance measures are given with their mean and standard deviation (std) values in the format mean±std. In Figures 6.2–6.13, only the average values of the performance measures are shown. For all the measures, higher mean and lower std indicate better performance. Convergence of the linear AEC is assumed if the normalized misalignment was lower than -10 dB for a given echo path [PCBG15]. The results are derived with respect to the entire test set.

Results for no echo-path change are given in Table 6.1, and for echo-path change are shown in Table 6.2, both after convergence. In Table 6.3, results for no echo-path change before convergence are reported. Comparing the RES, SS, and SE systems, we may conclude that the SE system obtains better average performance during double-talk

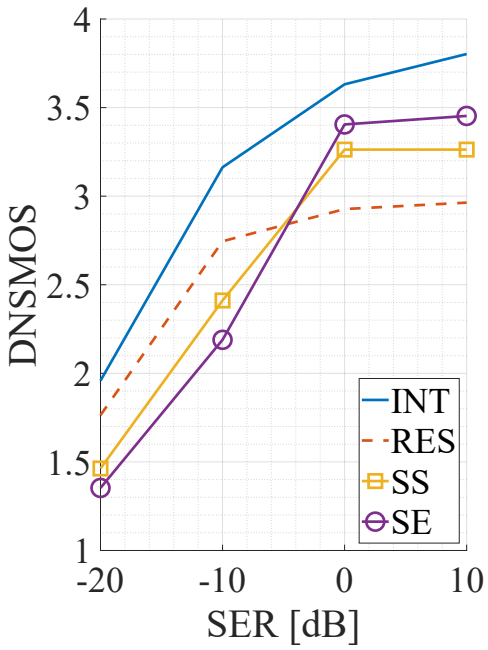

Figure 6.2: DNSMOS versus SER [dB] for no echo-path change scenarios.

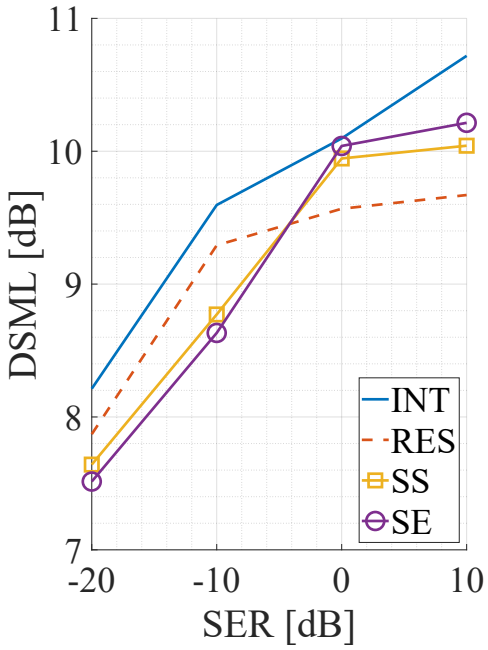

Figure 6.3: DSML [dB] versus SER [dB] for no echo-path change scenarios.

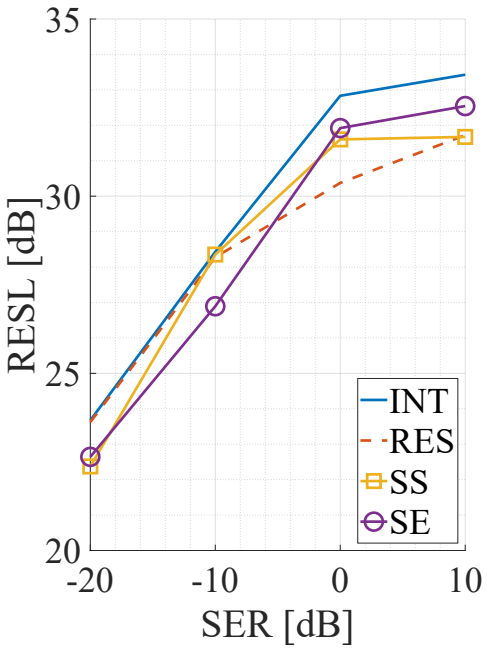

Figure 6.4: RESL [dB] versus SER [dB] for no echo-path change scenarios.

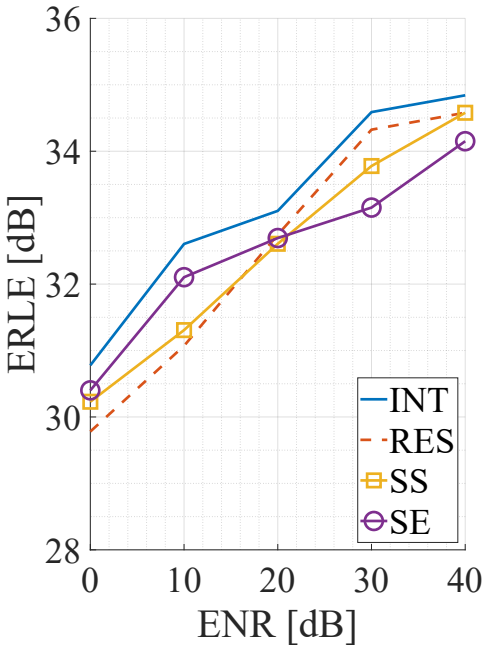

Figure 6.5: ERLE [dB] versus ENR [dB] for no echo-path change scenarios.

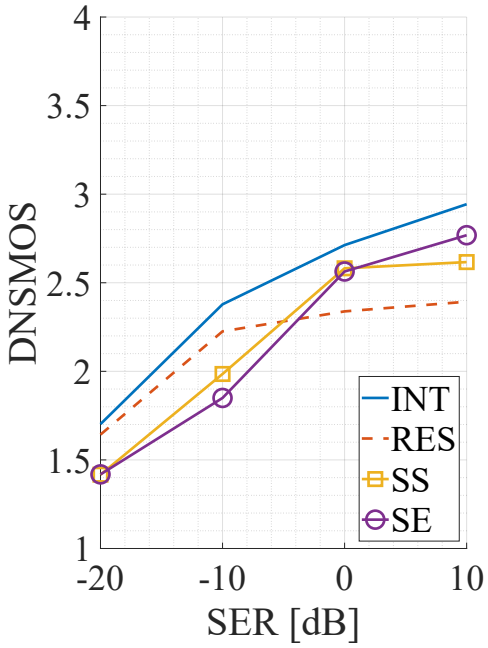

Figure 6.6: DNSMOS versus SER [dB] for echo-path change scenarios.

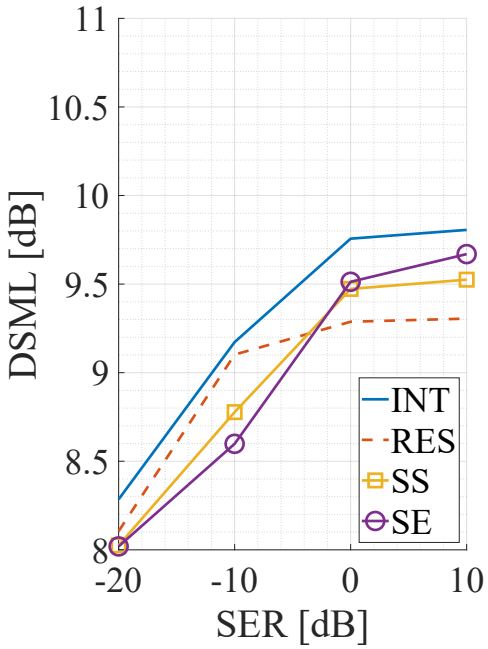

Figure 6.7: DSML [dB] versus SER [dB] for echo-path change scenarios.

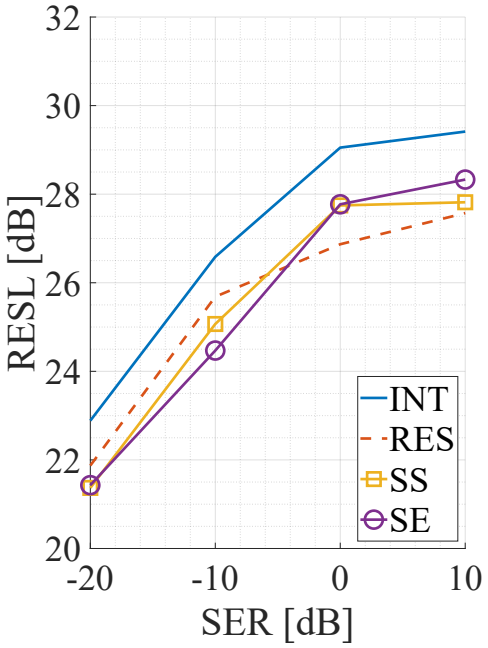

Figure 6.8: RESL [dB] versus SER [dB] for echo-path change scenarios.

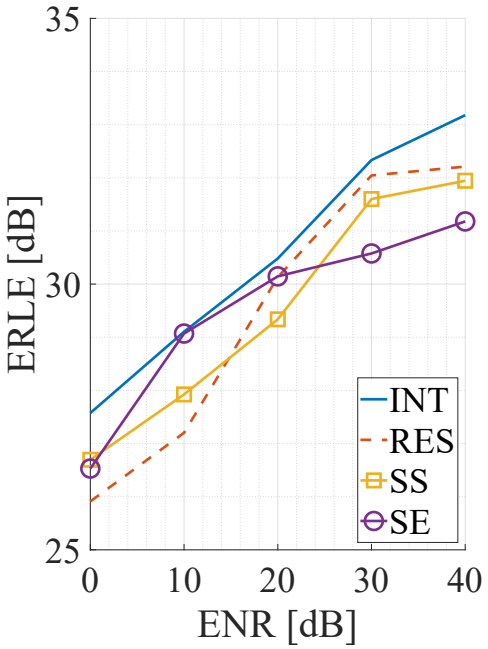

Figure 6.9: ERLE [dB] versus ENR [dB] for echo-path change scenarios.

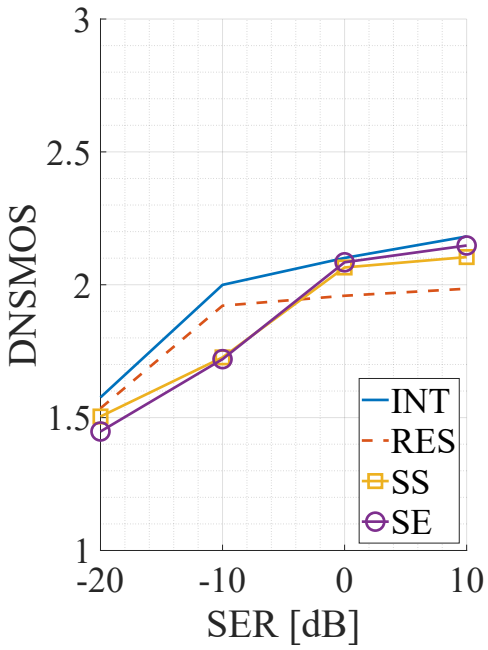

Figure 6.10: DNSMOS versus SER [dB] before linear AEC convergence.

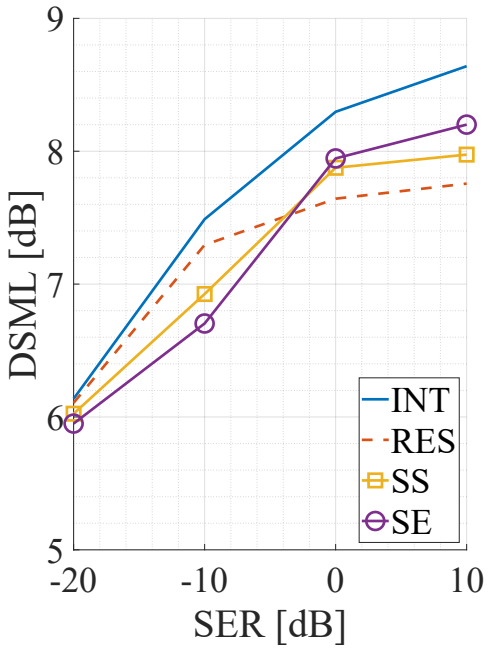

Figure 6.11: DSML [dB] versus SER [dB] before linear AEC convergence.

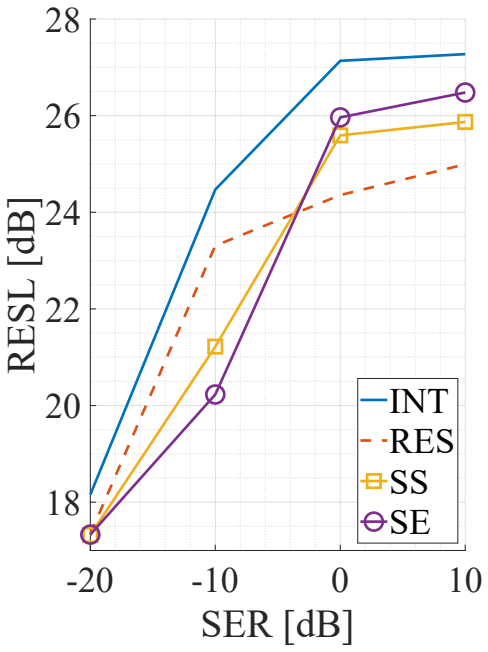

Figure 6.12: RESL [dB] versus SER [dB] before linear AEC convergence.

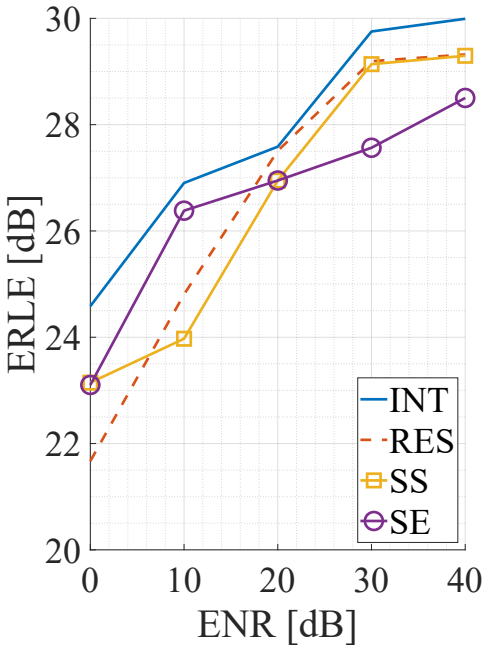

Figure 6.13: ERLE [dB] versus ENR [dB] before linear AEC convergence.

	INT	RES	SS	SE
DNSMOS	**3.1±0.8**	2.4±0.5	2.5±0.8	2.6±1.0
DSML	**9.6±1.0**	8.7±0.8	9.1±1.1	9.2±1.2
RESL	**29.6±4.5**	27.5±3.5	28.3±4.3	28.5±4.6
ERLE	**33.1±1.6**	32.5±2.0	32.2±1.7	32.1±1.4

Table 6.1: Performance with no echo-path change.

	INT	RES	SS	SE
DNSMOS	**2.4±0.5**	1.8±0.3	2.0±0.6	2.1±0.6
DSML	**9.2±0.7**	8.4±0.6	8.8±0.7	8.9±0.8
RESL	**27.0±3.0**	25.3±2.5	25.7±3.0	25.8±3.2
ERLE	**30.5±2.2**	28.5±2.8	28.2±2.3	28.3±1.8

Table 6.2: Performance with echo-path change.

	INT	RES	SS	SE
DNSMOS	**2.1±0.2**	1.7±0.2	1.8±0.3	1.9±0.3
DSML	**7.6±1.1**	7.1±0.7	7.2±0.9	7.4±1.1
RESL	**24.2±4.4**	22.5±3.0	22.8±3.5	23.0±4.4
ERLE	**27.7±2.2**	26.0±3.2	25.8±2.8	25.4±2.0

Table 6.3: Performance before convergence.

periods in terms of echo cancellation as shown by the RESL, desired-speech distortion as shown by the DSML, and speech quality as demonstrated by the DNSMOS. However, the SE system also obtains the highest std values in double-talk, making it less stable than competition. The RES system is favorable in single-talk echo cancellation with a higher average ERLE, but is also the least stable with the highest std value. These observations also hold for echo-path change and pre-convergence scenarios, but with an expected degradation in the values of all performance measures. Thus, neither the RES, SS, nor SE system is optimal across all measures and acoustic scenarios in

terms of higher average performance and in terms of lower std values. The proposed integrated system outperforms each individual system on average across all measures and scenarios during both single-talk and double-talk periods.

Average performance is also analyzed for various levels of SER during double-talk and multiple levels of ENR during single-talk. Results for segments without and with echo-path change are given in Figures 6.2–6.5 and 6.6–6.9, respectively, both after convergence. Results for segments with no echo-path change before convergence are shown in Figures 6.10–6.13. During double-talk, the SE system outperforms the RES and SS systems when SER levels are high, and the RES system is preferable when SER levels are low. During single-talk, the RES system obtains higher performance when ENR levels are high, and the SE system is preferable for low levels of ENR. These observations remain across all measures and also for echo-path change and pre-convergence scenarios, but again with an expected overall decrease on the average performance. These results reaffirm that the best performing system varies with speech, echo, and noise levels, and supports previous claims that no individual system can be considered best under all acoustic conditions.

A possible explanation for the behavior of the three systems with respect to acoustic conditions is suggested. The SE system is better suited to handle high SER levels since the noisy echo is significantly attenuated with respect to speech and appears as a noisy interference. Similarly, as ENR decreases, the SE system is successful since the echo is mainly screened by noise. The RES system is preferable when the SER level is low, since it was trained to detect residual-echo signatures that are mixed with speech. Likewise, when ENR levels are high, the residual-echo dominates the signal and can be successfully recognized and suppressed by the RES system. The proposed integrated system outperforms the RES, SS, and SE systems across all speech, echo, and noise levels, in both no echo-path change, echo-path change, and pre-convergence scenarios. Based on the presented results, it can be concluded that the proposed NN can estimate which system is best in real-time for various acoustic conditions during both single-talk and double-talk periods, and that the integrated system is better on average than each of its three components.

6.6 Conclusions

We have introduced a real-time data-driven system integration framework and applied it to the task of RES. This integration comprises three deep learning-based systems originally constructed and pre-trained for RES, SS, and SE. After fine-tuning all three systems and showing that none of these systems can be considered best for RES, we developed a deep NN that continuously selects the best of the RES, SS, and SE systems and enables it to perform RES. Using 100 hours of real and synthetic recordings, we showed that the NN can estimate the best system in real time and that the proposed integrated system outperforms, on average, each of the three individual systems in terms of echo cancellation and speech distortion during both single-talk and double-talk periods.

Chapter 7

Objective Metrics to Evaluate Residual-Echo Suppression During Double-Talk

7.1 Introduction

Hands-free communication often involves a conversation between two speakers located at near-end and far-end points. The near-end microphone can capture the desired-speech signal and two interfering signals: nonlinear echo produced by a loudspeaker playing the far-end signal, and background noises [BMS98, BGM$^+$01]. The acoustic coupling between the loudspeaker output and the microphone may lead to degraded speech intelligibility in the far-end due to echo presence [SMH95]. The most challenging scenarios are double-talk periods, when the desired speech and echo are captured by the microphone at the same time. To combat that, numerous NLAEC systems were proposed to remove the nonlinear echo and to preserve the near-end speech [GFLBJ03, ME12, CSAR$^+$13, HHK19, ICB21b]. However, often there is still a mismatch between true and estimated echo paths, especially during the NLAEC convergence and re-convergence [BG95a, MEB10]. As a result, the echo is not eliminated and the NLAEC should be followed by an RES system.

Human perception of speech quality is optimally evaluated using human subjective

evaluation [RBP+19]. Lately, the objective DNSMOS metric has been proposed to estimate human ratings and has shown great accuracy [RGC21]. Regarding the task of RES, speech quality during double-talk is traditionally evaluated using the objective SDR metric [VGF06], e.g., in [CSVH18, DDBW19, PP20, CXCL20, Fan20b, Fan20a]. Unfortunately, the SDR is affected by both desired-speech distortion and residual-echo presence, which renders it unreliable in predicting the DNSMOS and unreliable in predicting human perception of speech quality [RGC21].

This paper introduces two objective metrics that separately evaluate the DSML and the RESL during double-talk. Considering the RES system as a time-varying gain, the DSML is obtained by applying that gain to the desired speech and substituting the outcome in the definition of the SDR. The RESL is obtained by subtracting the desired speech from the double-talk segment and calculating the ratio of the noisy residual-echo before and after the gain is applied to it. To evaluate these metrics, we employ a deep learning-based RES system that also embeds a design parameter [ICB21a]. Experiments are done with 280 hours of real and simulated recordings in various scenarios and in high and low levels of echo and noise. Results show that the DSML and RESL have high correlation with human perception according to the DNSMOS, and high generalization to various setups, which renders them more suitable for speech quality evaluation than the SDR. We further investigate the empirical relation between tuning the design parameter and the DSML-RESL tradeoff it creates. Based on this relation, we offer a practical scheme for tuning the design parameter during training to optimally cope with dynamic system requirements.

The remainder of this chapter is organized as follows. In Section 7.2, we formulate the problem. In Section 7.3, we describe the proposed objective metrics. In Section 7.4, we revisit the tunable design parameter. In Section 7.5, we lay out the experimental setup. In Section 7.6, we present the experimental results. Finally, in Section 7.7, we draw conclusions.

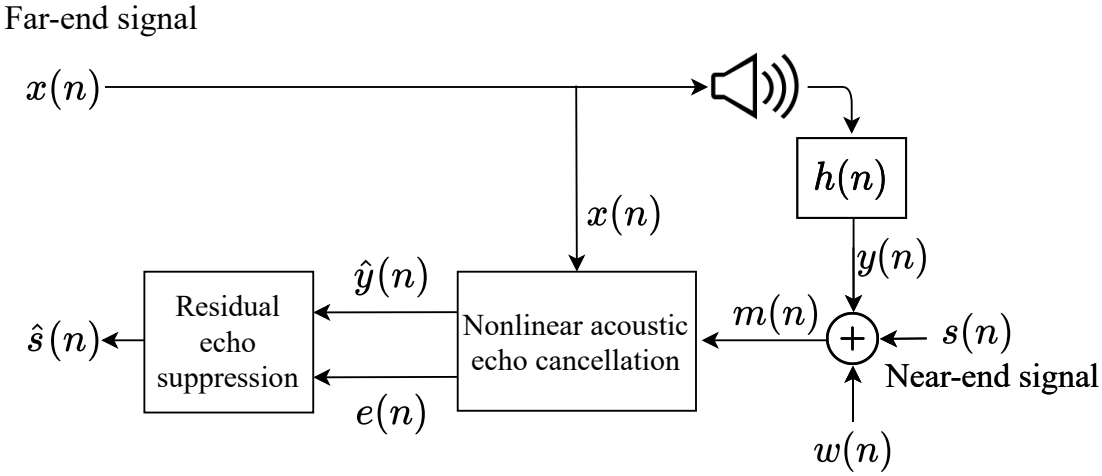

Figure 7.1: Residual-echo suppression scenario.

7.2 Problem Formulation

Figure 7.1 depicts the RES scenario. Let $s\,(n)$ be the desired near-end speech signal and let $x\,(n)$ be the far-end speech signal. The near-end microphone signal $m\,(n)$ is given by:

$$m\,(n) = s\,(n) + y\,(n) + w\,(n)\,, \tag{7.1}$$

where $w\,(n)$ represents additive environmental and system noises and $y\,(n)$ is a reverberant echo that is nonlinearly generated from $x\,(n)$. Before applying RES, the NLAEC system introduced in [ICB21b] is applied to reduce nonlinear echo. The NLAEC receives $m\,(n)$ as input and $x\,(n)$ as reference, and generates two signals: the echo estimate $\widehat{y}\,(n)$, and the desired-speech estimate $e\,(n)$, given by

$$e\,(n) = m\,(n) - \widehat{y}\,(n) = s\,(n) + (y\,(n) - \widehat{y}\,(n)) + w\,(n)\,. \tag{7.2}$$

The goal of the RES system is to suppress the residual echo $y\,(n) - \widehat{y}\,(n)$ without distorting the desired-speech signal $s\,(n)$.

7.3 DSML and RESL

To derive the DSML and RESL, a deep learning-based RES system is considered as a time-varying gain. During double-talk, $e(n) \neq 0$ and the gain is given by

$$g(n) = \left.\frac{\widehat{s}(n)}{e(n)}\right|_{\text{Double-talk}}. \tag{7.3}$$

Before introducing the DSML and RESL metrics, the SDR and its drawbacks are examined. According to [VGF06], the SDR is defined as

$$\begin{aligned} \text{SDR} &= 10\log_{10}\left.\frac{\|s(n)\|_2^2}{\|s(n)-\widehat{s}(n)\|_2^2}\right|_{\text{Double-talk}} \\ &= 10\log_{10}\left.\frac{\|s(n)\|_2^2}{\|s(n)-g(n)\,e(n)\|_2^2}\right|_{\text{Double-talk}}. \end{aligned} \tag{7.4}$$

The SDR is affected by both the desired-speech distortion and residual-echo presence, and makes no distinction between cases in which $g(n)\,e(n)$ comprises distortion-free speech and echo, or distorted speech without echo. Thus, the SDR does not correlate well with human ratings [RGC21], since these scenarios clearly exhibit different human perception ratings and different DNSMOS values. A distinction between desired-speech distortion and residual-echo suppression is extremely valuable for evaluating RES during double-talk. Hence, we propose two objective metrics by applying $g(n)$ separately to the desired speech and noisy residual-echo estimate.

Formally, the DSML is calculated similarly to the SDR, but $g(n)$ is applied only to the desired speech $s(n)$:

$$\text{DSML} = 10\log_{10}\left.\frac{\|\tilde{s}(n)\|_2^2}{\|\tilde{s}(n)-g(n)\,s(n)\|_2^2}\right|_{\text{Double-talk}}. \tag{7.5}$$

The RESL is derived by estimating the noisy residual-echo as $r(n) = e(n) - s(n)$, and evaluating the following ratio:

$$\text{RESL} = 10\log_{10}\left.\frac{\|r(n)\|_2^2}{\|g(n)\,r(n)\|_2^2}\right|_{\text{Double-talk}}. \tag{7.6}$$

Note that the RES system may introduce a constant attenuation that leads to an

artificial desired-speech distortion in the DSML. To ensure the DSML is invariant to that attenuation, it is compensated as in [CSVH18]. Explicitly, $\tilde{s}(n) = \widehat{g}(n)\, s(n)$, where:

$$\widehat{g}(n) = \frac{\langle g(n)\, s(n), s(n) \rangle}{\| s(n) \|_2^2}. \tag{7.7}$$

7.4 RES System with a Design Parameter

To evaluate the performances of the DSML and RESL, we employ a deep learning-based RES system that embeds a tunable design parameter [ICB21a]. This system comprises a UNet neural network [RFB15] with two input channels and one output channel. The network is fed with the STFT [GL84] amplitude of the NLAEC outputs and aims to recover the STFT amplitude of the desired speech. The design parameter $\alpha \geq 0$ is embedded in a custom loss function $J(\alpha)$ that is minimized during training:

$$J(\alpha) = \left\| \widehat{S}(f) - S(f) \right\|_2^2 + \alpha \cdot \left\| \widehat{S}(f) \right\|_2^2 + \sigma^2_{\widehat{S}(f)} \cdot \mathbb{I}_{\alpha>0} \ , \tag{7.8}$$

where $\widehat{S}(f)$ and $S(f)$, respectively, represent the desired-speech prediction and ground truth spectra amplitudes, $\sigma^2_{\widehat{S}(f)}$ denotes the variance of $\widehat{S}(f)$, and $\mathbb{I}_{\alpha>0}$ equals 1 when $\alpha > 0$ and 0 otherwise. During the training stage, $J(\alpha)$ is minimized while α penalizes $\| \widehat{S}(f) \|_2^2$, which allows a dynamic tradeoff between the desired-speech distortion and residual-echo suppression of the system, namely between the DSML and RESL. When $\alpha = 0$, the error between the desired-speech prediction and ground truth is minimized. However, when $\alpha > 0$, smaller prediction values are generated. This reduces the level of residual echo but compromises the level of desired-speech distortion. $\sigma^2_{\widehat{S}(f)}$ mitigates sub-band nullification that may occur when $\alpha > 0$. Note that α and the DSML-RESL tradeoff it creates can be tuned during the training process.

7.5 Experimental Setup

7.5.1 Database Acquisition

Two data corpora were employed in this study; the AEC-challenge database [CSP+21], and a database recorded in our lab, both sampled at 16 kHz. These corpora consider single-talk and double-talk periods both without and with echo-path change. In the former there is no movement during the recording, and in the latter either the near-end speaker or device are moving during the recording. In [CSP+21], two open sources of synthetic and real recordings are introduced. The synthetic data includes 100 hours, and the real data contains 140 hours of audio clips, generated from 5000 hands-free devices that are used in various acoustic environments. In both real and synthetic cases, SER and SNR levels were distributed on $[-10, 10]$ dB and $[0, 40]$ dB, respectively. Additional real recordings were conducted in our lab to test the generalization of the DSML and RESL to unseen setups and their robustness to extremely low levels of SERs. This database is fully described in [ICB21a]. For completion, it contains 40 hours of recordings from the TIMIT [GLF+93b] and LibriSpeech [PCPK15] corpora with SNR levels of 32 ± 5 dB and SER levels distributed on $[-20, -10]$ dB.

The SER is defined as $\text{SER}=10\log_{10}\left(\|s\left(n\right)\|_2^2/\|y\left(n\right)\|_2^2\right)$ and the SNR is defined as $\text{SNR}=10\log_{10}\left(\|s\left(n\right)\|_2^2/\|w\left(n\right)\|_2^2\right)$ in dB, each is calculated with 50% overlapping time frames of 20 ms.

7.5.2 Data Processing, Training, and Testing

The real and synthetic data from [CSP+21] was randomly split to create 185 hours of training set and 45 hours of validation set. The test set contains only real data that includes the remaining 10 hours from [CSP+21] and all 40 hours from [ICB21a]. Each set was divided into 10 seconds segments that contain recordings in different setups. This leads to frequent re-convergence during transitions between segments, both with and without echo-path change. These sets are balanced to prevent bias in the results, as detailed in [ICB21a]. The NLAEC system, which is also deep learning-based [ICB21b], and the succeeding RES system [ICB21a], were trained separately. During testing, in accordance with Section 7.3, the artificial gain that may be introduced by the RES

system is compensated as in [VGF06, CSVH18] before deriving the DSML and RESL.

7.5.3 Performance Measures

We employ additional metrics to evaluate RES. The ERLE [ITU12] measures echo reduction between the degraded and enhanced signals when only echo and noise are present:

$$\text{ERLE} = 10\log_{10}\left.\frac{\|e\left(n\right)\|_2^2}{\|\widehat{s}\left(n\right)\|_2^2}\right|_{\text{Far-end single-talk}}. \tag{7.9}$$

The SAR [VGF06] measures the desired-speech distortion during near-end single-talk periods:

$$\text{SAR} = 10\log_{10}\left.\frac{\|s\left(n\right)\|_2^2}{\|s\left(n\right)-\widehat{s}\left(n\right)\|_2^2}\right|_{\text{Near-end single-talk}}. \tag{7.10}$$

The PESQ [ITU01] metric, which correlates well with the DNSMOS [RGC21], is used in double-talk. The SAR and SDR are compensated as the DSML in eq. (7.5).

7.6 Experimental Results

The performance metrics are evaluated using the RES system and are calculated with 50% overlapping frames of 20 ms. The metrics are reported by their mean and standard deviation (std) values in Table 7.1, and by their mean in Figures 7.2–7.13, with respect to the test set specified in each experiment. For all metrics, higher mean and lower std indicate a better performance. In our study, the convergence of the NLAEC follows the definitions in [ICB21b, PCBG15], and the DNSMOS is calculated using the API provided by Microsoft [RGC21].

First, we explore the correlation of the DSML and RESL with the DNSMOS using PCC [BCHC09] and SRCC [Gau01], as done in [RGC21, SCS$^+$21]. This experiment includes segments without echo-path change after convergence for $\alpha \in \{0, 0.25, 0.5, 0.75, 1\}$, and the results are shown in Figures 7.2–7.3. The conclusion drawn in [RGC21] is reaffirmed in this study, i.e., the SDR does not correlate well with the DNSMOS, as the PCC and SRCC mean values are below 0.26 for all α. On the contrary, the DSML and

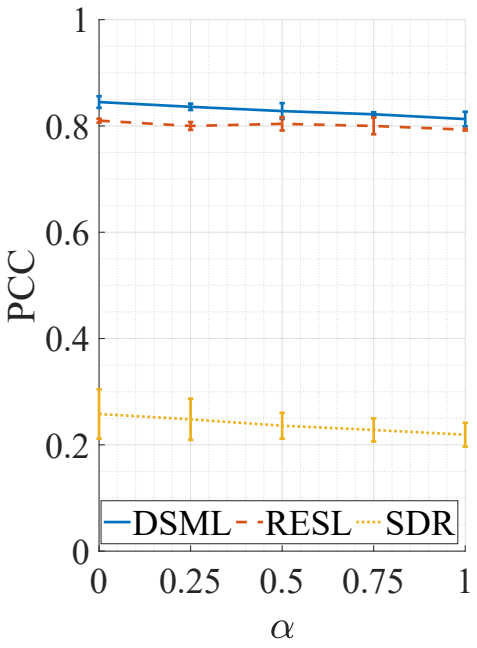

Figure 7.2: PCC of DNSMOS with the DSML, RESL, and SDR metrics.

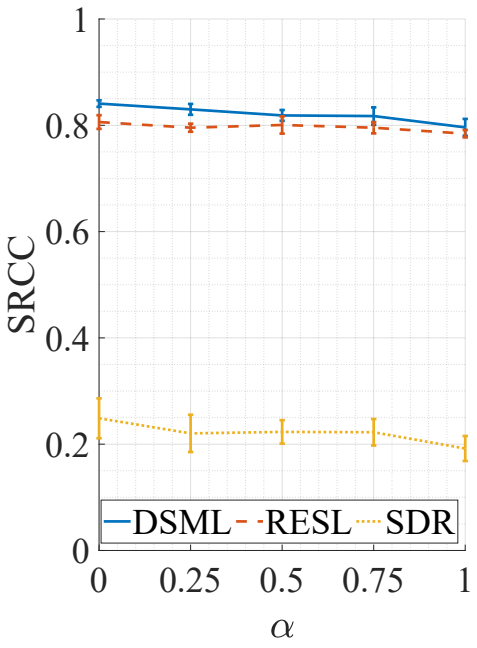

Figure 7.3: SRCC of DNSMOS with the DSML, RESL, and SDR metrics.

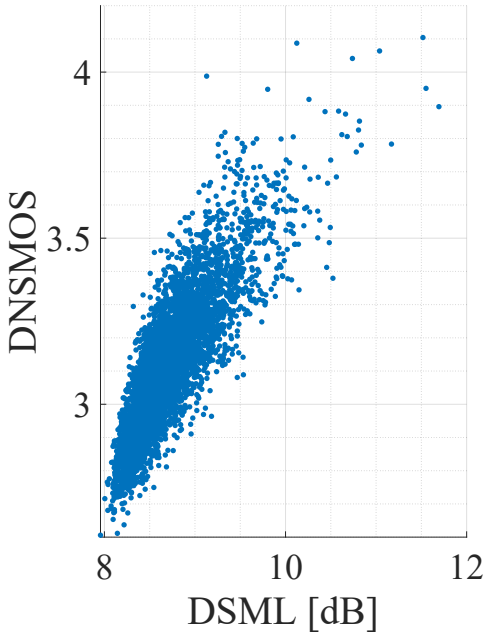

Figure 7.4: Scatter plots of DNSMOS versus the proposed DSML metric.

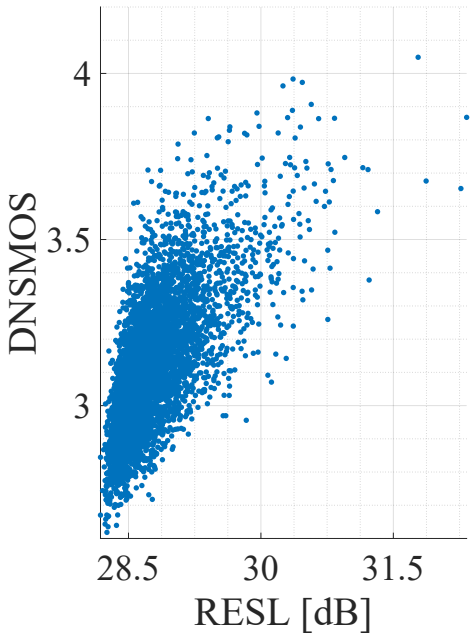

Figure 7.5: Scatter plots of DNSMOS versus the proposed RESL metric.

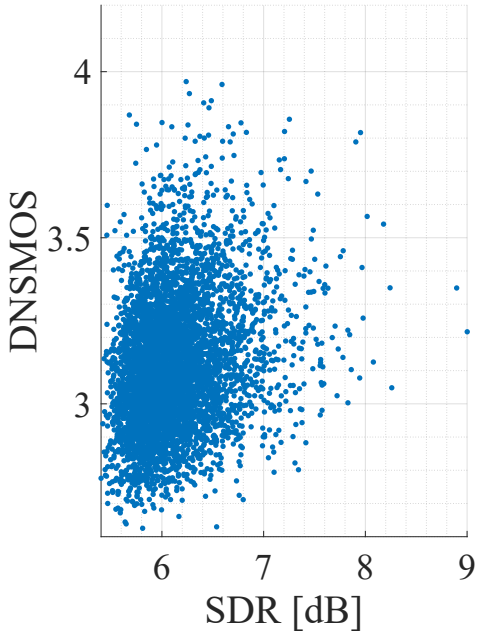

Figure 7.6: Scatter plots of DNSMOS versus the SDR metric.

RESL are highly correlated with the DNSMOS, with mean correlation scores between 0.78 and 0.85 for all α. Also, compared to the SDR, the DSML and RESL correlations are relatively more consistent across α values, as inferred from their lower std values. To visualize these correlations, Figures 7.4–7.6 depict scatter plots of the DNSMOS versus the DSML, RESL, and SDR metrics for random sample values with $\alpha = 0$. These plots validate the poor correlation between the DNSMOS and SDR, and the high correlation between the DNSMOS and the DSML and RESL. Conclusively, the DSML and RESL are better correlated with human perception and speech quality evaluation.

All performance metrics are evaluated in Table 7.1 with $\alpha = 0$. Separate results are shown for segments without and with echo-path change after NLAEC convergence, and for segments before convergence. The DSML and RESL are consistent with all other metrics, which degrade when shifting from no echo-path change to echo-path change scenarios, and further degrade when considering segments before convergence. This also implies high generalization of the DSML and RESL to various setups. The DSML is consistently higher than the SDR, as expected, since the definition in eq. (7.4) also considers echo and noise in the denominator. Also, the DSML is lower than the

	No echo-path change	Echo-path change	Before convergence
DNSMOS	3.12 ± 0.2	2.91 ± 0.3	2.56 ± 0.6
DSML	8.73 ± 0.4	8.34 ± 0.5	6.97 ± 0.7
RESL	29.1 ± 3.7	25.9 ± 4.4	22.1 ± 5.6
SDR	6.13 ± 0.4	5.94 ± 0.6	5.57 ± 0.8
PESQ	3.58 ± 0.2	3.35 ± 0.5	3.18 ± 0.6
SAR	9.88 ± 0.4	9.69 ± 0.5	9.51 ± 0.6
ERLE	33.2 ± 3.1	29.1 ± 4.2	26.4 ± 5.1

Table 7.1: Performance measures in various scenarios with $\alpha = 0$.

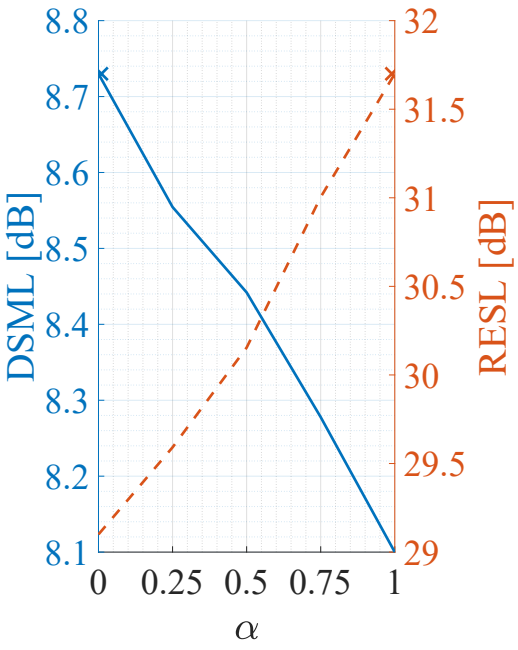

Figure 7.7: DSML-RESL tradeoff for various values of α in no echo-path change scenarios.

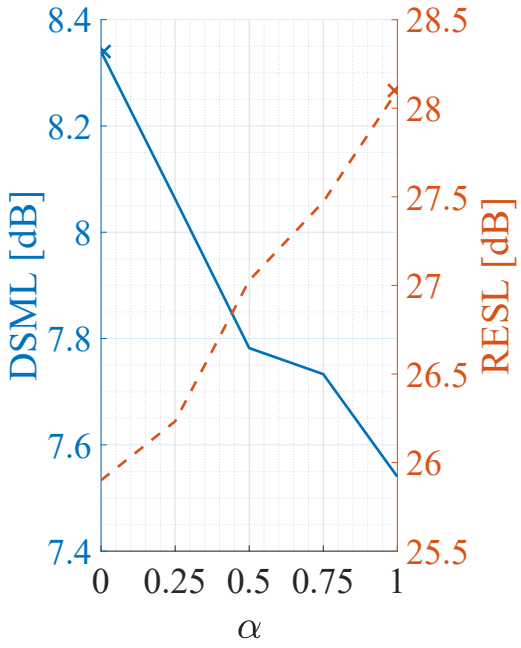

Figure 7.8: DSML-RESL tradeoff for various values of α in echo-path change scenarios.

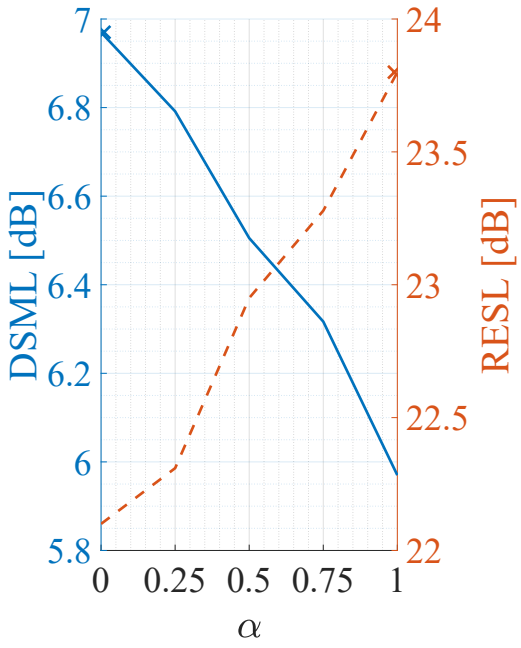

Figure 7.9: DSML-RESL tradeoff for various values of α before linear AEC convergence.

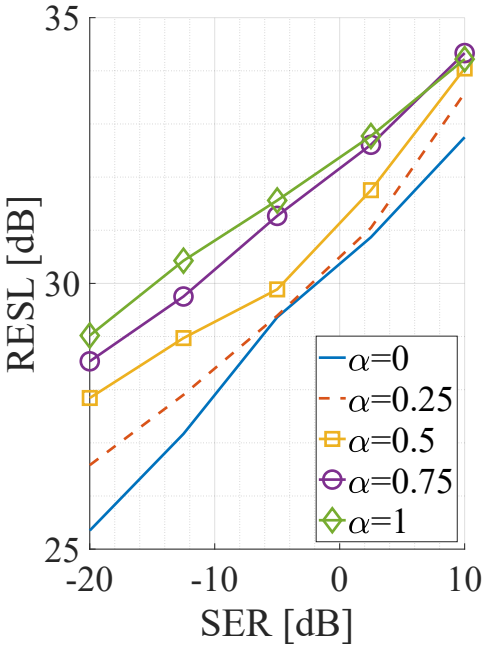

Figure 7.10: RESL for various values of α in different SER levels.

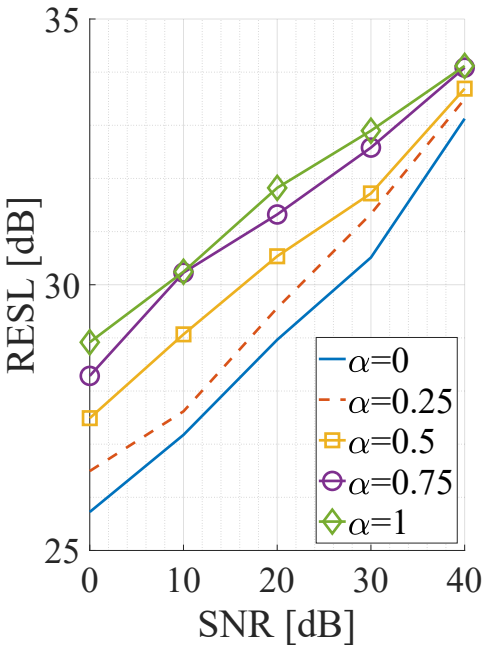

Figure 7.11: RESL for various values of α in different SNR levels.

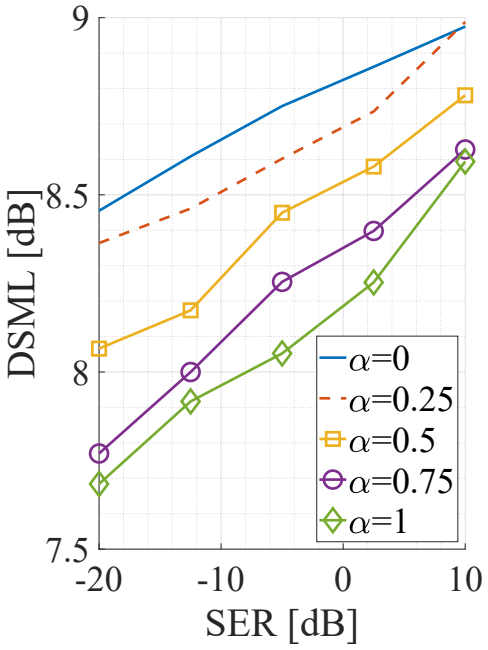

Figure 7.12: DSML for various values of α in different SER levels.

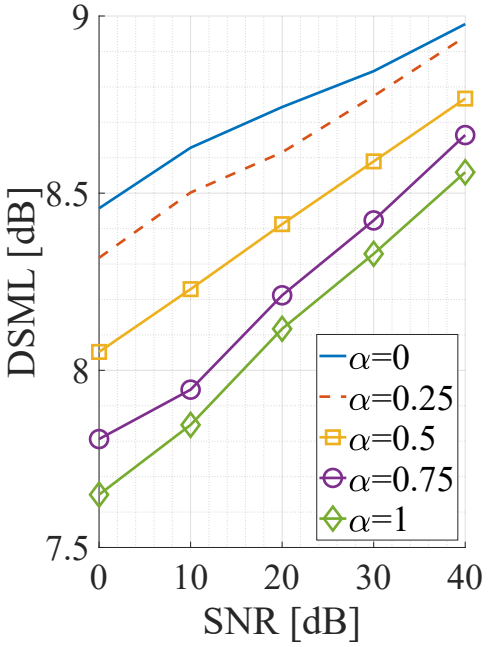

Figure 7.13: DSML for various values of α in different SNR levels.

SAR, which is applicable to single-talk segments where speech is less distorted by the RES system. The RESL is always lower than the ERLE, which is relevant to segments without desired speech where echo is more suppressed. These observations highlight the reliability of the DSML and RESL metrics.

Next, the relation between tuning α and the DSML-RESL tradeoff it creates is investigated. Figures 7.7–7.9 considers segments without and with echo-path change after convergence, and segments before convergence, for $\alpha \in \{0, 0.25, 0.5, 0.75, 1\}$. As α increases, speech is more distorted and the DSML decreases, while residual echo is more suppressed and the RESL increases. This tradeoff occurs across all scenarios and is empirically consistent for all α values. This tradeoff is also analyzed in various SER and SNR levels that occur in real-life setups. In this experiment, segments without echo-path change are considered and results are given in Figures 7.10–7.13. It can be observed that both the DSML and RESL are impaired when acoustic conditions deteriorate, as expected. Also, the relation between α and the metrics is retained, i.e., for all levels of echo and noise, increasing α degrades the DSML and enhances the RESL.

Finally, we offer a practical design scheme for possible dynamic user requirements. Assume an environment without echo-path change after convergence, which can be inferred by the user using the definitions in [ICB21b, PCBG15]. At first, the user requires an average RESL higher than 30 dB and DSML higher than 8.4 dB. According to Figure 7.10, $\alpha = 0.5$ is selected. Next, the user evaluates that SER = 0 dB and SNR = 20 dB, e.g., by respectively analyzing double-talk and near-end single-talk periods, and accordingly decides to suppress the maximal amount of echo that maintains DSML no lower than 8.3 dB. Then, according to Figure 7.12, the user shifts $\alpha = 0.5$ to $\alpha = 0.75$ during training, which decreases the average DSML to 8.3 dB and increases the average RESL to above 31 dB.

7.7 Conclusions

We introduced two objective metrics to separately assess the DSML and the RESL during double-talk. The performances of these metrics are evaluated using a deep

learning-based RES system with a tunable design parameter α, with 280 hours of real and synthetic recordings. We showed that the DSML and RESL correlate well with human perception compared to the popular SDR metric, which may suggest they are more suitable for speech quality evaluation. Also, we empirically learned the relation between tuning α and the resulting DSML-RESL tradeoff and offered a practical design scheme that benefits dynamic user preferences. Future work will analyze the DNSMOS as an appropriate evaluation for RES subjective quality in double-talk, and explore the DSML-RESL tradeoff to yield a practical design scheme for optimal speech quality.

Chapter 8

A User-centric Approach for Deep Residual-Echo Suppression in Double-talk

8.1 Introduction

Hands-free speech communication has become increasingly popular in recent years due to the growing trend of transitioning from face-to-face meetings to online meetings [SDT21], which are characterized by two conversation ends; far-end and near-end. In business calls, for instance, the far-end speaker is commonly a single participant that wears a set of headphones in a close-talk environment, while the near-end is an office conference room. In that setup, speech from the far-end is transmitted to the near-end, where it echoes via a nonlinear loudspeaker. In modern conferencing, loudspeakers are frequently not enclosed with, but are detached from the near-end microphone, which creates an acoustic coupling between the two [SCS+21]. Thus, in double-talk periods, the near-end microphone may capture reverberant echo, desired speech from participants in the near-end, and additional noises. This may cause echo to be transmitted back to the far-end and to severely impede the conversation intelligibility [BGM+01, SMH95].

Various linear AEC systems combat this issue [PSK+20, ICB22a, SDA22, ZGZ23,

YYC+21, IBMH22]. However, these methods often cannot eliminate echo presence in realistic setups due to nonideal hardware that induces nonlinearity between the echo and the far-end signal [ICB21b], the rapidly-varying nature of the echo path, and the complicated modeling of echo in double-talk. RES systems have achieved impressive results using deep learning to eliminate linear and nonlinear echo patterns that are still present after the linear AEC stage [ZWS+22, FF22, DDBW22, XYC22, LSK15, CSVH18, ZTW19, PP20, ZL20, ICB22c]. In double-talk, RES systems trade-off between residual-echo suppression and desired-speech distortion levels in their output [ICB21a]. To evaluate this trade-off, we have introduced two objective performance metrics for RES in double-talk [ICB21c]; the RESL and the DSML. In [ICB22b], we showed a strong correlation between these metrics and the recent AECMOS objective metric, which predicts subjective human ratings of speech quality of AEC systems with high accuracy in double-talk [SCS+21, PSS+22].

Existing studies on RES primarily focus on improving benchmark performance, rather than on supporting users inputs. For instance, the vast majority of RES systems neither offer a framework to trade-off between residual echo and speech-distortion levels at their output, nor do they report performance across various operating points that represent this trade-off. Rather, users employ existing RES systems based on an average benchmark-performance, which is frequently reported with metrics that do not distinguish residual-echo presence from desired-speech distortion [ICB21c], e.g., signal-to-distortion-ratio [VGF06] or perceptual evaluation of speech quality [RBHH01]. Even if an off-the-shelf model is rendered suitable by a user for a specific scenario, adjustments based on user preferences are not supported. Although the AECMOS is currently the most accurate objective assessment for speech quality by humans, no RES system provides a mechanism to maximize the AECMOS. These gaps limit the user experience and user flexibility in dynamic environments that often require personalized adjustments. In practice, a business presentation given in a near-end conference room may lead the far-end listener to incline towards low speech distortion. In contrast, residual-echo suppression may be more important during frequent abrupt echo-path changes that occur when transitioning from the presentation to a near-end multi-participant discussion.

We introduce the URES framework in double-talk. The URES is initiated with a

UOP that consists of two performance metrics values; the RESL and DSML [ICB21c] that the user wishes to experience from the RES prediction. The URES system then undergoes three stages. Firstly, we utilize an existing deep RES model that we introduced in [ICB21a]. This model embeds a design parameter that controls the trade-off between the RESL and DSML of the RES prediction. We consider 101 pre-trained instances from this model, each with a different design parameter value. Feeding the same input to all instances results in different RESL and DSML values in the prediction of every instance, which covers a wide range of UOPs. Second, each prediction is fed to a separate pre-trained deep model, which maps this prediction to its RESL and DSML estimates. This is essential since these metrics depend on the desired-speech signal that is unavailable in double-talk in practice. Third, the estimates from all instances are compared with the UOP. The ones that match it, up to a given tolerance threshold that specifies the allowed deviation from the UOP, are narrowed down to the single prediction with the maximal AECMOS, which is transmitted to the far-end. The proposed URES system has three unique advantages; the RESL and DSML of its output match or approach the UOP, real-time changes in the UOP are tracked, and the AECMOS of its output is maximized.

Experiments employ 60 hours of noisy real and synthetic data that include realistic acoustic scenarios with extremely high levels of echo and frequent echo-path changes. Average results can achieve an AECMOS of 4.4 out of 5 with RESL and DSML deviations of 1.95 dB and 2.1 dB from the UOP. Any user adjustment can be tracked in 38.4 ms. E.g., tightening the tolerance threshold can transition the above output to a lower AECMOS of 3.6 while the RESL and DSML deviations improve to 1.25 dB and 1.45 dB from the UOP, on average.

The remainder of this chapter is organized as follows. In Section 8.2, we formulate the problem. In Section 8.3, we describe the proposed solution. In Section 8.4, we lay out the experimental setup. In Section 8.5, we present the experimental results. Finally, in Section 8.6, we draw conclusions.

8.2 Problem Formulation

The proposed URES system is depicted in Fig. 8.1. The near-end microphone signal in time index n is expressed as:

$$m\left(n\right)=s\left(n\right)+w\left(n\right)+y\left(n\right), \tag{8.1}$$

where $s\left(n\right)$ holds the desired speech and $w\left(n\right)$ holds environmental and system noises. The reverberant echo $y\left(n\right)$ satisfies:

$$y\left(n\right)=\left(h*x^{\mathrm{NL}}\right)\left(n\right), \tag{8.2}$$

i.e., a convolution between $h\left(n\right)$ and $x^{\mathrm{NL}}\left(n\right)$, which respectively denote the near-end RIR from the loudspeaker to the microphone and a nonlinearly distorted far-end signal. We apply adaptive filtering for the linear AEC system that receives $m\left(n\right)$ as input and the far-end signal $x\left(n\right)$ as reference, and produces the echo-path estimate $\widehat{h}\left(n\right)$. The echo estimate $\widehat{y}\left(n\right)$ and the adaptation error $e\left(n\right)$ are:

$$\widehat{y}\left(n\right)=\left(\widehat{h}*x\right)\left(n\right), \tag{8.3}$$

$$e\left(n\right)=m\left(n\right)-\widehat{y}\left(n\right)\overset{(8.1)}{=}\left(y\left(n\right)-\widehat{y}\left(n\right)\right)+s\left(n\right)+w\left(n\right). \tag{8.4}$$

The signals $x\left(n\right)$, $\widehat{y}\left(n\right)$, $e\left(n\right)$, and $m\left(n\right)$ are inserted into the URES system that produces the desired-speech estimate $\widehat{s}\left(n\right)$ and then communicates it to the far-end. The goal is that $\widehat{s}\left(n\right)$ confines to a UOP and achieves the maximal AECMOS value.

8.3 A User-centric Approach for Deep RES

This process is comprised of three main stages. The first stage is described in subsection 8.3.1, where the user chooses a UOP that includes two values; the RESL and the DSML of the RES prediction. In the second stage, detailed in subsections 8.3.2 and 8.3.3, deep models we developed generate several RES predictions with RESL and DSML values that match the UOP, up to a given tolerance threshold. The third stage in subsection

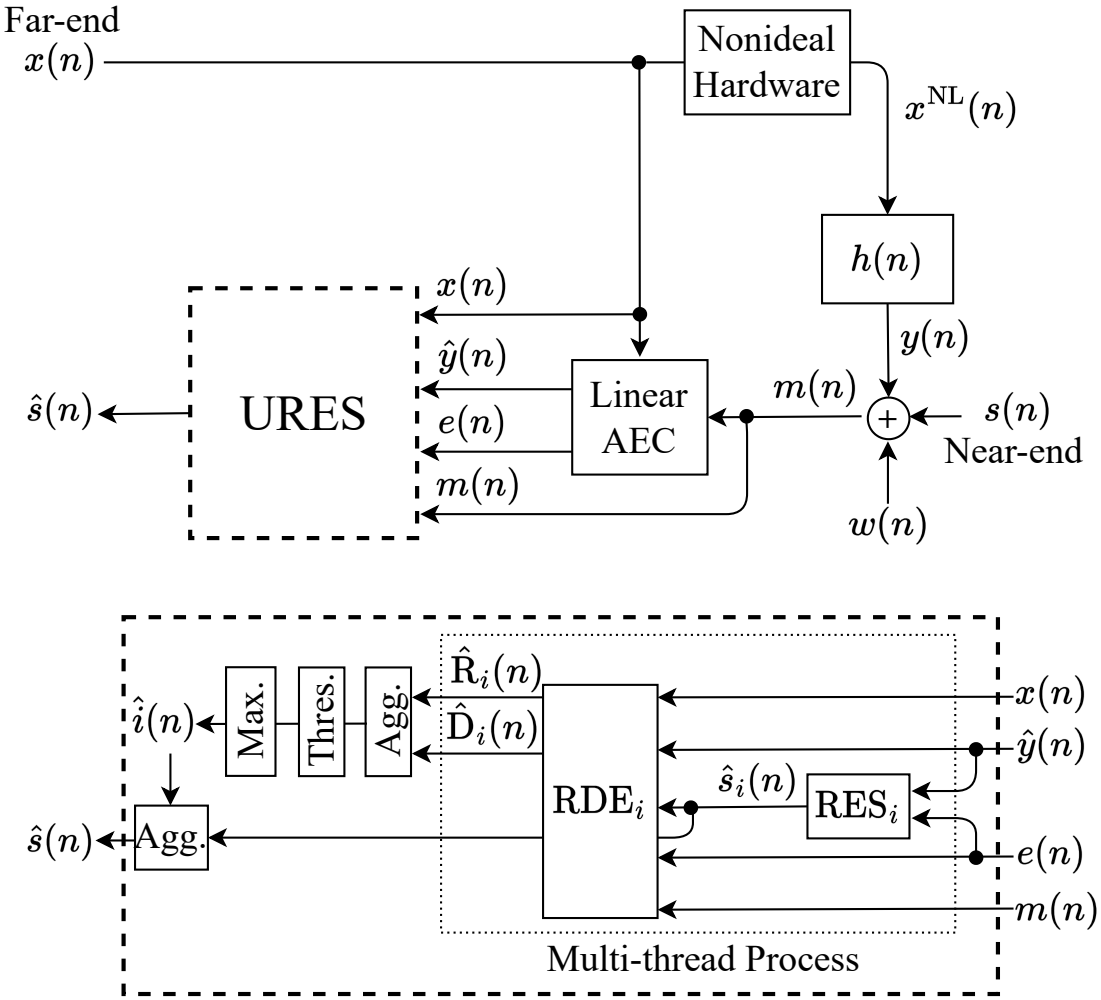

Figure 8.1: The three stages of the proposed URES framework. (1) For $0 \leq i \leq 100$, the i^{th} model instance RES_i produces a prediction $\widehat{s}_i\,(n)$. (2) $\widehat{s}_i\,(n)$ is inserted to the corresponding i^{th} model instance RDE_i, which estimates the RESL and DSML of $\widehat{s}_i\,(n)$, respectively notated $\widehat{\mathrm{R}}_i\,(n)$ and $\widehat{\mathrm{D}}_i\,(n)$. (3) These estimates are aggregated from all instances and undergo threshold filtering by their proximity to the UOP, followed by an AECMOS maximization. The prediction with the chosen index $\widehat{i}$, namely $\widehat{s}_{\widehat{i}}(n)$, is communicated to the far-end. Notice the RES and RDE models multi-thread for inference for all i.

8.3.4 depicts how the prediction with the highest AECMOS is chosen, before being communicated to the far-end.

8.3.1 Providing a user operating-point for the URES framework

The UOP consists of a pair of RESL and DSML values. In [ICB21c], we introduced the RESL and DSML metrics to separately assess residual echo and speech-distortion levels of RES systems in double-talk. We also provided empirical results of average

RESL and DSML values in which the RES system operates, which may guide a UOP selection. Let the UOP be (R, D), where R is the RESL and D is the DSML. This study supports $15 \leq \mathrm{R} \leq 30$ and $7.5 \leq \mathrm{D} \leq 15$, in dB.

8.3.2 RES with a tunable design parameter

Inspired by our work [ICB21a], we utilize a deep RES system that receives the outcomes of the linear AEC stage as two input channels, i.e., the echo estimate and the adaptation error. This system aims to remove residual-echo components and preserve the desired speech in the STFT domain [Zhi19]. The architecture is based on the UNet [RFB15] neural network. During training, the design parameter $\alpha \geq 0$ governs the trade-off between residual echo and speech distortion levels at the output of the RES system by regulating the following objective function:

$$J(\alpha) = \left\| \widehat{S}(f) - S(f) \right\|_2^2 + \alpha \cdot \left\| \widehat{S}(f) \right\|_2^2 + \sigma^2_{\widehat{S}(f)} \cdot \mathbb{I}_{\alpha>0} \ , \tag{8.5}$$

where $\widehat{S}(f)$ and $S(f)$ are the STFT amplitudes of $\widehat{s}(n)$ and $s(n)$, respectively, $\|\widehat{S}(f)\|_2$ is the ℓ_2-norm of $\widehat{S}(f)$, $\sigma^2_{\widehat{S}(f)}$ is the variance of $\widehat{S}(f)$, and $\mathbb{I}_{\alpha>0}$ equals 1 when $\alpha > 0$ and 0 otherwise. According to (8.5), when α increases then the training process inclines towards minimizing the norm of the prediction. This creates more residual-echo suppression, but constrains the speech component in the output to a higher distortion rate. In contrast, as α lowers and reaches $\alpha = 0$, more focus is put on minimizing the distortion between the system prediction and the desired speech for the price of high residual-echo presence. In [ICB21c], we have shown how the average RESL values rise and how the average DSML values lower when α increases, and vice versa. Since for both the RESL the DSML, higher values mean better performance, shifting α can change the operating point of the RES system and match it with the UOP. We exploit this property and separately pre-train 101 identical instances of the RES system, each with a different α value ranging from $\alpha = 0$ to $\alpha = 1$ with increments of 0.01. This large number of α values separated by a thin resolution was empirically shown to cover a wide range of RESL and DSML pairs in the support of the UOP. It was also shown that $\alpha > 1$ causes undesired nullification of sub-bands in the RES prediction. The

index $0 \leq i \leq 100$ is used to notate each pre-trained RES model instance and each of its corresponding predictions, i.e., RES_i and $\widehat{s}_i(n)$, respectively. For all i values, the design parameter value used to train RES_i is calculated by $\alpha_i = i/100$.

8.3.3 Estimation of the RESL and DSML metrics

Each prediction from the 101 RES system instances from subsection 8.3.2 separately undergoes RESL and DSML estimation. These estimates are then compared with the UOP. Formally, the RESL and DSML metrics [ICB21c] depend on the time-varying gain of the RES system in double-talk, given by:

$$g(n) = \left.\frac{\widehat{s}(n)}{e(n)}\right|_{\text{Double-talk}}, \tag{8.6}$$

where $e(n) \neq 0$. By applying the gain to the desired-speech only and calculating the following ratio, the DSML is derived:

$$\text{DSML} = 10\log_{10}\left.\frac{\|\tilde{s}(n)\|_2^2}{\|\tilde{s}(n) - g(n)\,s(n)\|_2^2}\right|_{\text{Double-talk}}. \tag{8.7}$$

The RESL is manufactured by considering $r(n) = e(n) - s(n)$ as the noisy residual-echo estimate and calculating the ratio:

$$\text{RESL} = 10\log_{10}\left.\frac{\|r(n)\|_2^2}{\|g(n)\,r(n)\|_2^2}\right|_{\text{Double-talk}}. \tag{8.8}$$

The inherent bias that deep models may apply is being compensated by defining $\tilde{s}(n) = \widehat{g}(n)\,s(n)$, where:

$$\widehat{g}(n) = \frac{\langle g(n)\,s(n), s(n)\rangle}{\|s(n)\|_2^2}. \tag{8.9}$$

According to (8.7)–(8.8), the RESL and DSML metrics cannot be calculated in practice since they require knowledge of the desired speech $s(n)$. Namely, during inference the prediction of the RES system cannot be translated into its RESL and DSML values. Thus, we developed a deep model notated an RDE that estimates the RESL and DSML by implicitly evaluating $s(n)$. To construct the inputs of the RDE, we first recognize

the following relation using eqs. (8.1), (8.2):

$$s\left(n\right)=m\left(n\right)-\left(h*x^{\mathrm{NL}}\right)\left(n\right)-w\left(n\right). \tag{8.10}$$

By considering $m\left(n\right)$ as an input to the RDE and by ignoring the noise $w\left(n\right)$, it is left to estimate $h\left(n\right)$ and $x^{\mathrm{NL}}\left(n\right)$. Based on the linear relation in (8.3), inserting both $\widehat{y}\left(n\right)$ and $x\left(n\right)$ to the RDE should yield $\widehat{h}\left(n\right)$, which estimates $h\left(n\right)$. Notice that $\widehat{h}\left(n\right)$ is practically available from the linear AEC stage, but its non-speech structure makes it more effective to feed the RDE with speech signals and derive implicit relations between them, which is empirically supported in our internal experiments. By (8.1) and (8.2), we estimate $x^{\mathrm{NL}}\left(n\right)$ by using $x\left(n\right)$ and $m\left(n\right)$. The former constitutes a linear part of $x^{\mathrm{NL}}\left(n\right)$, and the latter is a mix of signals that includes $x^{\mathrm{NL}}\left(n\right)$. The RDE is also fed with $e\left(n\right)$, which is employed in the RESL and DSML calculations. As a final input, we insert $\widehat{s}\left(n\right)$ to the model since it is both an integral component of the RESL and DSML calculations and because it is constructed to approximate $s\left(n\right)$. Similarly to subsection 8.3.2, we utilize 101 identical RDE model instances. RDE_i, which denotes the i^{th} RDE instance, receives 5 channels in the time domain, i.e., $x\left(n\right)$, $\widehat{y}\left(n\right)$, $e\left(n\right)$, $m\left(n\right)$, and $\widehat{s}_i\left(n\right)$. Let the predicted RESL and DSML values of RES_i be respectively denoted as $\widehat{\mathrm{R}}_i\left(n\right)$ and $\widehat{\mathrm{D}}_i\left(n\right)$. During training, the ℓ_2 distance is minimized between the pair of estimates $\widehat{\mathrm{R}}_i\left(n\right)$ and $\widehat{\mathrm{D}}_i\left(n\right)$, and the pair of ground truth calculations of the RESL and DSML using (8.6)–(8.9).

8.3.4 Maximizing the AECMOS

In this stage, we describe how the final prediction of the URES framework is determined before being communicated to the far-end. First, for all i values, $\widehat{\mathrm{R}}_i\left(n\right)$ and $\widehat{\mathrm{D}}_i\left(n\right)$ are aggregated into one batch that contains 101 pairs of values. Second, the UOP from subsection 8.3.1 is being compared against each pair in this batch. Let us respectively define the tolerance threshold values TH_{R} and TH_{D} as the maximal allowed deviation of $\widehat{\mathrm{R}}_i\left(n\right)$ and $\widehat{\mathrm{D}}_i\left(n\right)$ from the UOP coordinates R and D. Consider a subset of the

batch that contains the indices that follow these two conditions, evaluated for all i:

$$\left|\widehat{\mathrm{R}}_i\left(n\right)-\mathrm{R}\right|<\mathrm{TH}_{\mathrm{R}}, \tag{8.11}$$

$$\left|\widehat{\mathrm{D}}_i\left(n\right)-\mathrm{D}\right|<\mathrm{TH}_{\mathrm{D}}, \tag{8.12}$$

where $\mathrm{TH}_{\mathrm{R}}\geq 0$ and $\mathrm{TH}_{\mathrm{D}}\geq 0$ and are measured in dB. We notate the number of indices in this subset as $P\left(n\right)$, where $0\leq P\left(n\right)\leq 101$. Third, let $\widehat{i}$ denote the single index from the subset the corresponds to the prediction with the highest AECMOS. We denote $\Delta_{\mathrm{R}}\left(n\right)$ and $\Delta_{\mathrm{D}}\left(n\right)$ as the deviations of the chosen output prediction from the UOP, as follows:

$$\Delta_{\mathrm{R}}\left(n\right)=\left|\widehat{\mathrm{R}}_{\widehat{i}}(n)-\mathrm{R}\right|, \tag{8.13}$$

$$\Delta_{\mathrm{D}}\left(n\right)=\left|\widehat{\mathrm{D}}_{\widehat{i}}(n)-\mathrm{D}\right|, \tag{8.14}$$

where by definition $\Delta_{\mathrm{R}}\left(n\right)<\mathrm{TH}_{\mathrm{R}}$ and $\Delta_{\mathrm{D}}\left(n\right)<\mathrm{TH}_{\mathrm{D}}$. Finally, all original predictions $\widehat{s}_i\left(n\right)$ for all i are aggregated into one batch, and $\widehat{s}_{\widehat{i}}(n)$ is communicated to the far-end.

8.4 Experimental Setup

8.4.1 Database Acquisition

We utilize 50 hours from the AEC-challenge database and 10 hours of independent recordings performed in our lab. Both corpora contain only double-talk periods, i.e., where the far-end speech and near-end speech overlap. The AEC-challenge corpus was sampled at 16 KHz and is detailed in [CSP+22]. It includes acoustic scenarios when no echo-path change occurs and when echo-path change occurs regularly. No echo-path change describes scenarios when neither the near-end speakers nor near-end devices move, while echo-path change describes scenarios when at least one of the above does move regularly during the recording. We extract from this database 10 hours of synthetic data and 40 hours of real recordings, where the latter were captured using roughly $1,000$ hands-free devices in various acoustic environments. This data considers a wide range of noise and echo levels, having SER distributed in $[-10,10]$ dB and SNR

distributed in $[0, 40]$ dB. The independent recordings were sampled at 16 KHz and employ clips from the TIMIT [GLF+93a] and Librispeech [PCPK15] databases. This data only includes acoustic segments with no echo-path changes. A mouth simulator played the near-end speech and a loudspeaker modeled the effect of the nonlinear echo inside the near-end, where both devices were located in various positions in the room during the experiment. Both the speech and echo were captured by a microphone in the near-end. This database was collected to model especially challenging real-life acoustic scenarios that exhibit high echo levels. The SER levels were distributed in $[-20, -10]$ dB and SNR levels were roughly distributed in $[27, 37]$ dB. Formally, SER=$10\log_{10}\left(\|s\left(n\right)\|_2^2/\|y\left(n\right)\|_2^2\right)$ in dB and SNR=$10\log_{10}\left(\|s\left(n\right)\|_2^2/\|w\left(n\right)\|_2^2\right)$ in dB.

8.4.2 Preprocessing, Training, and Testing

The training set is comprised of 45 hours from the AEC-challenge database; 35 hours were randomly split from the 40 hours batch of real recordings, and 10 hours of synthetic data were included. The training set also contains 5 hours from the real independent recordings. The test set is comprised of only real recordings; the remaining 5 hours from the AEC-challenge and the remaining 5 hours from the independent recordings. The training and test sets are balanced to avoid bias by following guidelines from the preprocessing stage in [ICB21a]. Specifically, they contain equal representation for male and female participants, the far-end and near-end speakers are different, no speaker participates in both the training and test sets, and every speaker has been assigned as the far-end and near-end speaker. The linear AEC stage that precedes the URES system is a SNLMS adaptive filter [ICB22a, FD93] that operates in the time domain with a filter length of 150 ms. The training and test sets are each divided into 10 s segments and internally shuffled. This leads to abrupt echo-path changes that create frequent re-convergence of the linear AEC filter, as commonly occurs in real-life [DPBC20, FBD+22]. During training, each time domain signal is converted to its STFT amplitude that is normalized before inserted to the RES model. The output of this RES model then undergoes de-normalization and inverse STFT [Zhi19] using the overlap-add method [Cro80] by employing the phase from the adaptation error of

the linear AEC system. Normalization is carried by subtracting from the training set its minimal value and dividing it by its dynamic range. De-normalization is the inverse process. During inference, normalization and de-normalization are applied using the statistics from the training set [HQZ+23]. The RES and RDE models share the training samples of the echo estimate and adaptation error. The predictions of the RES models from the training stage are utilized to train the RDE models. The algorithmic and buffering latency of the URES framework is 38.4 ms using multi-threading [BNH18] of the RES and RDE models, confining with hands-free speech communication standards of maximal latency of 40 ms [ETS16].

8.4.3 Performance Measures

We use the AECMOS version number 4 from the API of Microsoft and calculate it using the output of the RES system and the adaptation error of the linear AEC stage. The AECMOS is unit-less and ranges in a scale of 1–5, where 5 is the best score [PSS+22]. In addition, $\Delta_{\mathrm{R}}(n)$ and $\Delta_{\mathrm{D}}(n)$ are used for evaluation and are calculated using eqs. (8.13) and (8.14), respectively. The RESL and DSML metrics are derived by considering the average value of a sliding analysis window in the time domain with 20 ms duration and with a step size of 10 ms. Lastly, results include the value of $P(n)$ as defined in subsection 8.3.4.

8.5 Experimental Results

During the inference stage, every utterance from the test set is inferred with a random UOP pair where R is uniformly drawn from $[15, 30]$ dB and D is uniformly drawn from $[7.5, 15]$ dB. Unless stated otherwise, results are reported using mean and standard deviation (std) values of performance metrics across the entire test set. In the tables, the format is mean±std, and in the figures the format includes mean values either with or without std error bars. This section addresses global results and neglects time indices from notation.

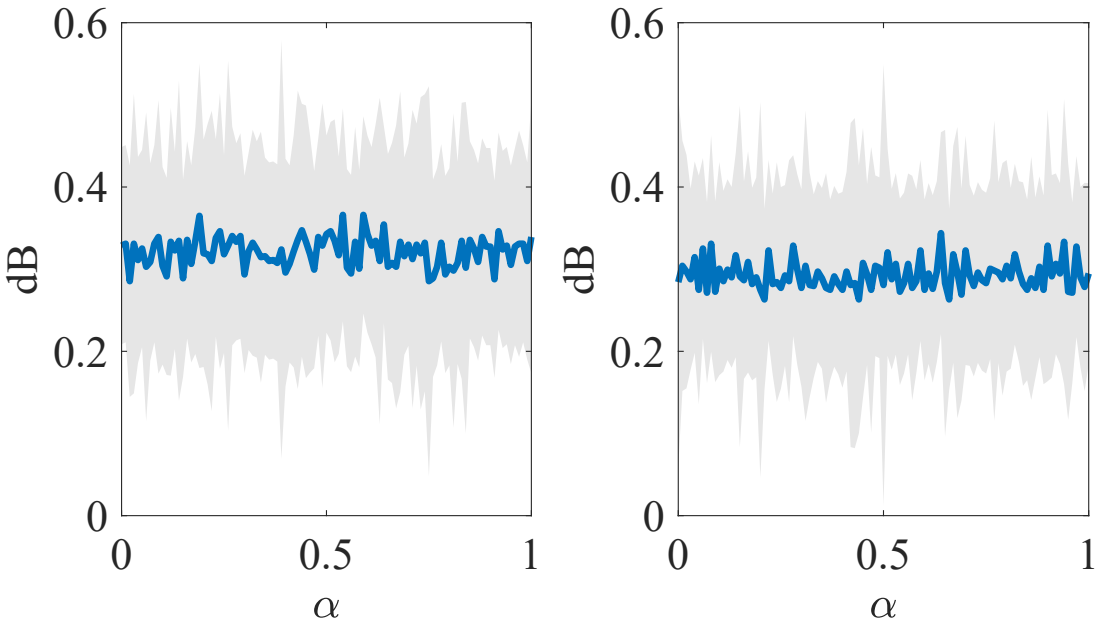

Figure 8.2: The ℓ_1 error of the RESL (left) and DSML (right) estimates for each of the 101 RDE model instances versus their α values.

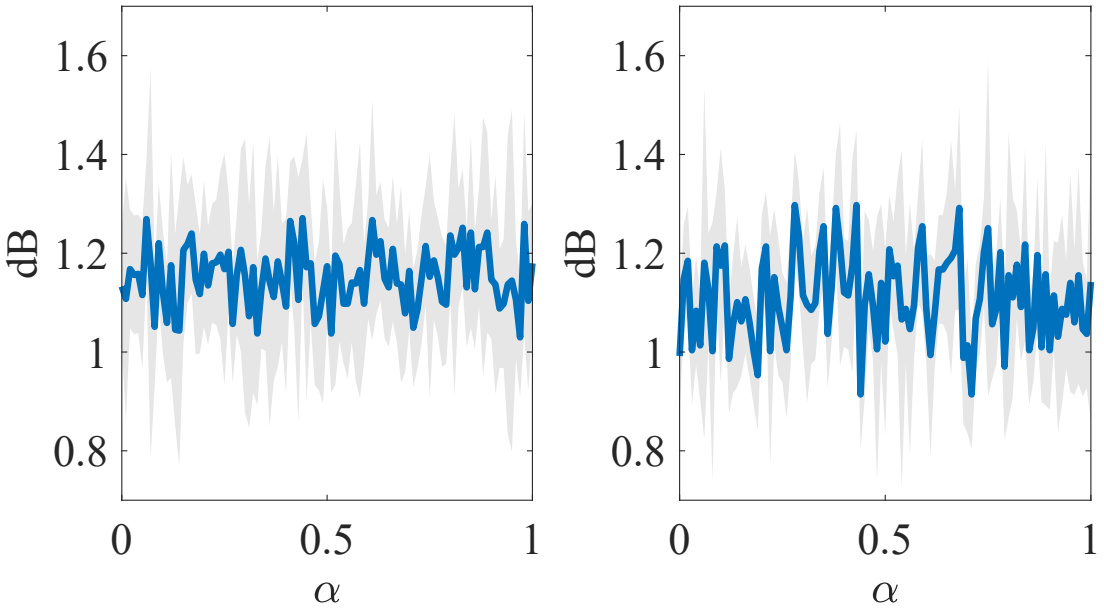

Figure 8.3: The ℓ_1 error of the RESL (left) and DSML (right) estimates of a single RDE model instance versus the α values associated with the preceding RES model instances.

8.5.1 Validating the performance of the RDE models

This experiment examines the estimation reliability of the RESL and DSML values by the 101 RDE model instances. Using 10-fold cross-validation [RPL09], 80% of the training set is utilized for training, and the remaining 20% is used for validation, where

the same bias-free principles between the training and test sets detailed in subsection 8.4.2 are applied between the crossed training and validation sets in every fold. For every fold and for every i, where $0 \leq i \leq 100$, the crossed training set is used to train the model instances RES_i and RDE_i by following the process in subsection 8.4.2. Then, RDE_i infers the crossed validation set and produces the corresponding RESL and DSML estimates. These estimates are being compared against the ground-truth RESL and DSML of the validation set. Figs. 8.2–8.3 show the RESL and the DSML estimation performance of all 101 RDE model instances. For both the RESL and the DSML, the reported values are the mean and std of the ℓ_1 distance between the estimates and their ground truth across all folds. Reminding that $\alpha_i = i/100$, it is shown that the RESL estimate experiences maximal mean error of 0.36 dB for $\alpha_{54} = 0.54$, and one std can bring the error up to 0.57 dB for $\alpha_{39} = 0.39$. The DSML estimate has a maximal mean error of 0.34 dB for $\alpha_{64} = 0.64$, and one std can bring the error up to 0.5 dB for $\alpha_{54} = 0.54$. Considering this study supports RESL in $[15, 30]$ dB and DSML in $[7.5, 15]$ dB, the maximal mean error values can also be viewed in a relative scale by normalizing them by their corresponding ranges; namely $100 \cdot 0.36/15 = 2.4\%$ and $100 \cdot 0.34/7.5 = 4.5\%$. Based on these results, a subjective view suggests that using 101 RDE model instances produces a consistently reliable average estimation of the RESL and DSML in various acoustic setups. A following experiment examines the less computationally-heavy possibility of employing a single RDE model for all α values. Similarly to the previous experiment, a 10-fold cross validation is used to train every RES model instance with its corresponding α value. This time, however, all the outputs of the RES model instances are aggregated and a single RDE model is used for training and validation for every fold. To ensure bias-free results, the distribution of segments associated with every α value is uniform in both the crossed training and validation sets of every fold. According to Figs. 8.2–8.3, it is shown that the RESL estimate experiences maximal mean error of 1.27 dB for $\alpha_{44} = 0.44$, and one std can bring the error up to 1.57 dB for $\alpha_7 = 0.07$. The DSML estimate has a maximal mean error of 1.29 dB for $\alpha_{68} = 0.68$, and one std can bring the error up to 1.59 dB for $\alpha_{75} = 0.75$. Again, the maximal mean error values can also be viewed in a relative scale; namely $100 \cdot 1.27/15 = 8.4\%$ and $100 \cdot 1.29/7.5 = 17.2\%$. Based on these results,

	$\text{TH}_\text{R} = 1$ [dB]		
	Δ_R [dB]	Δ_D [dB]	AECMOS
$\text{TH}_\text{D} = 1$ [dB]	0.4 ± 0.3	0.55 ± 0.25	3.1 ± 0.3
$\text{TH}_\text{D} = 2$ [dB]	0.55 ± 0.25	1.3 ± 0.2	3.45 ± 0.4
$\text{TH}_\text{D} = 3$ [dB]	0.65 ± 0.25	1.9 ± 0.2	3.7 ± 0.5

Table 8.1: The effect of tolerance threshold values on the URES framework performance for segments with no echo-path change for $\text{TH}_\text{R} = 1$ [dB].

	$\text{TH}_\text{R} = 2$ [dB]		
	Δ_R [dB]	Δ_D [dB]	AECMOS
$\text{TH}_\text{D} = 1$ [dB]	1.15 ± 0.45	0.6 ± 0.15	3.35 ± 0.3
$\text{TH}_\text{D} = 2$ [dB]	1.25 ± 0.45	1.45 ± 0.3	3.6 ± 0.4
$\text{TH}_\text{D} = 3$ [dB]	1.3 ± 0.4	2.05 ± 0.3	4.2 ± 0.5

Table 8.2: The effect of tolerance threshold values on the URES framework performance for segments with no echo-path change for $\text{TH}_\text{R} = 2$ [dB].

	$\text{TH}_\text{R} = 3$ [dB]		
	Δ_R [dB]	Δ_D [dB]	AECMOS
$\text{TH}_\text{D} = 1$ [dB]	1.75 ± 0.65	0.7 ± 0.15	3.5 ± 0.5
$\text{TH}_\text{D} = 2$ [dB]	1.85 ± 0.6	1.55 ± 0.25	4.0 ± 0.3
$\text{TH}_\text{D} = 3$ [dB]	1.95 ± 0.65	2.1 ± 0.3	4.4 ± 0.2

Table 8.3: The effect of tolerance threshold values on the URES framework performance for segments with no echo-path change for $\text{TH}_\text{R} = 3$ [dB].

a subjective view suggests that a single RDE model is not reliable in estimating the RESL and DSML values, on average. To recap, utilizing a single RDE model may cause an accumulated uncertainty and bias of results, while 101 RDE model instances provide confident results. This renders the computational load of the latter worthy.

8.5.2 The effect of the tolerance threshold values on performance

The performance of the URES framework is examined with respect to the tolerance threshold parameters TH_R and TH_D. We consider $(\text{TH}_\text{R}, \text{TH}_\text{D})$ pairs that confine to $\text{TH}_\text{R} \in \{1, 2, 3\}$ in dB and $\text{TH}_\text{D} \in \{1, 2, 3\}$ in dB, which yields 9 possible pairs combinations. These sets values are representative of the URES system behavior but do not significantly deviate from the UOP. For each $(\text{TH}_\text{R}, \text{TH}_\text{D})$ pair, the mean and std of Δ_R, Δ_D, and the AECMOS are reported. Tables 8.1–8.3 consider test set utterances only with no echo-path changes. A clear trade-off is shown between the tolerance

	$\text{TH}_\text{R} = 1$ [dB]		
	Δ_R [dB]	Δ_D [dB]	AECMOS
$\text{TH}_\text{D} = 1$ [dB]	0.5 ± 0.25	0.65 ± 0.2	2.95 ± 0.3
$\text{TH}_\text{D} = 2$ [dB]	0.65 ± 0.35	1.45 ± 0.3	3.2 ± 0.45
$\text{TH}_\text{D} = 3$ [dB]	0.7 ± 0.1	2.05 ± 0.45	3.5 ± 0.6

Table 8.4: The effect of tolerance threshold values on the URES framework performance for segments with echo-path change for $\text{TH}_\text{R} = 1$ [dB].

	$\text{TH}_\text{R} = 2$ [dB]		
	Δ_R [dB]	Δ_D [dB]	AECMOS
$\text{TH}_\text{D} = 1$ [dB]	1.25 ± 0.4	0.65 ± 0.2	3.05 ± 0.4
$\text{TH}_\text{D} = 2$ [dB]	1.3 ± 0.45	1.55 ± 0.3	3.3 ± 0.5
$\text{TH}_\text{D} = 3$ [dB]	1.45 ± 0.45	2.2 ± 0.3	3.8 ± 0.5

Table 8.5: The effect of tolerance threshold values on the URES framework performance for segments with echo-path change for $\text{TH}_\text{R} = 2$ [dB].

	$\text{TH}_\text{R} = 3$ [dB]		
	Δ_R [dB]	Δ_D [dB]	AECMOS
$\text{TH}_\text{D} = 1$ [dB]	1.85 ± 0.65	0.75 ± 0.2	3.35 ± 0.5
$\text{TH}_\text{D} = 2$ [dB]	1.9 ± 0.6	1.65 ± 0.2	3.7 ± 0.3
$\text{TH}_\text{D} = 3$ [dB]	2.05 ± 0.6	2.2 ± 0.35	3.9 ± 0.3

Table 8.6: The effect of tolerance threshold values on the URES framework performance for segments with echo-path change for $\text{TH}_\text{R} = 3$ [dB].

threshold values and the yielded AECMOS. Limiting the permitted deviation of both the RESL and DSML estimates from the UOP to 1 dB leads to a mean AECMOS value of 3.1 dB out of 5, which is considered a subjectively mediocre human evaluation. Allowing a larger deviation of $(\text{TH}_\text{R}, \text{TH}_\text{D}) = (3, 3)$ in dB, leads to an AECMOS average of 4.4, which is subjectively considered excellent [PSS+22]. The trade-off most probably occurs since increasing the TH_R and TH_D creates a larger set of possible predictions after the threshold stage, which increases the average maximal AECMOS value of these predictions. Tables 8.4–8.6 addresses segment only with echo-path changes. The trade-off described above remains, but with a consistent reduction in the average AECMOS values across all $(\text{TH}_\text{R}, \text{TH}_\text{D})$ pairs. This is associated with the linear AEC stage struggle with tracking and modeling linear echo in changing echo-path scenarios, which affects the average performance of the successive RES system [ICB21a]. Thus, the output of the URES pipeline that relies on the predictions of the RES system in-

stances, degrades in its overall subjective evaluation of speech quality that is quantified by the AECMOS. Interestingly, results are consistently not symmetric in both tables. E.g., $(\mathrm{TH_R}, \mathrm{TH_D}) = (2, 3)$ in dB and $(\mathrm{TH_R}, \mathrm{TH_D}) = (3, 2)$ in dB have respective average AECMOS values of 4.2 and 4 in Tables Tables 8.1–8.3. One explanation relies on the human auditory system, which is more sensitive to speech distortion than to residual echo [Vir99]. Having a larger range for the DSML to deviate from the UOP, i.e., controlling more of the speech distortion rate, enhances the average AECMOS more than symmetrically applying this logic to the RESL. These tables also give an intuition of how the objective Δ_R and Δ_D empirically relate to the subjective human rating prediction in the AECMOS. Therefore, relaying on Tables 8.1–8.3 and Tables 8.4–8.6 may allow an educated choice by the user regarding $\mathrm{TH_R}$ and $\mathrm{TH_D}$. It is highlighted that while an estimation error as discussed in subsection 8.5.1 of 1 dB, for instance, may cause uncertainty and bias in the results, the human perception of 1 dB deviation from the UOP tend to be imperceptible [Yos01]. Overall, the URES framework can enable a deviation from the UOP that is subjectively low-perceived [Yos01] along with a subjectively excellent AECMOS, on average, in various acoustic scenarios.

8.5.3 The effect of the tolerance threshold values on P

This experiment includes scenarios with and without echo-path changes and reports the average P value for every $(\mathrm{TH_R}, \mathrm{TH_D})$ pair that confines to $\mathrm{TH_R} \in \{1, 2, 3, 4, 5\}$ in dB and $\mathrm{TH_D} \in \{1, 2, 3, 4, 5\}$ in dB, which totals to 25 pairs combinations. By observing Figs. 8.4–8.5, P increases as the tolerance threshold values increase, and vice versa. This is expected since the construction of the URES framework ensures that, on average, the higher $\mathrm{TH_R}$ and $\mathrm{TH_D}$ become, the larger amount of RES predictions are available to undergo AECMOS maximization after the threshold stage, namely P increases, and vice versa. An important case is where $(\mathrm{TH_R}, \mathrm{TH_D}) = (1, 1)$ in dB, which averages approximately $P = 2$. This indicates that these tolerance threshold values are the lowest that are valid for the URES framework. A deeper dive reveals that $P = 0$ did not occur for this scenario and $P = 1$ was reported 17% of the time. On the other hand, $(\mathrm{TH_R}, \mathrm{TH_D}) = (5, 5)$, in dB, achieve an average of $P > 60$. Another observation is the proximity between the results with and without echo-path changes. Namely, even

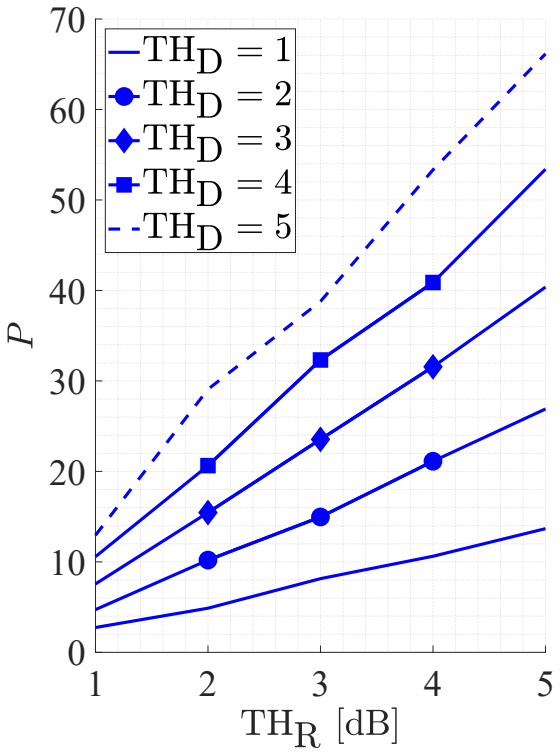

Figure 8.4: Average P values for various $(\mathrm{TH_R}, \mathrm{TH_D})$ pairs for scenarios with no echo-path change. The units of $\mathrm{TH_D}$ values in the legend are dB.

though in Tables 8.1–8.3 and Tables 8.4–8.6 the presence of echo-path changes clearly degraded the average AECMOS, it does not narrow the number of possible predictions that arrive at the AECMOS maximization stage. Conclusively, the URES framework supports even very narrow margins of 1 dB from the UOP. However, lightly relaxing this constraint enlarges P significantly, which increases the AECMOS, on average, as supported in Tables 8.1–8.3 and Tables 8.4–8.6 and in subsection 8.5.2.

8.5.4 The effect of echo and noise levels on performance

We recognize that the dynamic environment of hands-free speech communication exhibits various levels of echo and noise. Considering only segments with no echo-path changes and focusing on a tolerance threshold pair of $(\mathrm{TH_R}, \mathrm{TH_D}) = (2, 2)$ in dB, we report the average performance of the URES framework for SER levels from the set $\{-20, -10, 0, 10\}$ dB and for SNR levels from the set $\{0, 10, 20, 30, 40\}$ dB. It can be shown in Figs. 8.6–8.7 that in severe acoustic setups of -20 dB SER or of 0 dB SNR, the URES framework achieves average AECMOS values close to 3. In contrast, very

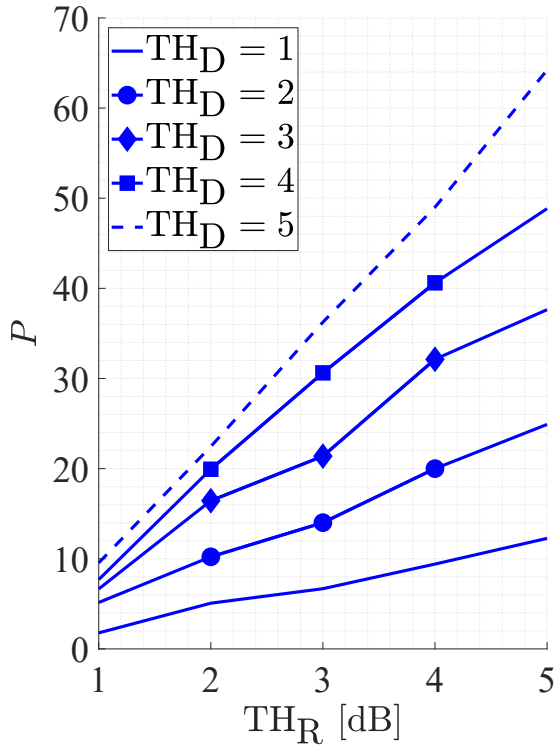

Figure 8.5: Average P values for various $(\mathrm{TH_R}, \mathrm{TH_D})$ pairs for scenarios with echo-path change. The units of $\mathrm{TH_D}$ values in the legend are dB.

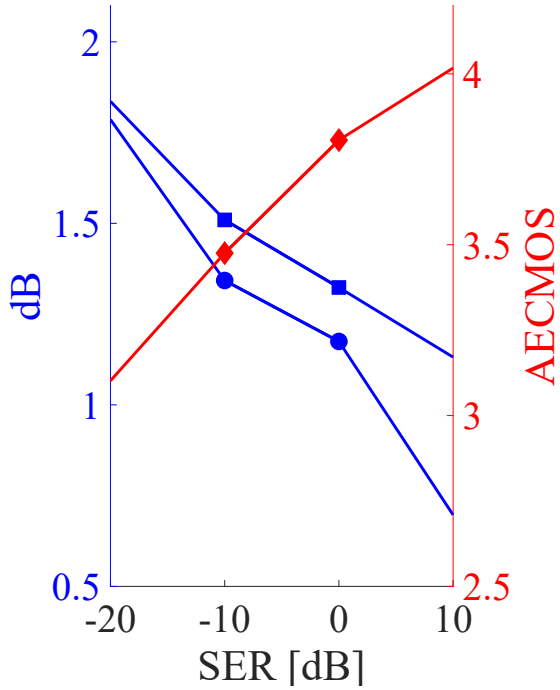

Figure 8.6: Average values of the AECMOS (diamonds), Δ_R in dB (circles) and Δ_D in dB (squares) for various levels of SER values with no echo-path change scenarios and $(\mathrm{TH_R}, \mathrm{TH_D}) = (2, 2)$ in dB.

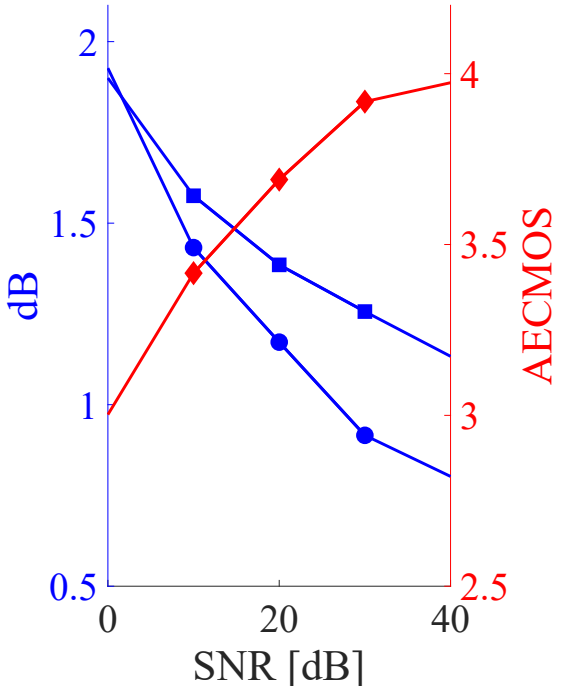

Figure 8.7: Average values of the AECMOS (diamonds), Δ_{R} in dB (circles) and Δ_{D} in dB (squares) for various levels of SNR values with no echo-path change scenarios and $(\mathrm{TH}_{\mathrm{R}}, \mathrm{TH}_{\mathrm{D}}) = (2, 2)$ in dB.

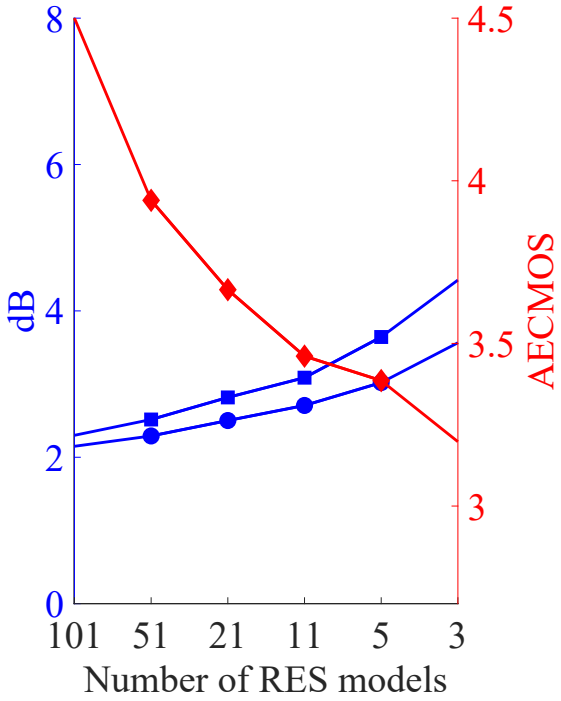

Figure 8.8: Average values of the AECMOS (diamonds), Δ_{R} in dB (circles) and Δ_{D} in dB (squares) versus number of trained RES model instances in scenarios without echo-path changes, considering $(\mathrm{TH}_{\mathrm{R}}, \mathrm{TH}_{\mathrm{D}}) = (5, 5)$ in dB.

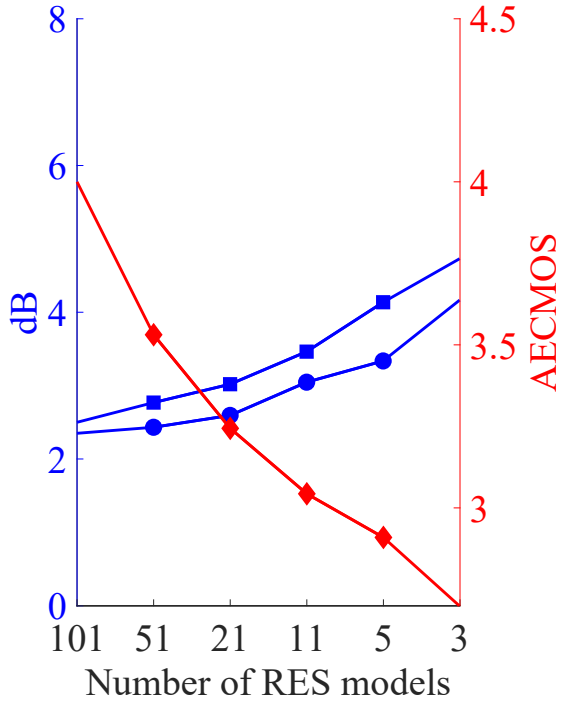

Figure 8.9: Average values of the AECMOS (diamonds), Δ_{R} in dB (circles) and Δ_{D} in dB (squares) versus number of trained RES model instances in scenarios with echo-path changes, considering $(\mathrm{TH_R}, \mathrm{TH_D}) = (5, 5)$ in dB.

friendly acoustics of 20 dB SER or 40 dB SNR allow an average AECMOS that approaches 4 or even exceeds it. It can be inferred that in degraded acoustic conditions, both the lowest average AECMOS and the largest average deviations from the UOP occur. One assumption is that in conditions of high echo and noise levels, subjective quality rating is maximized when the RESL and DSML are taken to their allowed extreme to suppress most echo and distort least speech possible. Another observation is that the Δ_{D} is almost consistently higher on average than Δ_{R} across all SER and SNR levels, which supports the claim made earlier of how the human auditory system favours less speech distortion over less echo suppression. In summary, challenging but practical conditions, e.g., SER = 0 dB and SNR = 20 dB, are handled well by the URES system, which allows a broad support of this framework in various acoustic environments.

8.5.5 The effect of the number of RES instances on performance

The URES system originally employs 101 pre-trained RES model instances, where every instance corresponds to an α value between 0 and 1 with 0.01 increments. In this

experiment, we examine how lowering the computational load by considering a fewer number of RES model instances affects the URES performance. This is done by applying identical training and testing processes as for the original URES framework, but with α increments now taken from the set $\{0.02, 0.05, 0.1, 0.25, 0.5\}$. In correspondence, the number of RES model instances examined are the set $\{51, 21, 11, 5, 3\}$, where for example taking an increment of 0.25 includes $\alpha \in \{0, 0.25, 0.5, 0.75, 1\}$ and an increment of 0.5 includes $\alpha \in \{0, 0.5, 1\}$. The number of RES and RDE model instances is identical, preserving the functionality of the framework. Across all increments, we fix the tolerance threshold pairs to $(\mathrm{TH_R}, \mathrm{TH_D}) = (5, 5)$ in dB. The motivation for this choice relates to how using less RES model instances, i.e., larger α increments, intrinsically decreases the average value of P per $(\mathrm{TH_R}, \mathrm{TH_D})$ pair. We wish to mitigate this bias and isolate the effect of how the α increment changes the AECMOS in the URES output. Based on Figs. 8.8–8.9, the average AECMOS degrades by more than 0.5 points when transitioning from 101 to 51 model instances. Narrowing down the number of instances even further lowers the average AECMOS to subjectively mediocre and below, reaching as low as 2.7 for scenarios with echo-path changes. The increase in the average Δ_R and average Δ_D values is also significant, almost doubling its size as the number of RES instances lowers from 101 to only 3. To summarize, employing the entire 101 RES model instances has a significant impact on the URES framework performance, mainly in terms of the average AECMOS. It should also be noted that modern hardware services can handle the requirements of the URES framework with the entire 101 model instances while confining with hands-free speech communication timing standards [ETS16]. Also, since the RES models and the following RDE models run inference in multi-thread processing, the buffering latency is not affected by changing the number of instances in the framework [CSP+23].

8.6 Conclusions

RES in double-talk periods is an integral requirement of many hands-free speech communication systems, and recent RES methods have shown impressive advancements on average benchmark-performance. However, existing studies do not support specific

user inputs, which has crucial commercial implications. In this work, we developed a user-centric framework for RES in double-talk, which introduces for the first time three attributes that aim to enhance user experience. First, the RESL and DSML of the RES output confine to a UOP, up to a given tolerance threshold. Second, our framework supports real-time tracking of changes in the UOP, which is essential in dynamic acoustic environment of rapidly-varying user preferences. Third, AECMOS maximization is applied to enhance the subjective speech quality of the output signal. Future work may involve a learning framework that maps acoustic information to UOP recommendations in real time.

Chapter 9

Deep Adaptation Control for Stereophonic Acoustic-Echo Cancellation

9.1 Introduction

In stereophonic hands-free speech communication, the near-end microphones may capture three types of acoustic signals; the desired speech, additional noises, and reverberant echoes. The echoes are nonlinearly distorted versions of the far-end signal played by loudspeakers and reverberate to the microphones via echo paths [SMH95]. These echoes may impede conversation intelligibility as perceived by the far-end participant. The SAEC task is two-fold; tracking the near-end echo-paths and subtracting them from the microphones signals, and communicating the undistorted desired-speech signal to the far-end [BMS98].

The popular NLMS adaptive filter is numerically stable and efficient [SB99, PCBG15]. Its SNLMS variation employs the polarity of the adaptation error [FZ11] and is favorable over the NLMS due to its protection against abrupt noises[FD93, NCC16, LWS16]. The adaptation of the SNLMS filter is governed by the step-size parameter, which balances the convergence pace and the adaptation accuracy of the filter. Controlling the step-size is desirable in scenarios of frequent acoustic changes, e.g., echo-path variations

and single-to-double-talk transitions. The VSS problem has motivated Haubner et al. to employ deep learning for near-end speech [HHBK21] and noise [HBEK21] evaluation, and to reduce the error of the adaptive process[HBK22]. Meta-learning-based solutions have also recently emerged in [CBS22]. The a priori adaptation error and the far-end signal undergo feature extraction for VSS estimate in [MK16b] and a NPVSS minimized the adaptation error in [BRVT06]. The mean-error SVSS combines adaptation-error history with current adaptation-error estimate [HA16].

The methods in [HHBK21, HBEK21, HBK22] model the far-end signal as linear with its respective echo-signal, and the studies in [MK16b, BRVT06, HA16] consider the echo path as time-invariant. Unfortunately, both assumptions restrict performance in realistic setups and may cause low adaptation accuracy with slow convergence-pace [ICB21b]. On top of that, parameter-tuning, as in the NPVSS [BRVT06], involves heuristics that are inaccurate in practice. Thus, SAEC in real-life scenarios remains a relevant challenge and an active research area.

Inspired by [ICB22a], we mitigate these disparities by introducing a data-driven framework for DVSS that avoids heuristics and does not require acoustic setup hypotheses. First, the update rule of the adaptation process, governed by the step size, integrates the widely-linear model in the complex time domain. The mismatch between the actual echo paths and their filtered estimate is quantified by the normalized misalignment, which is then minimized with respect to the step size. A NN relates acoustic signals to the optimal step-size in training, and the predicted step-size feeds the SNLMS filter in real time for tracking the echo paths. The described framework is novel for SAEC.

We compare our approach with the competition by considering a pair of near-end loudspeakers and microphones, although this framework generalizes to any number of channels. Experimenting with 100 hours from the AEC-challenge corpus [CSP+21] reveals the consistent advantage of the DVSS in single and double-talk periods across various acoustic setups. The DVSS-SNLMS system also re-converges more rapidly and accurately after abrupt echo-path changes and is more robust to single-to-double-talk transitions.

The remainder of this chapter is organized as follows. In Section 9.2, we formulate

the problem. In Section 9.3, we describe the proposed solution. In Section 9.4, we lay out the experimental setup. In Section 9.5, we present the experimental results. Finally, in Section 9.6, we draw conclusions.

9.2 Problem Formulation

Our DVSS-SNLMS setup is in Figure 9.1. The left and right near-end microphones $m_{\mathrm{L}}(n)$ and $m_{\mathrm{R}}(n)$ at time index n are, respectively,

$$m_{\mathrm{L}}(n) = s_{\mathrm{L}}(n) + y_{\mathrm{L}}(n) + w_{\mathrm{L}}(n), \tag{9.1}$$

$$m_{\mathrm{R}}(n) = s_{\mathrm{R}}(n) + y_{\mathrm{R}}(n) + w_{\mathrm{R}}(n), \tag{9.2}$$

where $s_{\mathrm{L}}(n)$ and $s_{\mathrm{R}}(n)$ are the near-end speech signals, $w_{\mathrm{L}}(n)$ and $w_{\mathrm{R}}(n)$ represent environmental and system noises, and $y_{\mathrm{L}}(n)$ and $y_{\mathrm{R}}(n)$ are the nonlinear reverberant echo signals, as correspondingly captured by the left and right microphones:

$$y_{\mathrm{L}}(n) = \mathbf{h}_{\mathrm{LL}}^{T}(n)\,\mathbf{x}_{\mathrm{L}}^{\mathrm{NL}}(n) + \mathbf{h}_{\mathrm{RL}}^{T}(n)\,\mathbf{x}_{\mathrm{R}}^{\mathrm{NL}}(n), \tag{9.3}$$

$$y_{\mathrm{R}}(n) = \mathbf{h}_{\mathrm{LR}}^{T}(n)\,\mathbf{x}_{\mathrm{L}}^{\mathrm{NL}}(n) + \mathbf{h}_{\mathrm{RR}}^{T}(n)\,\mathbf{x}_{\mathrm{R}}^{\mathrm{NL}}(n). \tag{9.4}$$

For sake of readability, in this chapter we define notations by using explicit vector representations. Here, $\mathbf{x}_{\mathrm{L}}^{\mathrm{NL}}(n)$ and $\mathbf{x}_{\mathrm{R}}^{\mathrm{NL}}(n)$ respectively denote the L-recent samples from the left and right far-end signals, i.e., $\mathbf{x}_{\mathrm{L}}(n)$ and $\mathbf{x}_{\mathrm{R}}(n)$, subsequent to nonlinear distortions by nonideal hardware [ICB21b]:

$$\mathbf{x}_{\mathrm{L}}^{\mathrm{NL}}(n) = \left[x_{\mathrm{L}}^{\mathrm{NL}}(n), \ldots, x_{\mathrm{L}}^{\mathrm{NL}}(n-L+1)\right]^{T}, \tag{9.5}$$

$$\mathbf{x}_{\mathrm{R}}^{\mathrm{NL}}(n) = \left[x_{\mathrm{R}}^{\mathrm{NL}}(n), \ldots, x_{\mathrm{R}}^{\mathrm{NL}}(n-L+1)\right]^{T}, \tag{9.6}$$

and each of the column vectors $\mathbf{h}_{\mathrm{LL}}(n)$, $\mathbf{h}_{\mathrm{RL}}(n)$, $\mathbf{h}_{\mathrm{LR}}(n)$, $\mathbf{h}_{\mathrm{RR}}(n)$ has L samples and represents an echo path from the loudspeakers to the microphones, also known as a room impulse response (RIR). Instead of tracking $4L$ real-valued coefficients, we turn

to the more compact widely-linear model [BPGC11] by defining the complex signals:

$$\mathbf{h}\left(n\right)=\mathbf{h}_{1}\left(n\right)+j\mathbf{h}_{2}\left(n\right), \tag{9.7}$$

$$\mathbf{h}'\left(n\right)=\mathbf{h}'_{1}\left(n\right)+j\mathbf{h}'_{2}\left(n\right), \tag{9.8}$$

where $j=\sqrt{-1}$, and

$$\mathbf{h}_{1}\left(n\right)=0.5\Big(\mathbf{h}_{\mathrm{LL}}\left(n\right)+\mathbf{h}_{\mathrm{RR}}\left(n\right)\Big), \tag{9.9}$$

$$\mathbf{h}_{2}\left(n\right)=0.5\Big(\mathbf{h}_{\mathrm{RL}}\left(n\right)-\mathbf{h}_{\mathrm{LR}}\left(n\right)\Big), \tag{9.10}$$

$$\mathbf{h}'_{1}\left(n\right)=0.5\Big(\mathbf{h}_{\mathrm{LL}}\left(n\right)-\mathbf{h}_{\mathrm{RR}}\left(n\right)\Big), \tag{9.11}$$

$$\mathbf{h}'_{2}\left(n\right)=-0.5\Big(\mathbf{h}_{\mathrm{RL}}\left(n\right)+\mathbf{h}_{\mathrm{LR}}\left(n\right)\Big). \tag{9.12}$$

The complex echo signal $y\left(n\right)=y_{\mathrm{L}}\left(n\right)+jy_{\mathrm{R}}\left(n\right)$ can now be expressed in a widely-linear manner by $y\left(n\right)=\tilde{\mathbf{h}}^{H}\left(n\right)\tilde{\mathbf{x}}^{\mathrm{NL}}\left(n\right)$:

$$\tilde{\mathbf{h}}\left(n\right)=\begin{bmatrix}\mathbf{h}\left(n\right)\\ \mathbf{h}'\left(n\right)\end{bmatrix}, \tag{9.13}$$

$$\tilde{\mathbf{x}}^{\mathrm{NL}}\left(n\right)=\begin{bmatrix}\mathbf{x}^{\mathrm{NL}}\left(n\right)\\ \mathbf{x}^{\mathrm{NL}^{*}}\left(n\right)\end{bmatrix}, \tag{9.14}$$

where $\mathbf{x}^{\mathrm{NL}}\left(n\right)=\mathbf{x}_{\mathrm{L}}^{\mathrm{NL}}\left(n\right)+j\mathbf{x}_{\mathrm{R}}^{\mathrm{NL}}\left(n\right)$. The superscripts ^{H} and * correspondingly notate the transpose-conjugate and conjugate operations. As a result, the complex microphone signal $m\left(n\right)=m_{\mathrm{L}}\left(n\right)+jm_{\mathrm{R}}\left(n\right)$ can be formulated by

$$m\left(n\right)=\tilde{\mathbf{h}}^{H}\left(n\right)\tilde{\mathbf{x}}^{\mathrm{NL}}\left(n\right)+s\left(n\right)+w\left(n\right), \tag{9.15}$$

where $s\left(n\right)=s_{\mathrm{L}}\left(n\right)+js_{\mathrm{R}}\left(n\right)$ and $w\left(n\right)=w_{\mathrm{L}}\left(n\right)+jw_{\mathrm{R}}\left(n\right)$.

The echo estimation $\hat{y}\left(n\right)=\widehat{\tilde{\mathbf{h}}}^{H}\left(n\right)\tilde{\mathbf{x}}\left(n\right)$, where $\tilde{\mathbf{x}}\left(n\right)$ and $\tilde{\mathbf{x}}^{\mathrm{NL}}\left(n\right)$ follow the same notation, is evaluated by tracking the $2L$ complex-coefficients of $\widehat{\tilde{\mathbf{h}}}\left(n\right)$ with the SNLMS

adaptive filter. Subsequently, the complex near-end speech estimate can be drawn by

$$e\left(n\right)=m\left(n\right)-\hat{y}\left(n\right)=\left(y\left(n\right)-\hat{y}\left(n\right)\right)+s\left(n\right)+w\left(n\right), \tag{9.16}$$

where $e\left(n\right)=e_{\mathrm{L}}\left(n\right)+je_{\mathrm{R}}\left(n\right)$. Our focus is two-fold; tracking and cancelling the echo signal, i.e. nullifying $y\left(n\right)-\hat{y}\left(n\right)$, and avoiding distortion of the near-end speech, i.e. preserving $s\left(n\right)$.

9.3 DVSS-SNLMS Filter for SAEC

9.3.1 Modeling the SNLMS Filter and Step-size in Double-talk

By placing (9.15) and the definition of $\hat{y}\left(n\right)$ into (9.16), we respectively derive the a priori and a posteriori errors of the SNLMS filter [PCBG15]:

$$\epsilon\left(n\right)=\tilde{\mathbf{h}}^{H}\left(n\right)\tilde{\mathbf{x}}^{\mathrm{NL}}\left(n\right)-\hat{\tilde{\mathbf{h}}}^{H}\left(n-1\right)\tilde{\mathbf{x}}\left(n\right)+s\left(n\right)+w\left(n\right), \tag{9.17}$$

$$e\left(n\right)=\tilde{\mathbf{h}}^{H}\left(n\right)\tilde{\mathbf{x}}^{\mathrm{NL}}\left(n\right)-\hat{\tilde{\mathbf{h}}}^{H}\left(n\right)\tilde{\mathbf{x}}\left(n\right)+s\left(n\right)+w\left(n\right). \tag{9.18}$$

The update rule of the $2L$ complex-valued filter coefficients is [FZ11]:

$$\hat{\tilde{\mathbf{h}}}\left(n\right)=\hat{\tilde{\mathbf{h}}}\left(n-1\right)+\mu\left(n\right)\tilde{\mathbf{x}}\left(n\right)\mathrm{sgn}\left(\epsilon^{*}\left(n\right)\right), \tag{9.19}$$

where $\hat{\tilde{\mathbf{h}}}\left(0\right)$ is a column vector of $2L$ zeros, the step-size is given by $\mu\left(n\right)\in\mathbb{R}$, and $\mathrm{sgn}\left(z\right)=z/|z|$ for every $z\in\mathbb{C}$, where $|\cdot|$ is the absolute value. From (9.17)–(9.19):

$$e\left(n\right)=\epsilon\left(n\right)-\mu\left(n\right)\mathrm{sgn}\left(\epsilon\left(n\right)\right)\tilde{\mathbf{x}}^{H}\left(n\right)\tilde{\mathbf{x}}\left(n\right). \tag{9.20}$$

We now force the a posteriori error to a complete echo-cancellation and extract the corresponding expression of the step-size $\mu\left(n\right)$ [BPGC11]. Assuming $s\left(n\right)$ and $w\left(n\right)$ are zero-mean and uncorrelated [PCBG15]:

$$\sigma_{e}^{2}\left(n\right)=\sigma_{s}^{2}\left(n\right)+\sigma_{w}^{2}\left(n\right), \tag{9.21}$$

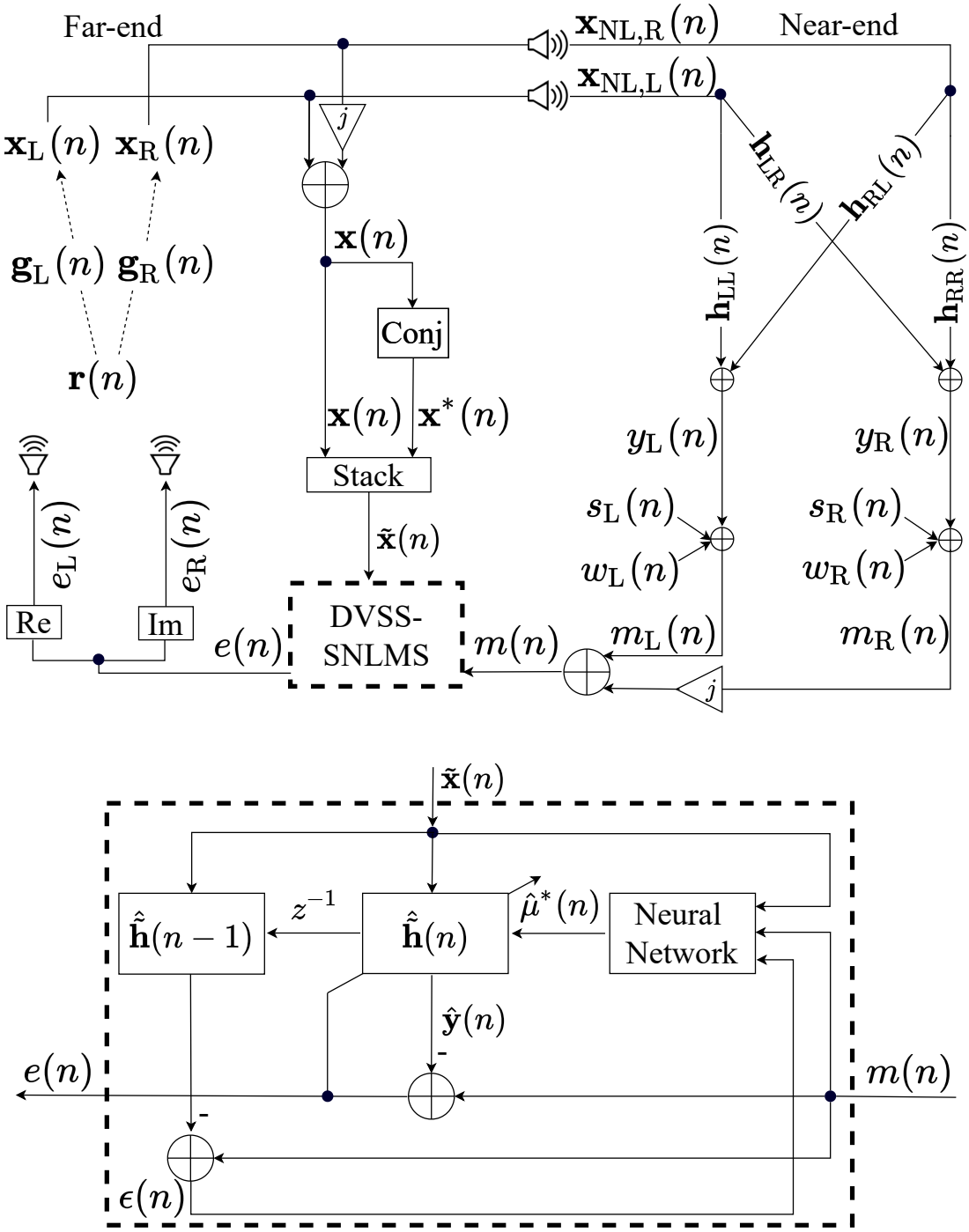

Figure 9.1: Top - SAEC scenario under the widely-linear model. Bottom - the DVSS-SNLMS block, where a NN estimates the step-size $\hat{\mu}^*(n)$ and the SNLMS filter estimates the acoustic paths via $\hat{\hat{\mathbf{h}}}(n)$.

where $\sigma_e^2(n) = E\left[|e(n)|^2\right]$ and $\sigma_s^2(n)$, $\sigma_w^2(n)$ follow the same definition. Now, the $E\left[|\cdot|^2\right]$ operator is applied on both sides of (9.20), and then (9.21) is substituted into

(9.20). This process yields

$$\mu(n) = c + \sqrt{\frac{\sigma_s^2(n) + \sigma_w^2(n) - \sigma_\epsilon^2(n)}{E\left(\left(\tilde{\mathbf{x}}^H(n)\,\tilde{\mathbf{x}}(n)\right)^2\right) + \delta} - c^2}, \tag{9.22}$$

where $c = E\left(|\epsilon|\tilde{\mathbf{x}}^H(n)\,\tilde{\mathbf{x}}(n)\right)/E\left(\left(\tilde{\mathbf{x}}^H(n)\,\tilde{\mathbf{x}}(n)\right)^2\right)$ and δ is a regularization parameter to avoid division by small numbers.

9.3.2 Step-size Optimization with a Data-driven Approach

The mismatch between the adaptive and true filter coefficients is often assessed using the normalized misalignment measure [BPGC11]:

$$\begin{aligned}\mathcal{D}(n) &= \frac{\left\|\tilde{\mathbf{h}}(n) - \hat{\tilde{\mathbf{h}}}(n)\right\|_2}{\left\|\tilde{\mathbf{h}}(n)\right\|_2} \\ &= \frac{\left\|\tilde{\mathbf{h}}(n) - \hat{\tilde{\mathbf{h}}}(n-1) - \mu(n)\,\tilde{\mathbf{x}}(n)\,\mathrm{sgn}\left(\epsilon^*(n)\right)\right\|_2}{\left\|\tilde{\mathbf{h}}(n)\right\|_2},\end{aligned} \tag{9.23}$$

where (9.19) was employed to the second transition and $\|\cdot\|_2$ is the ℓ_2 norm. We now solve a constrained nonlinear optimization problem [Rus11] to yield the optimal step-size. Formally, the normalized misalignment is minimized with respect to the step-size, in dB:

$$\mu^*(n) = \underset{0<\mu(n)<1}{\operatorname{argmin}}\, 20\log_{10}\mathcal{D}(n), \tag{9.24}$$

where the condition $0 < \mu(n) < 1$ is dictated by the stability requirements of NLMS-based adaptive filters [PCBG15]. The active-set optimization algorithm [HZ06] is utilized to perform optimization. According to (9.23), the only values involved in deriving $\mathcal{D}(n)$ are the far-end and the a priori error signals. This data-driven approach does not require heuristic parameter tuning to estimate $\mu^*(n)$.

9.3.3 Deep Adaptation to the Optimal Step-size

A deep NN is integrated into our system to model the relation derived in (9.22) between acoustic signals and the optimal step-size $\mu^*(n)$. Despite the near-end speech and noise signals being inaccessible in reality, the microphone signal can serve as an approximation. Thus, the microphone signal, along with the far-end and a priori error signals, are mapped to their respective step-size. The NN architecture is of a convolutional form [AMAZ17] with six input channels, one for the real and one for the imaginary part of each of the three input signals. These six waveforms undergo short-time Fourier transform (STFT) [GL84] separately before being fed to the NN. The specific architecture is standard and follows the one in [ICB22a]. During training, optimization is carried to minimize the ℓ_2 norm between the optimal step-size $\mu^*(n)$ and the output of the network. During inference, the step-size estimate $\hat{\mu}^*(n)$ is evaluated by the network and injected to an SNLMS filter that tracks the echo paths. Addressing complexity analysis, the NN and the SNLMS filter consume 4.2 Mflops and 4.8 MB of memory, by employing 1.05 Million parameters. Embedding this system into real-life edge devices for hands-free speech communication is thus considered feasible in terms of resources [ETS16]. One example of dedicated hardware for this task is the NDP120 neural processor by Syntiant$^{\text{TM}}$ [Syn21].

9.4 Experimental Setup

9.4.1 Database Acquisition

The database corpus utilized in this study includes 100 hours of noisy and clean segments taken from the AEC-challenge [CSP+21], where 25 hours are simulated recordings, and 75 hours are real recordings. The AEC-challenge data involves scenarios with no echo-paths change, where the near-end speaker and devices do not move, and scenarios with echo-paths change, where either the near-end speaker or devices are moving. We consider both double-talk periods and single-talk periods with far-end speakers only. Practically, audio clips are assigned to the original far-end source signal $\mathbf{r}(n)$ and to the near-end speech and noise signals, where $s_{\text{L}}(n) = s_{\text{R}}(n)$ and $w_{\text{L}}(n) = w_{\text{R}}(n)$ in

this study. To produce the far-end signals $\mathbf{x}_{\mathrm{L}}(n)$ and $\mathbf{x}_{\mathrm{R}}(n)$, $\mathbf{r}(n)$ is randomly propagated via one of 4500 pairs of RIRs that generate $\mathbf{g}_{\mathrm{L}}(n)$ and $\mathbf{g}_{\mathrm{R}}(n)$, i.e., the acoustic paths between $\mathbf{r}(n)$ and the left and right far-end microphones, respectively. To account for realistic acoustic environments, one of 4500 simulated nonlinear functions is applied to every $\mathbf{x}_{\mathrm{L}}(n)$ and $\mathbf{x}_{\mathrm{R}}(n)$ pair in a random fashion. These nonlinearities are modeled after realistic power amplifiers and loudspeakers in current hands-free hardware [ICB21b]. Each pair of nonlinearly-distorted far-end signals $\mathbf{x}_{\mathrm{L}}^{\mathrm{NL}}(n)$ and $\mathbf{x}_{\mathrm{R}}^{\mathrm{NL}}(n)$ is randomly propagated via one of 4500 foursomes of near-end RIRs. All RIRs are generated using the Image Method [AB79] with L coefficients and reverberation times RT_{60}, where $\mathrm{RT}_{60} \sim U[0.2, 0.5]$ seconds. The near-end SESR and SENR levels were drawn from $[-10, 10]$ dB and $[0, 40]$ dB, respectively, where $\mathrm{SESR}=10\log_{10}\left(|y(n)|^2/|s(n)|^2\right)$ and $\mathrm{SENR}=10\log_{10}\left(|y(n)|^2/|w(n)|^2\right)$ in dB [BPGC11]. These ratios are derived by running 20 ms frames that overlap by 50%.

9.4.2 Preprocessing, Training, and Testing

We recognize the well-known non-uniqueness problem in setups of SAEC, where strong coherence between $\mathbf{x}_{\mathrm{L}}^{\mathrm{NL}}(n)$ and $\mathbf{x}_{\mathrm{R}}^{\mathrm{NL}}(n)$ may degrade the adaptation process [GB02]. To mitigate that, we apply the following channel-wise transformation introduced in the context of the widely-linear model [BPGC11]. First, we define the positive and negative half-wave rectifiers [MHB01]:

$$\mathbf{x}_{\mathrm{L}}^{\mathrm{NL}'}(n) = \mathbf{x}_{\mathrm{L}}^{\mathrm{NL}}(n) + 0.5\left(\mathbf{x}_{\mathrm{L}}^{\mathrm{NL}}(n) + \|\mathbf{x}_{\mathrm{L}}^{\mathrm{NL}}(n)\|\right), \tag{9.25}$$

$$\mathbf{x}_{\mathrm{R}}^{\mathrm{NL}'}(n) = \mathbf{x}_{\mathrm{L}}^{\mathrm{NL}}(n) + 0.5\left(\mathbf{x}_{\mathrm{R}}^{\mathrm{NL}}(n) - \|\mathbf{x}_{\mathrm{R}}^{\mathrm{NL}}(n)\|\right). \tag{9.26}$$

With the element-wise operation $\tan\boldsymbol{\theta}(n) = \mathbf{x}_{\mathrm{L}}^{\mathrm{NL}'}{}_{\mathrm{R}}(n)/\mathbf{x}_{\mathrm{L}}^{\mathrm{NL}'}{}_{\mathrm{L}}(n)$:

$$\mathbf{x}_{\mathrm{L}}^{\mathrm{NL}''}(n) = \cos\boldsymbol{\theta}(n)\,\|\mathbf{x}^{\mathrm{NL}}(n)\|, \tag{9.27}$$

$$\mathbf{x}_{\mathrm{L}}^{\mathrm{NL}''}(n) = \sin\boldsymbol{\theta}(n)\,\|\mathbf{x}^{\mathrm{NL}}(n)\|, \tag{9.28}$$

where eqs. (9.27) and (9.28) use element-wise arithmetic. This transformation modifies only phase information, so employing $\mathbf{x}_{\mathrm{L}}^{\mathrm{NL}''}(n)$ and $\mathbf{x}_{\mathrm{L}}^{\mathrm{NL}''}(n)$ instead of $\mathbf{x}_{\mathrm{L}}^{\mathrm{NL}}(n)$ and

$\mathbf{x}_{\mathrm{R}}^{\mathrm{NL}}(n)$ allows a desired reduction in coherence with the advantage of little distortion.

The entire 100 hours batch of data is split to yield training, validation, and test sets of sizes 80 hours, 10 hours, and 10 hours, respectively. The split is random, but constrained to preserve balance and avoid bias by following the principles in [ICB21a]. Using the training and validation parts, the step-size is evaluated once every 8 ms according to (9.24) with parameter values of $\mu(0) = 3\times10^{-5}$ and $L = 2400$. Echo-paths are abruptly changed once every t seconds, where $t \sim U[4, 10]$, which characterizes in-the-wild conversations. Waveforms undergo STFT with running time frames that are 16 ms long and have 50% overlap. Before being inserted into the network, every STFT representation of every channel is attached to its 96 ms past context. Training the network using back-propagation involves learning rate of 10^{-4} that decays by 10^{-6} every 5 epochs, mini-batch size of 32 ms, and 40 epochs, using Adam optimizer [KB15]. The real-time inference is done on the test set. After the artificial gain of the network is calibrated according to [ICB21c], the step-size estimate is injected from the network output into the SNLMS, which constantly evaluates the echo paths. Training the network took 32 minutes for every 1 hours of input data from all channels. The inference time for the end-to-end system, from the network entry to the echo-paths estimate, is 26 ms on average using an Intel Core i7-8700K CPU @ 3.7 GHz with two GPUs of Nvidia GeForce RTX 2080 Ti.

9.4.3 Performance Measures

In single-talk periods with only far-end signals and noise presence, we estimate the echo suppression level between the microphone and enhanced signals using the ERLE [ITU12], defined as $10\log_{10}\left(|m(n)|^2/|e(n)|^2\right)$. In double-talk, we consider both the SDR [VGF06] and the PESQ [ITU17], where $\mathrm{SDR} = 10\log_{10}\left(|s(n)|^2/|e(n)-s(n)|^2\right)$ and is affected by both echo levels and speech distortion levels. The PESQ we report is the average of the PESQ score between $s_{\mathrm{L}}(n)$ and $e_{\mathrm{L}}(n)$, and the PESQ score between $s_{\mathrm{R}}(n)$ and $e_{\mathrm{R}}(n)$. These measures are derived with running time frames of 20 ms with an overlap of 50%. For a complete view of performance, we report the adaptation convergence times and convergence success rates. Convergence is considered achieved when $\mathcal{D}(n)$ falls below -10 dB and is considered successful if $\mathcal{D}(n)$ remains below -10

dB until echo-paths change [PCBG15].

9.5 Experimental Results

	SDR [dB]	PESQ	ERLE [dB]	Norm. Mis. [dB]
DVSS	**3.31±0.6**	**2.45±0.3**	**16.1±4.8**	**-17.3±4.4**
NPVSS	2.47±1.0	2.01±0.4	12.9±5.9	-14.8±4.6
NNVSS	2.41±1.0	1.86±0.5	12.5±6.0	-14.2±4.8
SVSS	2.12±0.9	1.73±0.5	10.7±6.5	-13.1±4.8
SNLMS	1.93±1.1	1.60±0.3	9.7±6.8	-11.6±5.2

Table 9.1: Performance with no echo-paths change.

	SDR [dB]	PESQ	ERLE [dB]	Norm. Mis. [dB]
DVSS	**3.05±0.8**	**2.27±0.4**	**10.9±6.3**	**-12.5±5.7**
NPVSS	2.20±1.2	1.79±0.5	7.9±6.5	-9.9±5.9
NNVSS	1.98±1.1	1.71±0.5	7.4±6.8	-9.4±6.1
SVSS	1.91±1.3	1.63±0.5	7.0±6.8	-9.2±5.9
SNLMS	1.74±1.5	1.51±0.3	6.6±6.3	-8.2±6.0

Table 9.2: Performance with echo-paths change.

DVSS	NPVSS	NNVSS	SVSS	SNLMS
4.4s, 79%	7.1s, 63%	8.5s, 55%	8.6s, 51%	9.1s, 48%

Table 9.3: Convergence times [sec] and success rates [%].

Our DVSS-SNLMS approach is matched against the VSS approaches in [MK16b, BRVT06, HA16], correspondingly abbreviated "NNVSS", "NPVSS", and "SVSS". These competing algorithms were integrated with the widely-linear model and the SNLMS filter for an unbiased comparison. The SNLMS filter with step-size of $\mu = 3 \times 10^{-5}$, briefly "SNLMS", is the classic approach baseline. Performance in Tables 9.1 and 9.2 is outlined with mean and std values, and Table 9.3 shows average test set values.

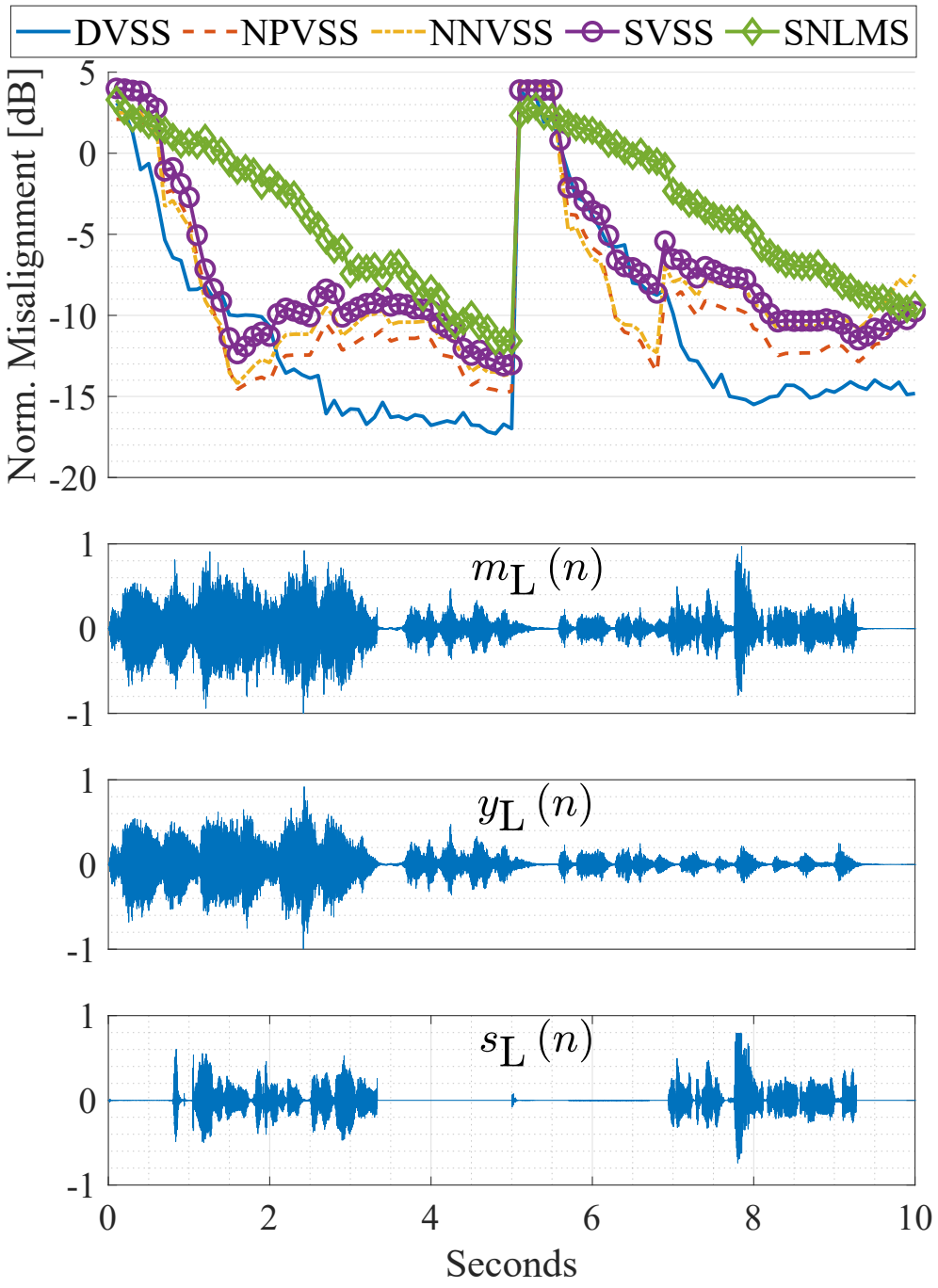

Figure 9.2: Convergence comparison to near-end echo paths abrupt change at 5 s, while SESR and SENR values regularly vary.

We distinguish between the performance when no echo-paths changes occur, i.e., in Table 9.1, from segments where echo-paths change, as in Table 9.2. In both cases, we only consider the post-convergence of the adaptive filter. In Table 9.1 and Table 9.2, the mean value of the results reflects the advantage of the DVSS method over the competition. The ERLE stresses the leading echo suppression of the DVSS method,

and the SDR and PESQ measures reveal its ability to maintain low speech distortion and high speech quality. It is also noted that the low std values of the DVSS indicate the stability of its performance across various acoustic setups. Table 9.3 affirms that our method achieves the shortest re-convergence times and the most successful convergence rates in scenarios with no echo-path changes. Unlike the competition, our method has shown a prominent ability to track echo paths by adapting the step-size accurately and rapidly while maintaining high robustness and generalization capabilities. This can be associated with our approach avoidance of heuristic parameter tuning and of making acoustic assumptions that often mismatch realistic scenarios.

Fig. 9.2 depicts the desired convergence behavior of the DVSS-SNLMS filter in a two-fold manner; it shows the most rapid convergence and re-convergence after abrupt echo-paths change, and it is also the least disturbed by the occurrence of double-talk. On the contrary, competing VSS-based methods slightly diverge due to double-talk, which impedes their convergence success afterward.

9.6 Conclusions

Controlling the step-size in adaptive filtering can allow for optimally operate between convergence rate and adaptation accuracy. This study attempts to bring this ability a step closer to practice by introducing a general adaptation-control framework that is both non-parametric and does not require acoustic assumptions and apply it to SAEC. Using the widely-linear model, we first derive the optimal step-size by minimizing the filter misalignment in the complex time domain. Then, we train a neural network to predict this optimal step-size from acoustic data in real time. Based on this step-size estimate, the SNLMS filter tracks the echo paths and performs well over competition across various acoustic setups. Future work may focus on generalization to scenarios where near-end microphones capture different versions of the speech and noise signals.

Chapter 10

Objective Metrics to Evaluate Residual-Echo Suppression During Double-Talk in the Stereophonic Case

10.1 Introduction

A conversation between a pair of speakers, based in near-end and far-end points, is common in hands-free communication. The desired-speech captured by the near-end microphone can be interrupted by echo, which is created by a loudspeaker that emits nonlinearly-distorted version of the far-end signal that reverberates in the room, and by additional noises [BGM$^+$01]. An acoustic coupling between the loudspeaker and the microphone potentially occurs due to this echo presence, which impairs the quality of acoustic information transmitted to the far-end [GV92]. In SAEC, the echo paths between a pair of loudspeakers and a pair of microphones are modeled by adaptive filtering. The echo paths are converted into acoustic-echo approximations that are subtracted from the microphones [SMH95, BMS98]. Double-talk segments are most challenging, since the echoes overlap with desired speech. Various studies tried to cope with it by preserving the speech and removing the echoes [SBP$^+$13, PBC14, CRPP12,

KS17, MHB01, WQW10, RCP+10, GT98]. In practice, however, echo paths are not estimated accurately, e.g., when the adaptive filter has not yet converged [BGM+01]. Therefore, a RES system must succeed the SAEC system to eliminate the echoes.

Subjective human evaluation is currently the most accurate assessment of human perception for speech quality [RBP+19, CNL+21]. Recently, an objective metric called the AECMOS was introduced. In double-talk specifically, the AECMOS has obtained impressive accuracy in estimating human ratings [PSS+22]. In contrast, RES systems conventionally use the SDR metric [VGF06] to assess speech quality in double-talk, e.g., in [CSVH18, DDBW19, PP20, CXCL20, Fan20b, Fan20a, WJ11, KJS21]. It will be empirically shown that the SSDR is by definition influenced by both distortion of stereo speech and presence of stereo residual-echo. Thus, for the task of RES in the stereophonic case, the SSDR is not an adequate indicator of neither the human evaluation for quality of speech nor of the AECMOS.

To combat it, we introduce a pair of objective metrics to distinctly assess the SDSML and the SRESL in double-talk. We first consider an RES system that acts as a time-dependent gain, with a pair of input and output channels. To calculate the SDSML, this gain is projected into the stereo desired-speech and the result is substituted inside the SSDR expression. The SRESL requires an estimate of the noisy stereo residual-echo, achieved by subtracting the stereo desired-speech from the double-talk frame. The ratio between this estimate without and with the gain applied to it generates the SRESL. The SDSML and SRESL metrics are evaluated with an RES system, based on deep learning, which incorporates a tunable design parameter. This study employs 100 hours of recordings that comprise of real signals and of simulations in various acoustic setups, with a range of echo and noise levels. Results reveal the AECMOS is well correlated with the SDSML and SRESL with high generalization to various scenarios. An additional empirical study investigates how the design parameter affects the tradeoff between the SDSML and SRESL. We then show how varying the design parameter during training can benefit interchangeable user demands of the RES system, which often occur in real-life. This study extends a recent work by the authors, which address the monophonic AEC case [ICB21c].

The remainder of this chapter is organized as follows. In Section 10.2, we formulate

the problem. In Section 10.3, we describe the proposed solution. In Section 10.4, we revisit the tunable design parameter. In Section 10.5, we lay out the experimental setup. In Section 10.6, we present the experimental results. Finally, in Section 10.7, we draw conclusions.

10.2 Problem Formulation

The RES scenario in the stereophonic case is detailed in Figure 10.1. Here, bold letters notate vectors and matrices, and normal letters notate scalars. The left and right near-end microphones $m_{\mathrm{L}}(n)$ and $m_{\mathrm{R}}(n)$ at time index n are respectively:

$$m_{\mathrm{L}}(n) = s_{\mathrm{L}}(n) + y_{\mathrm{L}}(n) + w_{\mathrm{L}}(n), \tag{10.1}$$

$$m_{\mathrm{R}}(n) = s_{\mathrm{R}}(n) + y_{\mathrm{R}}(n) + w_{\mathrm{R}}(n), \tag{10.2}$$

where $s_{\mathrm{L}}(n)$ and $s_{\mathrm{R}}(n)$ are the near-end speech signals, $w_{\mathrm{L}}(n)$ and $w_{\mathrm{R}}(n)$ represent environmental and system noises, and $y_{\mathrm{L}}(n)$ and $y_{\mathrm{R}}(n)$ are the nonlinear reverberant echo signals, as correspondingly captured by the left and right microphones:

$$y_{\mathrm{L}}(n) = \mathbf{h}_{\mathrm{LL}}^{T}(n)\,\mathbf{x}_{\mathrm{L}}^{\mathrm{NL}}(n) + \mathbf{h}_{\mathrm{RL}}^{T}(n)\,\mathbf{x}_{\mathrm{R}}^{\mathrm{NL}}(n), \tag{10.3}$$

$$y_{\mathrm{R}}(n) = \mathbf{h}_{\mathrm{LR}}^{T}(n)\,\mathbf{x}_{\mathrm{L}}^{\mathrm{NL}}(n) + \mathbf{h}_{\mathrm{RR}}^{T}(n)\,\mathbf{x}_{\mathrm{R}}^{\mathrm{NL}}(n). \tag{10.4}$$

For sake of readability, in this chapter we define notations by using explicit vector representations. Here, $\mathbf{x}_{\mathrm{L}}^{\mathrm{NL}}(n)$ and $\mathbf{x}_{\mathrm{R}}^{\mathrm{NL}}(n)$ respectively denote the L last samples of the left and right far-end signals, $\mathbf{x}_{\mathrm{L}}^{\mathrm{NL}}(n)$ and $\mathbf{x}_{\mathrm{R}}^{\mathrm{NL}}(n)$, after nonlinear distortions by nonideal hardware [ICB21b]:

$$\mathbf{x}_{\mathrm{L}}^{\mathrm{NL}}(n) = \left[x_{\mathrm{L}}^{\mathrm{NL}}(n), \ldots, x_{\mathrm{L}}^{\mathrm{NL}}(n-L+1)\right]^{T}, \tag{10.5}$$

$$\mathbf{x}_{\mathrm{R}}^{\mathrm{NL}}(n) = \left[x_{\mathrm{R}}^{\mathrm{NL}}(n), \ldots, x_{\mathrm{R}}^{\mathrm{NL}}(n-L+1)\right]^{T}, \tag{10.6}$$

and each of the column vectors $\mathbf{h}_{\mathrm{LL}}(n)$, $\mathbf{h}_{\mathrm{RL}}(n)$, $\mathbf{h}_{\mathrm{LR}}(n)$, $\mathbf{h}_{\mathrm{RR}}(n)$ has L samples and represents a RIR from the loudspeakers to the microphones. Preliminary, linear echo is reduced by employing the system in [ICB22a]. This system receives $m_{\mathrm{L}}(n)$ and

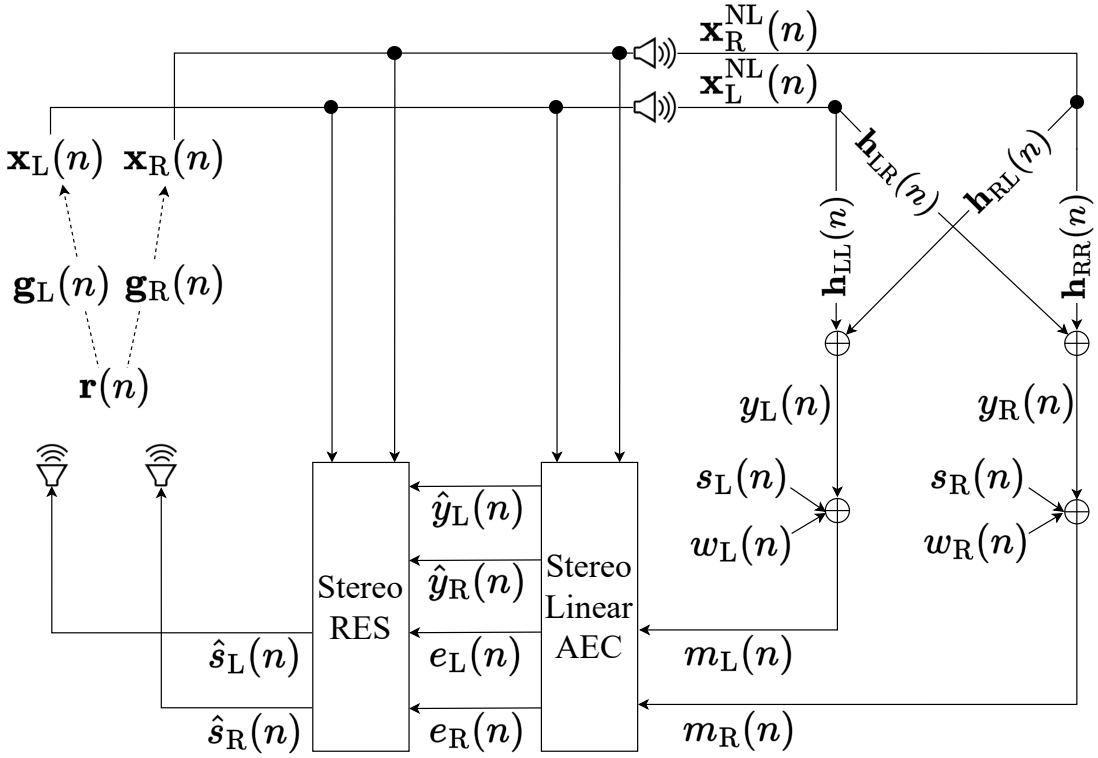

Figure 10.1: RES scenario in the stereophonic case.

$m_{\mathrm{R}}(n)$ as inputs, and $\mathbf{x}_{\mathrm{L}}(n)$ and $\mathbf{x}_{\mathrm{R}}(n)$ as reference channels, and generates two pairs of signals: a pair of echo estimates $\hat{y}_{\mathrm{L}}(n)$ and $\hat{y}_{\mathrm{R}}(n)$, and a pair of near-end speech signal estimates $e_{\mathrm{L}}(n)$ and $e_{\mathrm{R}}(n)$, given by:

$$e_{\mathrm{L}}(n) = m_{\mathrm{L}}(n) - \hat{y}_{\mathrm{L}}(n) = (y_{\mathrm{L}}(n) - \hat{y}_{\mathrm{L}}(n)) + s_{\mathrm{L}}(n) + w_{\mathrm{L}}(n), \tag{10.7}$$

$$e_{\mathrm{R}}(n) = m_{\mathrm{R}}(n) - \hat{y}_{\mathrm{R}}(n) = (y_{\mathrm{R}}(n) - \hat{y}_{\mathrm{R}}(n)) + s_{\mathrm{R}}(n) + w_{\mathrm{R}}(n). \tag{10.8}$$

The RES system aims to suppress the residual echoes, i.e., both $y_{\mathrm{L}}(n) - \hat{y}_{\mathrm{L}}(n)$ and $y_{\mathrm{R}}(n) - \hat{y}_{\mathrm{R}}(n)$, without distorting the desired-speech signals, i.e., $s_{\mathrm{L}}(n)$ and $s_{\mathrm{R}}(n)$.

10.3 The SDSML and SRESL Metrics

The SDSML and SRESL are developed by assuming a two-input and two-output RES system that acts as a time-varying gain matrix. The gain matrix in double-talk periods

is given by:

$$\mathbf{g}(n) = 0.5 \begin{bmatrix} \hat{s}_{\mathrm{L}}(n)/e_{\mathrm{L}}(n) & \hat{s}_{\mathrm{L}}(n)/e_{\mathrm{R}}(n) \\ \hat{s}_{\mathrm{R}}(n)/e_{\mathrm{L}}(n) & \hat{s}_{\mathrm{R}}(n)/e_{\mathrm{R}}(n) \end{bmatrix}, \tag{10.9}$$

where in double-talk $e_{\mathrm{L}}(n) \neq 0$ and $e_{\mathrm{R}}(n) \neq 0$. Before introducing the SDSML and SRESL definitions, we inspect the shortcomings of the SSDR. Extending the traditional SDR definition [VGF06] to the stereophonic case, it follows that:

$$\begin{aligned} \mathrm{SSDR} &= 10\log_{10} \frac{\|\mathbf{s}(n)\|_2^2}{\|\mathbf{s}(n) - \hat{\mathbf{s}}(n)\|_2^2}\Big|_{\text{Double-talk}} \\ &= 10\log_{10} \frac{\|\mathbf{s}(n)\|_2^2}{\|\mathbf{s}(n) - \mathbf{g}(n)\,\mathbf{e}(n)\|_2^2}\Big|_{\text{Double-talk}}, \end{aligned} \tag{10.10}$$

where:

$$\mathbf{s}(n) = \begin{bmatrix} s_{\mathrm{L}}(n) \\ s_{\mathrm{R}}(n) \end{bmatrix}, \hat{\mathbf{s}}(n) = \begin{bmatrix} \hat{s}_{\mathrm{L}}(n) \\ \hat{s}_{\mathrm{R}}(n) \end{bmatrix}, \mathbf{e}(n) = \begin{bmatrix} e_{\mathrm{L}}(n) \\ e_{\mathrm{R}}(n) \end{bmatrix}. \tag{10.11}$$

Both stereo desired-speech distortion and stereo residual-echo presence influence the SSDR value. Since the SSDR employs the term $\mathbf{g}(n)\,\mathbf{e}(n)$, a scenario of distortion-free speech and echo and a scenario of distorted speech without echo may produce an identical SSDR value. These scenarios, however, are perceived differently by humans and present different AECMOS values. It will be empirically shown that the SSDR and subjective human perception are poorly matched according to the AECMOS. Reliable evaluation of RES systems during double-talk can be achieved by separating the quantification of speech distortion from one of residual-echo suppression. Such distinction is not provided by the AECMOS metric. Thus, a pair of objective metrics is introduced by separately employing $\mathbf{g}(n)$ to the stereo desired-speech and to the noisy stereo residual-echo estimate.

The SDSML definition is analogous to the SSDR, except that $\mathbf{g}(n)$ is projected to the stereo desired-speech $\mathbf{s}(n)$ solely:

$$\mathrm{SDSML} = 10\log_{10} \frac{\|\tilde{\mathbf{s}}(n)\|_2^2}{\|\tilde{\mathbf{s}}(n) - \mathbf{g}(n)\,\mathbf{s}(n)\|_2^2}\Big|_{\text{Double-talk}}. \tag{10.12}$$

Next, the noisy stereo residual-echo is evaluated as $\mathbf{r}(n) = \mathbf{e}(n) - \mathbf{s}(n)$, and the SRESL is calculated by:

$$\text{SRESL} = 10\log_{10} \frac{\|\mathbf{r}(n)\|_2^2}{\|\mathbf{g}(n)\,\mathbf{r}(n)\|_2^2}\Big|_{\text{Double-talk}}. \tag{10.13}$$

It is noted that a constant attenuation may occur by the RES system, which deviates the SDSML from its real value. The SDSML must be unvaried by this attenuation, so it is being restored as shown in [ICB21c]. Expressly, $\tilde{\mathbf{s}}(n) = \tilde{g}(n)\,\mathbf{s}(n)$, where:

$$\tilde{g}(n) = \frac{\langle \mathbf{g}(n)\,\mathbf{s}(n), \mathbf{s}(n) \rangle}{\|\mathbf{s}(n)\|_2^2}. \tag{10.14}$$

10.4 A Tunable Stereophonic RES System

An RES system based on deep learning, inspired by [ICB21b], is employed to assess the SDSML and SRESL metrics. It contains six input channels, and two output channels and operates in the waveform domain. The proposed architecture is comprised of blocks of NLMs. Each NLM comprises 3 GRUs that contain 16 cells each [CGCB14] and dropout [SHK+14a] in the recurrent layers, an FCNN with a two-neuron output, and a PLU activation function with trainable parameters [Nic18] that is applied to each output neuron. The architecture is modeled by 3 consecutive NLMs. The first NLM receives the outputs of the linear SAEC system, i.e. $\hat{y}_{\text{L}}(n)$, $\hat{y}_{\text{R}}(n)$, $e_{\text{L}}(n)$, $e_{\text{R}}(n)$, and the two reference channels $\mathbf{x}_{\text{L}}(n)$ and $\mathbf{x}_{\text{R}}(n)$, and emits two output channels. The two succeeding NLMs are fed with four entrances each; a pair of output channels of the previous NLM, and the two reference channels. The last NLM produces the speech estimates $\hat{s}_{\text{L}}(n)$ and $\hat{s}_{\text{R}}(n)$. A tunable design parameter $0 \leq \alpha \leq 1$, originally introduced in [ICB21a], controls an intrinsic tradeoff that occurs inside a customized loss function $J(\alpha)$:

$$\begin{aligned} J(\alpha) &= \alpha \cdot \text{SDSML}^{-1} + (1-\alpha) \cdot \text{SRESL}^{-1} \\ &= \alpha \cdot \left(\frac{\|\tilde{\mathbf{s}}(n)\|_2^2}{\|\tilde{\mathbf{s}}(n) - \mathbf{g}(n)\,\mathbf{s}(n)\|_2^2} \right)^{-1} + (1-\alpha) \cdot \left(\frac{\|\mathbf{r}(n)\|_2^2}{\|\mathbf{g}(n)\,\mathbf{r}(n)\|_2^2} \right)^{-1}. \end{aligned} \tag{10.15}$$

where during double-talk $\tilde{\mathbf{s}}(n), \mathbf{r}(n) \neq \mathbf{0}$ and $\mathbf{0}$ is a vector of zeros. The parameter α

compromises between the SDSML and SRESL values in the training stage while $J(\alpha)$ is minimized. As a result, the stereo desired-speech distortion and stereo residual-echo suppression levels that the system permits can be adjusted dynamically. For instance, setting $\alpha = 1$ forces the stereo desired-speech prediction to coincide with its ground truth. Shifting to $\alpha = 0$, however, focuses on suppressing the stereo residual-echo, but causes a more substantial stereo desired-speech distortion. Tuning α, i.e., tuning the SDSML-SRESL tradeoff, can be done dynamically during training.

This RES system contains 23 thousand parameters that consume 550 million FLOPS and 65 KB of memory. Its embedding on hands-free platforms is thus feasible, e.g., by considering the NDP120 neural processor by Syntiant$^{\text{TM}}$ [Syn21]. The preceding linear AEC system employs the SNLMS adaptive filter in the sub-band domain [ICB22a].

10.5 Experimental Setup

10.5.1 Database Acquisition

This study makes use of the AEC challenge database [CSP+21] that is sampled at 16 kHz and incorporates English double-talk segments both with and without echo-paths change. In scenarios of no echo-paths change, the near-end setup does not include movements. In scenarios of echo-paths change, however, the recording involves movement in the near-end, either by the speaker or the device. This database contains 75 hours of real clean and noisy recordings and additional 25 hours of synthetic data, which are assigned to the original far-end source signal $\mathbf{r}\left(n\right)$ and to the near-end speech and noise signals, where $s_{\text{L}}\left(n\right) = s_{\text{R}}\left(n\right)$ and $w_{\text{L}}\left(n\right) = w_{\text{R}}\left(n\right)$ in this study. To produce the far-end signals $\mathbf{x}_{\text{L}}\left(n\right)$ and $\mathbf{x}_{\text{R}}\left(n\right)$, $\mathbf{r}\left(n\right)$ is randomly propagated via one of 4500 pairs of RIRs that generate $\mathbf{g}_{\text{L}}\left(n\right)$ and $\mathbf{g}_{\text{R}}\left(n\right)$, i.e., the acoustic paths between $\mathbf{r}\left(n\right)$ and the left and right far-end microphones, respectively. To account for realistic acoustic environments, $\mathbf{x}_{\text{L}}\left(n\right)$ and $\mathbf{x}_{\text{R}}\left(n\right)$ randomly undergo one of 4500 artificial nonlinearities that confine with practical characteristics of power amplifiers and loudspeakers in modern hands-free devices [ICB21b]. Each pair of nonlinearly-distorted far-end signals $\mathbf{x}_{\text{L}}^{\text{NL}}\left(n\right)$ and $\mathbf{x}_{\text{R}}^{\text{NL}}\left(n\right)$ is randomly propagated via one of 4500 foursomes of near-end RIRs. All RIRs are generated using the Image Method [AB79] with L coefficients and reverber-

ation times RT_{60}, where $\mathrm{RT}_{60} \sim U[0.2, 0.5]$ seconds. The near-end SSER and SSNR levels were distributed on $[-10, 10]$ dB and $[0, 40]$ dB, respectively, and are defined as SSER=$10\log_{10}\left(\|\mathbf{s}(n)\|_2^2/\|\mathbf{y}(n)\|_2^2\right)$ and SSNR=$10\log_{10}\left(\|\mathbf{s}(n)\|_2^2/\|\mathbf{w}(n)\|_2^2\right)$ in dB, where both $\mathbf{y}(n)$ and $\mathbf{w}(n)$ follow the notations in eq. (10.11) and both ratios are calculated with 50% overlapping time frames of 5 seconds.

10.5.2 Data Preprocessing, Training, and Testing

The database is divided into 80 hours of training, 10 hours of validation, and 10 hours of test sets randomly. Bias is averted by following our conventions [ICB21a]. Since real scenarios often involve an abrupt change in the echo paths, we simulate these to reoccur every t seconds, where $t \sim U[4, 10]$, and set $L = 2400$. The NN is fed with 50% overlapping time frames of 20 ms and is trained with a learning rate of 10^{-4} that decays by 10^{-6} every 5 epochs, mini-batch size of 60 ms, and 40 epochs, using Adam optimizer [KB15] and back-propagation through time. Training the RES system lasted 25 minutes per 10 hours of data and inference took 8 ms per batch on an Intel Core i7-8700K CPU @ 3.7 GHz with two GPUs of type Nvidia GeForce RTX 2080 Ti.

10.5.3 Performance Measures

Performance is also evaluated with the SSDR, which is influenced by both echo presence and distortion of speech, and with the PESQ [ITU17] between $\mathbf{s}(n)$ and $\hat{\mathbf{s}}(n)$. The AECMOS is also reported, and is calculated using the API provided by Microsoft as the average between the AECMOS of $\hat{\mathbf{s}}_{\mathrm{L}}(n)$ and the AECMOS of $\hat{\mathbf{s}}_{\mathrm{R}}(n)$.

10.6 Experimental Results

Results are reported on the test set. In Tables 10.1 and 10.2, both mean and standard deviation (std) values are given. In Figures 10.7–10.10, only mean values are shown. Higher mean and lower std values entail better performance for all metrics. The linear filter convergence confines with the description in [ICB22a, PCBG15]. We use 50% overlapping time frames of 5 seconds for metrics calculations.

We employ the Pearson correlation coefficient (PCC) [BCHC09] and Spearman's

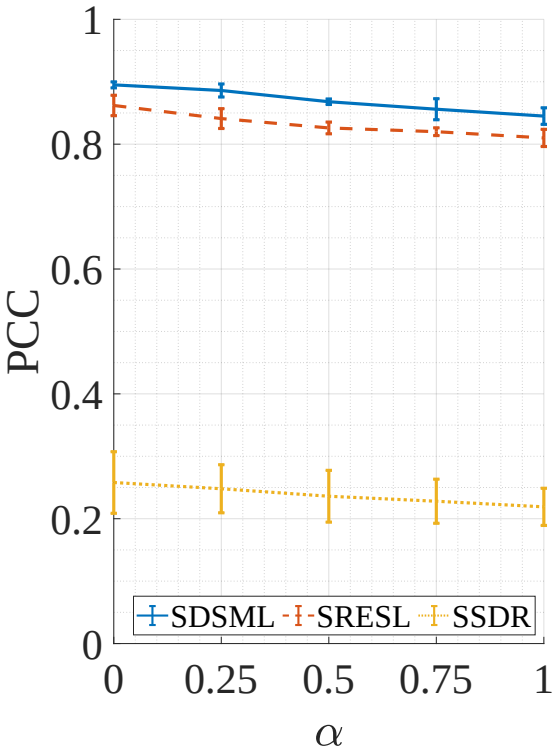

Figure 10.2: PCC of the AECMOS with the SDSML, SRESL, and SSDR metrics.

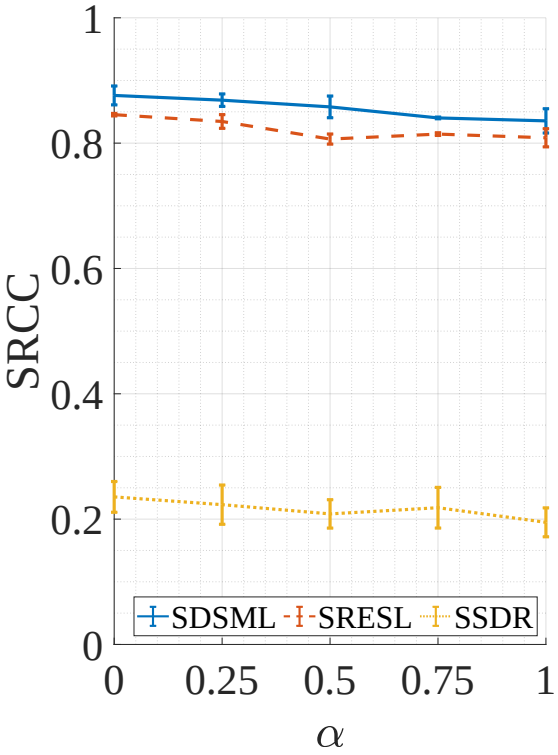

Figure 10.3: SRCC of the AECMOS with the SDSML, SRESL, and SSDR metrics.

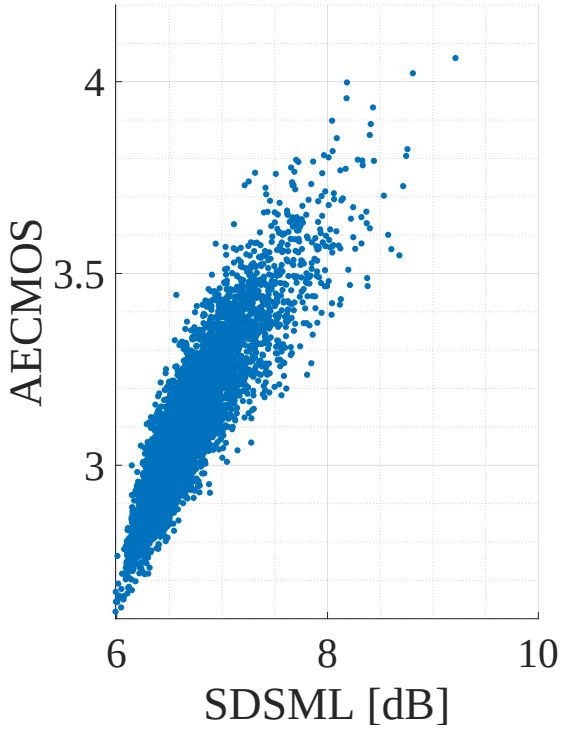

Figure 10.4: Scatter plots of the AECMOS versus the SDSML metric.

rank correlation coefficient (SRCC) [Gau01] to discover how much the SDSML and SRESL correlate with the AECMOS, similarly to [PSS+22, RGC21, SCS+21]. This experiment includes segments both with and without echo-paths change after the linear SAEC system has converged for $\alpha \in \{0, 0.25, 0.5, 0.75, 1\}$, and the results are shown in Figures 10.2–10.3. The SSDR and AECMOS are poorly correlated, as pointed out by the PCC and SRCC mean values that fall below 0.26 for all α. However, with average PCC and SRCC values between 0.8 and 0.89 for all α, the proposed SDSML and SRESL metrics highly coordinate with the AECMOS. Observing std values, the SDSML and SRESL show more consistent correlations across α than the SSDR. Figures 10.2–10.3 visualizes the AECMOS versus the SDSML, SRESL, and SSDR metrics for random sample values drawn from $\alpha \sim U[0.25, 0.75]$ with no echo-paths change. The low matching between the AECMOS and the SSDR and the high correlation between the AECMOS and the SDSML and SRESL are now verified. The SDSML and SRESL are therefore more indicative to subjective human perception of speech-quality evaluation than the SSDR, according to the AECMOS.

In Tables 10.1 and 10.2 performance metrics are evaluated for scenarios without

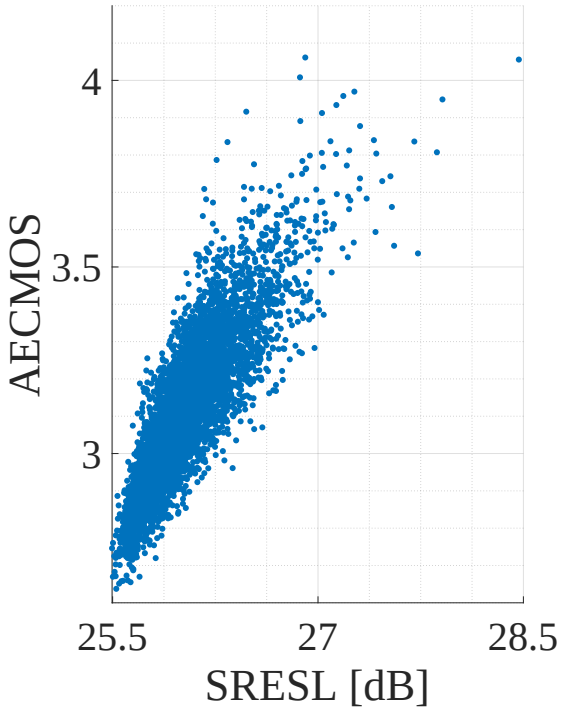

Figure 10.5: Scatter plots of the AECMOS versus the SRESL metric.

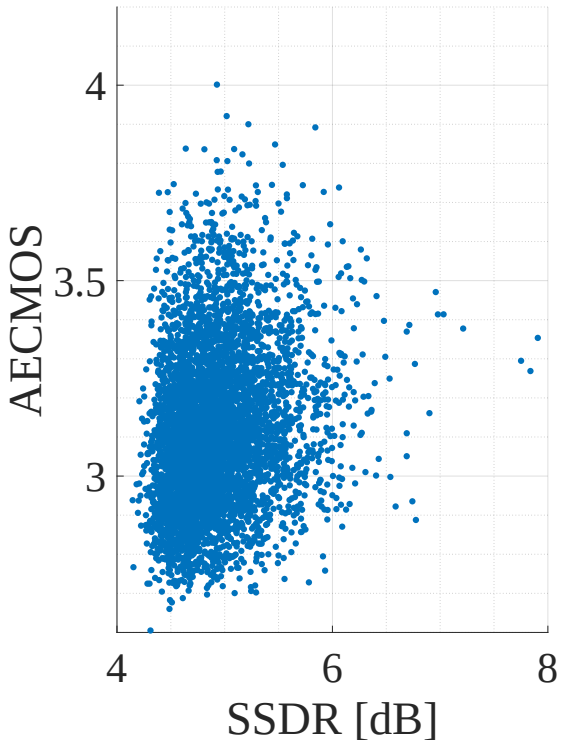

Figure 10.6: Scatter plots of the AECMOS versus the SSDR metric.

α	SDSML	SRESL	SSDR
0	6.15±0.6	29.6±3.3	4.55±0.9
0.5	7.38±0.6	26.4±3.4	5.52±0.8
1	8.13±0.5	23.1±3.8	6.73±0.7

Table 10.1: Performance with no echo-paths change.

α	SDSML	SRESL	SSDR
0	5.41±1.1	24.3±3.5	3.61±1.4
0.5	6.29±1.0	21.7±4.0	4.60±1.2
1	7.01±0.8	18.9±4.9	5.54±1.0

Table 10.2: Performance with echo-paths change.

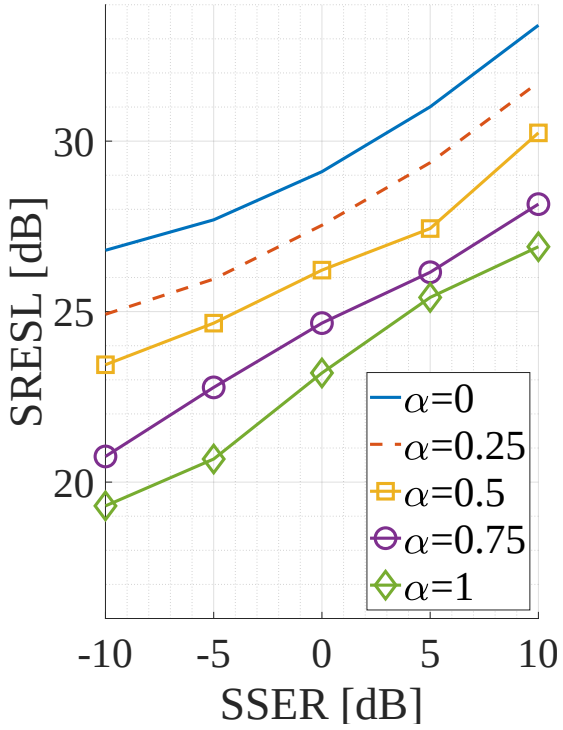

Figure 10.7: SRESL versus SSER for various values of α.

and with echo-paths change, respectively, after convergence with $\alpha \in \{0, 0.5, 1\}$. The trend of the SDSML and SRESL is consistent with the one of the SSDR, all of which deteriorate in the transition from no echo-paths change to echo-paths change periods. This consistency across various test set setups indicates high generalization of

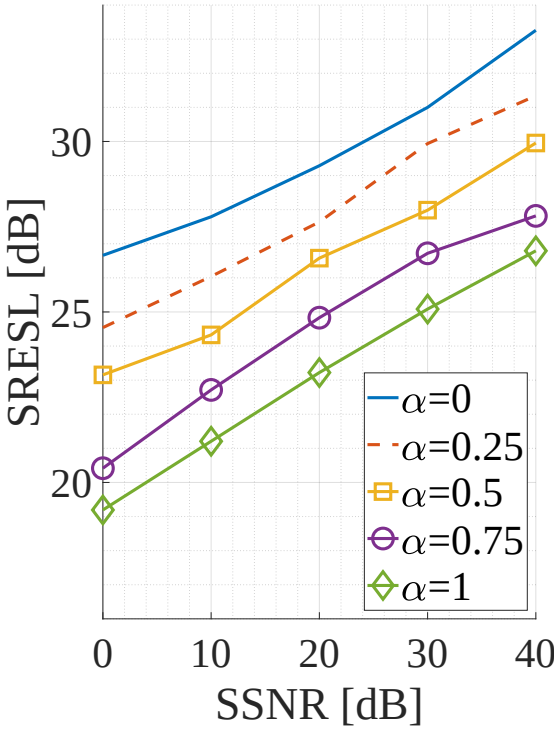

Figure 10.8: SRESL versus SSNR for various values of α.

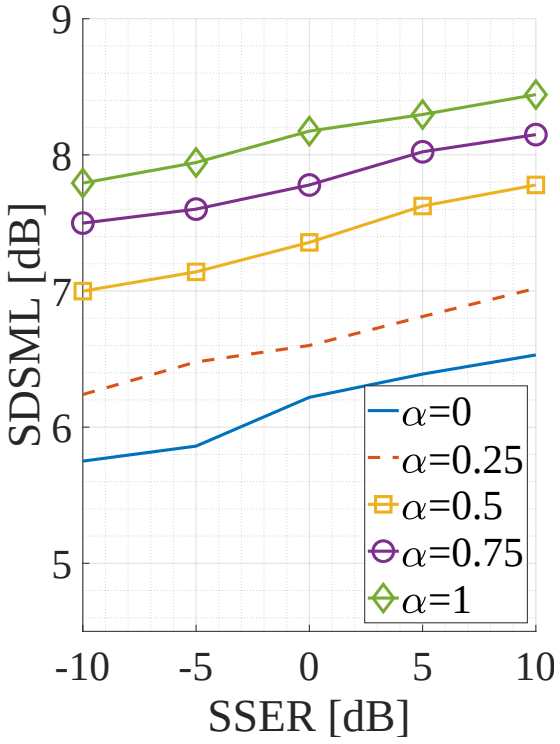

Figure 10.9: SDSML versus SSER for various values of α.

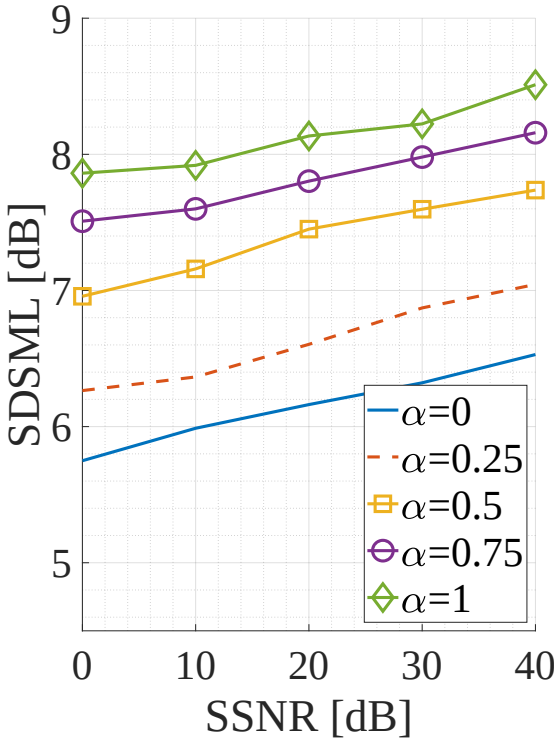

Figure 10.10: SDSML versus SSNR for various values of α.

the SDSML and SRESL. The average SDSML values are regularly higher than the average SSDR values, as expected. This is directly derived from eq. (10.10), in which the denominator takes into account both echo and noise. As values of α increase, the average SDSML values increase while the average SRESL values decrease, both with and without echo-paths change, as expected.

We now explore how α governs the tradeoff between the SDSML and SRESL. In Figures 10.7–10.10, results for no echo-paths change periods after convergence are included, for $\alpha \in \{0, 0.25, 0.5, 0.75, 1\}$. Lower α values relate to lower SDSML values because distortion is higher for stereo speech. The SRESL values rise, however, since more suppression is applied to the stereo residual-echo. Empirically, this tradeoff is consistent on average for all α. We also explore how practical SSER and SSNR levels influence this tradeoff. As expected, the more acoustic conditions degrade, the more the values of both metrics are impaired. Also, regardless of acoustic conditions, it is maintained that the lower α becomes, the lower the SDSML and the higher SRESL values appear.

Practical user demands of the RES system may vary. Thus, we propose a design

scheme that addresses this dynamic need. E.g., let us assume convergence has been achieved and no echo-paths change occurs. This can be verified by following the definitions in [ICB22a, PCBG15]. Initially, the user requires an average SRESL higher than 24.5 dB and an average SDSML higher than 7.6 dB. They inspect Figures 10.7–10.10 and select $\alpha = 0.75$. Next, the user concludes that SSER = 0 dB and SSNR = 20 dB, e.g., by respectively analyzing double-talk and near-end single-talk periods. Thus, they demand a SDSML no lower than 7.4 dB with a maximal SRESL. The user follows Figures 10.7–10.10 and decides to shift $\alpha = 0.75$ to $\alpha = 0.5$ during training. Indeed, this lowers the average SDSML to 7.45 dB and enhances the average SRESL to over 26.5 dB. These conclusions also hold for the monophonic AEC case [ICB21c].

10.7 Conclusions

We focused on the task of RES in the stereophonic case during double-talk. We first showed that the widely-used SSDR metric poorly correlates with human speech-quality ratings. We then proposed a pair of objective measures that distinct between desired-speech distortion and residual-echo suppression during double-talk. By considering a deep RES system with a tunable parameter α, we showed that the SDSML and SRESL correlate well with the AECMOS metric, which may render they are more appropriate to assess quality of speech. Also, by tuning α during training, we offered a practical design scheme that allows flexible adjustment of the RES system to a specific SDSML-SRESL tradeoff. Future work may focus on enhancing subjective experience for RES systems during double-talk periods by optimizing the AECMOS through tuning of α.

Chapter 11

Voice-Activity Detection for Transient Noisy Environment Based on Diffusion Nets

11.1 Introduction

voice-activity detection refers to a family of methods that perform segmentation of an audio signal into parts that contain speech and silent parts. In this study, audio signals are captured by a single microphone and contain clean sequences of speech and silence. These signals are mixed with stationary and non-stationary noises (transients), e.g., door knocks and keyboard tapping [DC14, DTC16]. Our objective is to correctly assign each captured audio frame into the category of speech presence or absence. A solution to this problem may benefit many speech-based applications such as speech and speaker recognition, speech enhancement, emotion recognition and dominant speaker identification.

In acoustic environments that contain neither stationary or non-stationary noise, speech is detected by using methods that rely on frequency and energy values in short time frames [KN91, JMR94, VGX97]. These methods show significant deterioration in performance when noise is present, even with mild levels of SNRs. To cope with this problem, several approaches assume statistical models of the noisy signal in order

to estimate its parameters [CK11, CKM06, SKS99, RSB+04, CB01, Coh03]. Nonetheless, these methods are incapable of properly modeling transient interferences, which constitute an essential part of this study. Ideas that involve dimensionality reduction through kernel-based methods are introduced in [DTC15], where both supervised and unsupervised approaches have been exploited. However, its main limitation is a significant low-dimensional overlap between speech and non-speech representations.

Machine learning techniques have been employed for voice-activity detection in recent studies [SCK10, WZ11]. In contrast to classic methods, these approaches learn to implicitly model data without assuming an explicit model of a noisy signal. In particular, deep learning based methods have gained popularity in recent years due to a substantial increase in both computational power and data resources. Mendelev et al. [MPP15] constructed a deep neural network for voice-activity detection, and suggested to employ the dropout technique [SHK+14b] for enhanced robustness. The main drawback of this method is that temporal information between adjacent audio frames is ignored, due to independent classification of each time frame. Studies presented in [LHB15, GMH13, HM13, HL13] used a RNN to integrate temporal context with the use of past frames. However, the rapid time variations and prominent energy values of non-stationary noises in comparison to speech are still the main cause of degraded performance in these methods. A recent study conducted by Ariav et al. [ADC18a] proposed to use an auto-encoder to implicitly learn an audio signal embedded representation. To enhance temporal relations between frames, this auto-encoder feeds an RNN. Despite its leading performance, the reported results are still unsatisfactory. Our study found that the main limitation of this algorithm is the dense low-dimensional representation forced by the auto-encoder and into the RNN. This density occurs largely due to the joint training of speech and non-speech frames, which fails to enhance their unique features. Thus, their low-dimensional representations, which are the sole information that feeds the RNN, are embedded closely in terms of Euclidean distance. Eventually, this poses a difficulty in separation of speech from non-speech frames based merely on temporal information, which is the core advantage of using RNN architecture.

In this work, we propose an algorithm that addresses the limitations found in the methods proposed in [DTC15] and [ADC18a]. We independently learn the low-

dimensional spatial patterns of speech and non-speech audio frames through the DM method. DM is a method that performs non-linear dimensionality reduction by mapping high-dimensional data points to a manifold, embedded in a low-dimensional space [TCGC13]. The mapped coordinates that lay on this manifold are referred to as DM coordinates. Since this method preserves locality, frames with similar contents in the original high dimension are mapped closely in the low, embedded dimension, with respect to their Euclidean distance. We separately apply DM for speech and non-speech frames through a pair of independent deep encoder-decoder structures. Inspired by the Diffusion nets architecture [MSCC17], the end of each encoder is forced to coincide with the embedded DM coordinates of its high-dimensional input. This approach allows us to differ the intrinsic structure of speech from the ones of transients and background noises based on the Euclidean metric.

We suggest two variations for the voice-activity detection algorithm, one for real-time applications and one for batch processes. We test both approaches on five comparative experiments conducted in [DTC15, ADC18a, TIH$^+$10]. Results show enhanced voice-activity detection performance, that surpasses the known state-of-the-art speech detection results. Furthermore, our proposed architecture is more robust and has better generalization ability than competing methods, as demonstrated through experiments.

The remainder of this chapter is organized as follows. In Section 11.2, we formulate the problem. In Section 11.3, we describe the proposed solution. In Section 11.4, we lay out the database acquisition and feature extraction processes. In Section 11.5, we discuss the experimental setup. In Section 11.6, we present the experimental results. Finally, in Section 11.7, we draw conclusions.

11.2 Problem Formulation

Let $s\left(n\right)$ denote the following audio signal:

$$s\left(n\right) = s^{\mathrm{sp}}\left(n\right) + s^{\mathrm{st}}\left(n\right) + s^{\mathrm{t}}\left(n\right), \tag{11.1}$$

where sp, st and t stand for speech, stationary background noise and transient interference, respectively. The time domain signal is processed in overlapping time frames of length M. Let $\mathbf{f}_n \in \mathbb{R}^M$ denote the nth audio frame and let $\{\mathbf{f}_n\}_{n=1}^N$ denote the audio data set of N time frames. Let $\mathcal{H}^0$ and $\mathcal{H}^1$ be two hypotheses that stand for speech absence and presence, respectively. In addition, let $\mathbb{I}(\mathbf{f}_n)$ be a speech indicator of the nth audio frame, defined as:

$$\mathbb{I}(\mathbf{f}_n) = \begin{cases} 1, & \mathbf{f}_n \in \mathcal{H}^1 \\ 0, & \mathbf{f}_n \in \mathcal{H}^0 \end{cases}. \tag{11.2}$$

The goal of this study is to estimate $\mathbb{I}(\mathbf{f}_n)$, i.e., to correctly classify each audio frame $\mathbf{f}_n$ as a speech or non-speech frame.

11.3 Proposed Algorithm for Voice-Activity Detection

Our proposed approach comprises several steps, as illustrated in Fig. 11.1. Initially, feature extraction is employed in the time domain. The features include the MFCCs and their low-dimensional representation, generated by the DM method. A detailed description is given in Section 11.4.2. Subsequently, a deep encoder-decoder based neural network is used to learn the unique patterns of speech and non-speech signals. Since this structure makes use of the DM method, it is regarded in this study as DED. Next, error measures are extracted from the deep architecture. Those errors are represented in a coordinate system, notated in this study as error map. It should be highlighted that no mathematical operation is applied on the errors extracted from the network. i.e., the error map is merely a representation form which allows us to conduct better analysis and gain deeper insights on the performance of our detector, as will be shown along this paper. A classifier, fed by the coordinates of the error map, is constructed to separate speech presence and absence. In this study, two different modes are used for classification. First, a batch mode is considered, where a substantial corpus of speech and non-speech audio frames must be at hand, in order to evaluate the outcome of the DM process correctly. In batch mode, both low and high-dimensional

errors are taken into account during classification. The second classification mode is real-time, which exploits merely high-dimensional error information. In this case, integration of DM is not required, which allows a frame-by-frame classification with negligible delay.

11.3.1 Deep Encoder-Decoder Neural Network

Our approach suggests that speech frames can be separated from non-speech frames based on their intrinsic low-dimensional representation. Ideas from [MSCC17] are adopted to merge DM with two independent, identically constructed DEDs, notated by DED^i, where $i \in \{0, 1\}$. DM allows a geometric interpretation of the data by constructing its underlying embedding, which can be represented by the middle layer of any basic encoder-decoder network [ADC18a]. To exploit this property, the middle layer is forced to coincide with the true DM coordinates of the input layer. As a result, the encoder of DED^i is trained to map spectral features affiliated with $\mathcal{H}^i$ from their original space to the lower diffusion space. Subsequently, the decoder of DED^i learns the inverse mapping back to the high-dimensional feature space.

A deep architecture is constructed to implement the above notion, as illustrated in Fig. 11.1. In this proposed system, each DED comprises two stacked parts, an encoder and a decoder. The former is constructed from a 72 neurons input layer followed by two layers of 200 neurons each and a final layer of 3 neurons. The deep decoder is a reflection of the deep encoder. While the size of the middle and hidden layers are determined empirically, the size of the input (and thus, the output) layer of each DED is derived from the feature extraction process, as described in Section 11.4.2. In the output of each layer, an identical activation function is employed on each neuron (11.12).

11.3.2 Error Maps and Voice-Activity Detection Classifier

Let us denote a single observation of an input feature vector as $\mathbf{a}$ and its true DM coordinates as $\mathbf{m}$. Additionally, $\hat{\mathbf{m}}$ and $\hat{\mathbf{a}}$ denote the encoding of $\mathbf{a}$ by a trained encoder and its reconstruction by a trained decoder, respectively. Each observation is fed into the trained DEDs simultaneously. That way, the relations between each hypothesis and the constructed embeddings are compared under the same conditions. These measures

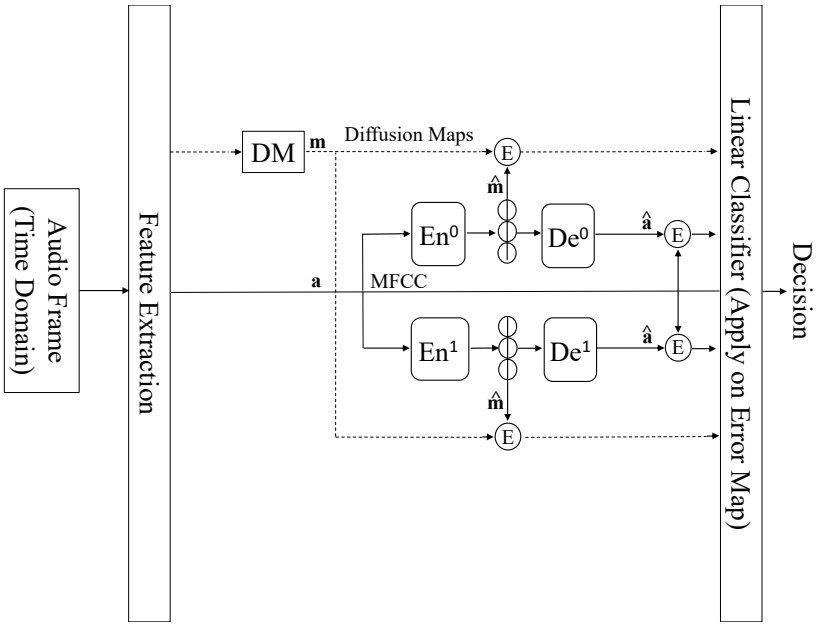

Figure 11.1: Proposed architecture for voice-activity detection. Dashed line is valid only for batch mode and solid line is constantly employed for both batch and real-time modes. 'En' and 'De' are abbreviations for encoder and decoder, respectively. Superscripts 0 and 1 relate to the index of the corresponding trained DED. The circled 'E' notation refers to an error calculation unit, defined in (11.3).

are employed through $e_{\text{en}}(\mathbf{m})$ and $e_{\text{de}}(\mathbf{a})$, where:

$$e_{\text{en}}(\mathbf{m}) = \|\mathbf{m} - \hat{\mathbf{m}}\|_1 \ ; \ e_{\text{de}}(\mathbf{a}) = \|\mathbf{a} - \hat{\mathbf{a}}\|_1, \tag{11.3}$$

while $\|\cdot\|_1$ denotes the ℓ_1 norm. Namely, as $e_{\text{en}}(\mathbf{m})$ represents the mapping error, $e_{\text{de}}(\mathbf{a})$ is associated with the reconstruction error of $\mathbf{a}$.

In this study, two classification modes are considered. In the batch mode, both $e_{\text{en}}(\mathbf{m})$ and $e_{\text{de}}(\mathbf{a})$ are taken into account. Namely, each observation $\mathbf{a}$ ultimately generates two pairs of errors, one from each DED. These errors are interpreted as a four-dimensional coordinate that is embedded into an error map. In the real-time mode, on the other hand, only the decoder error $e_{\text{de}}(\mathbf{a})$ is extracted from each DED. i.e., in this scenario a two-dimensional coordinate is embedded into the error map.

Subsequently, a SVM classifier with linear kernel is trained on the error map, which contains the generated error measures from a corpus of observations. The objective of this classifier is to separate between coordinates affiliated with different hypotheses. As a result, two decision regions are created, for speech presence and absence. Since DED^i

is trained to construct a low-dimensional manifold on which $\mathcal{H}^i$ is embedded, frames related to $\mathcal{H}^i$ highly fit the learned mapping of DED^i. This leads to substantially lower errors, which could be easily identified as a separate cluster. This assumption is derived from the property of the DM method, in which the diffusion distance in the original feature dimension is proportional to the ℓ_1 norm in the diffusion space. In this study, a classic SVM classifier is shown to be sufficient.

It is worth noting that we have also implemented an alternative architecture to the one presented in Fig. 11.1, which involves a unified network instead of an SVM. The goal of this was to assert the improvement, and thus justify the employment, of our suggested system over the alternative of the fully connected neural network, which is commonly used in deep learning algorithms. We concatenated the output layer of both DED branches to each other and to the input layer. Then, this augmented layer was connected to a single-bit output neuron that carries the VAD decision. Results have shown very similar performance, with a slight tendency to the SVM based method. As a results, we have decided to use the originally presented architecture. Two minor advantages of the SVM can be noted over the unified neural network. First, it is less computationally expensive in comparison to using an additional hidden layer, which will consume higher memory and time during back propagation. Second, the original method explicitly constructs the error measures and feeds them to the SVM, which leads to high separation of speech from silence. Therefore, the hidden layer attempts to implicitly represent the data in a similar manner, i.e., to find the relation between the neurons which will ultimately lead to good separation. Representing the error measures in the two-dimensional space and applying the SVM on it both serves as a more natural, intuitive classification algorithm and avoids the infamous "black box" property of the neural network, as well as grants us the ability to analyze the decision of the detector in a helpful and profound manner, as will be done later on.

11.4 Database and Feature Extraction

11.4.1 Database

We adopt the audio database presented in [DTC15] to construct a DED training set, a classifier training set and a back to end test set. This database is obtained from 11 different speakers reading aloud an article chosen from the web, while making natural pauses every few sentences. Naturally, these recordings are composed of sequences of speech followed by non-speech frames. Each sequence varies from several hundred milliseconds to several seconds in length. These signals were recorded with an estimated SNR of 25 dB at a sampling rate of 8 kHz. Each of the 11 signals is 120 seconds long and it is processed using short time frames of 634 samples with 50% overlap, which effectively generates a 25 frames/second rate. The clean speech signal $s^{\mathrm{sp}}(n)$, defined in (11.1), is used to determine the presence or absence of speech in each time frame, and to construct a label set accordingly.

These clean audio signals are contaminated by 42 different pairs of additive stationary and non-stationary noises, which construct a varied data set. The noise signals employed include white and colored Gaussian noise, babble and musical instruments. Transients include keyboard taps, scissors snapping, hammering and door knocks.

11.4.2 Feature Extraction

We wish to exploit the ability of deep neural networks to learn complex relations between their inputs and outputs. Hence, our objective is to feed our architecture with features that express the unique patterns of each hypothesis (11.2). To generate spectral information from the time domain database, MFCCs are employed. These coefficients are concatenated along a fixed number of adjacent frames, in order to gain temporal context between them. DM is applied to integrate spatial properties and to find a relation between the spectrum of the signal and its geometric low-dimensional structure.

Mel Frequency Cepstral Coefficients

Features based on a spectral representation of audio signals are fully adopted from the study of Dov et al. [DTC15]. To construct them, weighted MFCCs are employed.

MFCCs use the perceptually meaningful Mel-frequency scale, which allows a compact representation of the spectrum of speech [Log00]. MFCC features are used in the presence of highly non-stationary noise, where they were found to perform well for speech detection tasks [MC13b].

However, speech frames may have similar MFCC representation to frames comprising highly non-stationary noise as well, since they both have akin spectral attributes. To address this challenge, noise estimation is performed in each frame and the MFCCs in that frame are weighted accordingly [DTC15, MC13b, DM80]. This enables better analysis by separating the background noise from the rest. Next, several consecutive time frames are taken into account. Hence, the nature of transients, which their typical duration is assumed to be of the order of a single time frame, can be exploited.

Formally, consider $\mathbf{a}_n \in \mathbb{R}^C$ as a row vector of C coefficients, consisting of weighted MFCCs, and their first and second derivatives, Δ and $\Delta\Delta$, respectively. These values are extracted from the nth time domain audio frame $\mathbf{f}_n$, introduced in Section 5.2. Let:

$$\underline{\mathbf{a}}_n = [\mathbf{a}_{n-J}, \ldots, \mathbf{a}_n, \ldots, \mathbf{a}_{n+J}] \in \mathbb{R}^{(2J+1)C} \tag{11.4}$$

denote concatenation of feature vectors from $2J + 1$ adjacent frames, where J is the number of past and future time frames. For $J \geq 1$, the elements of $\underline{\mathbf{a}}_n$ in the presence of transients are expected to vary faster than in the presence of speech.

In this study, the number of MFCCs is 8, as commonly used. Thus, $\mathbf{a}_n$ comprises of $C = 24$ coefficients. For practical considerations, we assign a relatively small value of $J = 1$. This allows informative characterization of audio frames based on past-future relations, while consuming low computational load. Thus:

$$\underline{\mathbf{a}}_n = [\mathbf{a}_{n-1}, \mathbf{a}_n, \mathbf{a}_{n+1}] \in \mathbb{R}^{72}. \tag{11.5}$$

Next, standardization is applied on (11.5). Let us assume a set of N observations, while the nth observation is given by (11.5), for $n \in \{1, \ldots, N\}$. For each feature index

$l \in \{1, \ldots, 72\}$, a row vector $\mathbf{O}_l \in \mathbb{R}^N$ is defined as:

$$\mathbf{O}_l = [\underline{\mathbf{a}}_1(l), \ldots, \underline{\mathbf{a}}_N(l)]. \tag{11.6}$$

Then, the mean and standard deviation of $\mathbf{O}_l$ are extracted and termed μ_l and σ_l, respectively. Next, the following vectors are constructed:

$$\boldsymbol{\mu} = [\mu_1, \ldots, \mu_{72}] \ ; \boldsymbol{\sigma} = [\sigma_1, \ldots, \sigma_{72}]. \tag{11.7}$$

Let $\tilde{\underline{\mathbf{a}}}_n(l)$ denote the lth element of the standardized $\underline{\mathbf{a}}_n(l)$, defined as:

$$\tilde{\underline{\mathbf{a}}}_n(l) = \frac{\underline{\mathbf{a}}_n(l) - \mu_l}{\sigma_l}. \tag{11.8}$$

Diffusion Maps

The middle layer of any basic autoencoder architecture can be viewed as a low dimensional representation of its input layer [LTMH13]. Our method exploits this by forcing the middle layer to coincide with the embedded coordinates of $\tilde{\underline{\mathbf{a}}}_n$, generated by the DM method [CL06a]. Thus, the encoder learns to approximate this low-dimensional mapping, while the decoder learns the inverse high-dimensional mapping. DM is a manifold learning approach that is established on the graph Laplacian of the high-dimensional data corpus [CL06b]. DM has been employed well in several signal processing, image processing and machine learning applications [LKC06, FFL10, SSN09, TCG12, DA12, GK13, MC13a, HKC14, CH14].

Let us consider a set of feature vectors $\{\tilde{\underline{\mathbf{a}}}_n\}_n$, constructed according to (11.8). A weighted graph is created with the elements of the set as nodes (or points), where the weight of the edge connecting these nodes is given by the commonly used radial basis function kernel. The scaling parameter of the kernel is set separately for each edge as in [ZMP05]. Practically, merely the 10 nearest neighbors of every point are used to compute the edges. Namely, edges that are not among the nearest neighbors of $\tilde{\underline{\mathbf{a}}}_n$ are nullified.

In order for the embedding and the distribution of the nodes to be independent,

we perform normalization of the data. Therefore, an approximation of the Laplace-Beltrami operator on the data is obtained [CL06a, LKC06]. This operation generates a row-stochastic matrix P which can be viewed as the transition matrix of a Markov chain on the data set $\{\tilde{\underline{\mathbf{a}}}_n\}$. Two sets of bi-orthogonal left and right eigenvectors, $\{\phi_n\}$ and $\{\psi_n\}$, are constructed by employing an Eigenvalue decomposition of P. This process also yields a series of strictly positive eigenvalues $1 = |\lambda_0| \geq |\lambda_1| \geq ... \geq |\lambda_{n-1}| > 0$.

Through informal experiments, we found that for retaining the desired patterns of speech and non-speech frames, it is sufficient that the embedded dimension is set to $d = 3$ (excluding the trivial dimension associated with λ_0). Furthermore, d is small enough to exclude undesired high frequency noise, mostly represented by higher dimensions. The low-dimensional embedding of $\tilde{\underline{\mathbf{a}}}_n$ (11.8) is notated by $\mathbf{m}_n$ and defined as:

$$\mathbf{m}_n = \left(\lambda_1\psi_1\left(\tilde{\underline{\mathbf{a}}}_n\right), \ldots, \lambda_3\psi_3\left(\tilde{\underline{\mathbf{a}}}_n\right)\right). \tag{11.9}$$

Therefore, the set $\{\tilde{\underline{\mathbf{a}}}_n\}$ is embedded into the Euclidean space $\mathbb{R}^3$. In this space, the Euclidean distance is equal to the diffusion distance in the high-dimensional space of $\{\tilde{\underline{\mathbf{a}}}_n\}$.

Our architecture integrates an activation function which maps its input to the interval $[0, 1]$. On the other hand, $\mathbf{m}_n$ often holds values which may exceed this interval. Therefore, this mismatch increases the error measures defined in (11.3). Earlier works have demonstrated that prediction accuracy can be improved by normalizing DM coordinates [Bri90]. We employ these notions to overcome the aforementioned mismatch, by mapping the dynamic range of $\mathbf{m}_n$ to $[0, 1]$. Specifically, the transformation that is employed corresponds to connecting $\mathbf{m}_n$ to $\tilde{\mathbf{m}}_n$ through a softmax layer [GR96], as follows:

$$\tilde{\mathbf{m}}_n(k) = \frac{e^{\mathbf{m}_n(k)}}{\sum_{l=1}^{3} e^{\mathbf{m}_n(l)}}, \tag{11.10}$$

where $1 \leq k \leq 3$. As a result, $0 \leq \tilde{\mathbf{m}}_n(k) \leq 1$ and $\sum_{k=1}^{3} \tilde{\mathbf{m}}_n(k) = 1$.

11.5 Experimental Setting

11.5.1 Notation

Let $\mathbf{s}_j \in \mathbb{R}^L$ denote the noisy audio signal associated with speaker $j \in \{1, \ldots, 11\}$, comprising of L samples. Let $\mathbf{s}_j^i$ denote the union of audio time frames in $\mathbf{s}_j$ that belong to hypothesis $\mathcal{H}^i$. Then, $\underline{\mathbf{s}}^i$ is defined as the concatenation of $\mathbf{s}_j^i$ with respect to all 11 speakers, namely:

$$\underline{\mathbf{s}}^i = \left[\mathbf{s}_1^i, \ldots, \mathbf{s}_{11}^i\right], \tag{11.11}$$

where $i \in \{0, 1\}$.

11.5.2 DED Training Process

Let us consider the two distinct sets $\underline{\mathbf{s}}^0$ and $\underline{\mathbf{s}}^1$. Two training sets, notated $\underline{\mathbf{s}}^0_{\text{tr,ded}}$ and $\underline{\mathbf{s}}^1_{\text{tr,ded}}$, are created by randomly extracting 70% of $\underline{\mathbf{s}}^0$ and $\underline{\mathbf{s}}^1$, respectively. Following Section 11.4.2, the feature vector extracted from the nth frame of $\underline{\mathbf{s}}^i_{\text{tr,ded}}$ is denoted $\underline{\mathbf{a}}^i_{\text{tr,ded},n} \in \mathbb{R}^{72}$. Next, standardization process (11.8) is applied on the latter, which yields $\underline{\tilde{\mathbf{a}}}^i_{\text{tr,ded},n} \in \mathbb{R}^{72}$. This reveals two advantages; first, the network performs a faster learning process. This occurs since standardization implicitly weights all features equally in their representation. Thus, the rate at which the weights connected to the input nodes learn is balanced. This balance allows to rescale the learning rate through the learning process. As a result, the adaptive gradient descent optimization method can be deployed instead of the traditional gradient descent. Second, this approach reduces saturation effects, caused by large values assigned to activation functions. Next, the DM method is applied on the set $\{\underline{\tilde{\mathbf{a}}}^i_{\text{tr,ded},n}\}_n$ separately, for each $i \in \{0, 1\}$, as described in Section 11.4.2. The resulting low-dimensional embedding is clipped to the dynamic range $[0, 1]$ and denoted by $\tilde{\mathbf{m}}^i_{\text{tr,ded},n} \in \mathbb{R}^3$. The proposed architecture entails that while $\underline{\tilde{\mathbf{a}}}^i_{\text{tr,ded,n}}$ is fed to DED^i, the middle layer of the latter is enforced to coincide with $\tilde{\mathbf{m}}^i_{\text{tr,ded},n}$. Let us denote DED^i_{tr} as the ith trained DED.

We integrate the Positive Saturating Linear Transfer (PSLT) activation function,

defined as follows:

$$\sigma(z) = \left\{ \begin{array}{ll} 0, & z \leq 0 \\ z, & 0 \leq z \leq 1 \\ 1, & z \geq 1 \end{array} \right\}. \tag{11.12}$$

The dynamic range that $\sigma(z)$ generates, differently from the known ReLU, suggests maintaining the fluctuations which may appear along the tangled network. Employing $\sigma(z)$ is beneficial in terms of low computational load that is consumed during back propagation, since the derivative of $\sigma(z)$ is simply 1 or 0, neglecting singularities. During back propagation, a nullified derivative will decrease computation time even further, at the expense of updating the weights of the network with less information. Empirically, it was shown not to deteriorate performance. Also, it should be highlighted that complex non-linear patterns can still be learned by the deep architecture. Pre-training is applied on each layer separately in an unsupervised manner, using encoder-decoder structures with 1 epoch and learning rate of 0.1. The optimized weights obtained by this process are considered instead of the random initialization commonly used, which enhances performance since it helps the network to avoid local minima. Pre-training is extremely effective in case there is a relatively small amount of training data, as in our scenario. Next, fine-tuning is applied separately on the encoder and the decoder. Namely, $\underline{\tilde{\mathbf{a}}}^{i}_{\mathrm{tr,ded,n}}$ is encoded into a low-dimensional representation and decoded back to the output layer independently. Subsequently, the two tuned parts are merged and fine-tuning is again utilized, this time on the full stacked DED. Optimization is employed by back propagation through time, which makes use of gradient descent method, parameterized with learning rate of 10^{-5} and momentum of 0.9. Prior to pre-training, the weights are initialized with values drawn from a random normal distribution with zero mean and variance 0.01. Cost function with L_2 weight regularization of 10^{-7}, sparsity regularization of 4 and sparsity proportion of 0.1 is employed. Relatively large sparsity related parameters were assigned, to achieve two goals. First, this allows the networks to avoid over-fitting by effectively ignoring weights with negligible values. Second, it decreases the computational load, since the embedding process involves a

sparse affinity matrix. The network was trained until either 1,000 epochs or minimum gradient value of 10^{-6} were achieved. A typical simulation as such took approximately 10 hours on a i7-7820HQ CPU 64-bit operating system, x64 based processor.

In this study, the architecture was trained using a batch size of 128 observations. As a result, less memory was used compared with feature-by-feature feeding, since fewer registers were employed at the same time. Moreover, the training was accelerated due to less updates performed, i.e., less propagations through the network. On the other hand, batch training may lead to less accurate and stable estimation of the gradient.

11.5.3 Classifier Training Process

Let $\underline{\mathbf{s}}^i_{\text{tr,cl}}$ contain random 15% of observations contained in $\underline{\mathbf{s}}^i$, and $\underline{\mathbf{s}}_{\text{tr,cl}} = \left[\underline{\mathbf{s}}^0_{\text{tr,cl}}, \underline{\mathbf{s}}^1_{\text{tr,cl}}\right]$ to be the full classifier training set. $\underline{\mathbf{s}}_{\text{tr,cl}}$ is built so it is disjoint with the DED training set. Similarly to the DED training process, $\underline{\tilde{\mathbf{a}}}_{\text{tr,cl},n} \in \mathbb{R}^{72}$ and $\tilde{\mathbf{m}}_{\text{tr,cl},n} \in \mathbb{R}^3$ represent feature vectors extracted from the nth frame of $\underline{\mathbf{s}}_{\text{tr,cl}}$, according to (11.8) and (11.10), respectively.

Error measures are defined to distinguish between features that are mapped and reconstructed well and features that are not. Consider two outcomes of propagating $\underline{\tilde{\mathbf{a}}}_{\text{tr,cl},n}$ through DED^i_{tr}. Namely, its low-dimensional predicted representation, denoted by $\tilde{\mathbf{m}}^i_{\text{pr},n}$, and its subsequently predicted reconstruction, denoted by $\underline{\tilde{\mathbf{a}}}^i_{\text{pr},n}$. Consequently, the following error measures are defined, given $\underline{\tilde{\mathbf{a}}}_{\text{tr,cl},n}$:

$$e^i_{\text{en}}(n) \triangleq \|\tilde{\mathbf{m}}_{\text{tr,cl},n} - \tilde{\mathbf{m}}^i_{\text{pr},n}\|_1, \tag{11.13}$$

which is associated with encoder^i, and:

$$e^i_{\text{de}}(n) \triangleq \|\underline{\tilde{\mathbf{a}}}_{\text{tr,cl},n} - \underline{\tilde{\mathbf{a}}}^i_{\text{pr},n}\|_1, \tag{11.14}$$

associated with decoder^i. In both cases, $\|\cdot\|_1$ denotes the ℓ_1 norm.

According to (11.13) and (11.14), it can be inferred that a pair of numerical errors is generated by feeding $\underline{\tilde{\mathbf{a}}}_{\text{tr,cl},n}$ to each trained DED. In this study, the two pairs of errors, associated with DED^0_{tr} and DED^1_{tr}, are interpreted as a coordinate in $\mathbb{R}^4$ and are

represented by $\left(e^0_{\text{en}}(n),\ e^0_{\text{de}}(n), e^1_{\text{en}}(n), e^1_{\text{de}}(n)\right)$. Namely, $\underline{\tilde{\mathbf{a}}}_{\text{tr,cl},n}$ is eventually represented in a four-dimensional coordinate system.

An SVM classifier, notated by $\mathcal{C}$, is applied on the error map, as detailed in Section 11.3.2. In this study, $\mathcal{C}$ is trained to separate coordinates held by $\mathcal{H}^0$ from coordinates held by $\mathcal{H}^1$ (11.2). Thus, two decision regions are created. In this study, both real-time and batch modes are considered, as described in Section 11.5.4. For batch mode, $\mathcal{C}$ is trained on both the encoder and decoder errors projected on the error map, i.e., $\mathcal{C}$ is a three-dimensional hyper plane, embedded in $\mathbb{R}^4$. Real-time mode only exploits the decoder error. Namely, in this case the error map is a two-dimensional coordinate system, and correspondingly $\mathcal{C}$ divides $\mathbb{R}^2$ into two regions.

11.5.4 Testing Process

The DM method requires a batch of both speech and non-speech frames to estimate the low-dimensional embedding. This is impractical for real-time mode where a very small number of frames is available. Therefore, two testing processes are presented; a frame-by-frame testing process in which employment of the DM method is not required, and a batch testing process, which is shown to be more accurate, with substantially higher delay.

Batch Mode Testing Process

In batch mode, both the encoder and the decoder errors are exploited, which increases prediction accuracy. On the other hand, the encoder error is well approximated as long as a large batch of time domain audio data from both hypotheses (11.2) is at hand, which leads to delay in prediction. The test set, notated by $\underline{\mathbf{s}}_{\text{te}}$, is constructed by following similar steps as in the previous section, while ensuring that the intersection of $\underline{\mathbf{s}}_{\text{te}}$ and the training sets of the DED neural network and classifier is empty. $\underline{\mathbf{s}}_{\text{te}}$ includes 15% of both $\underline{\mathbf{s}}^0$ and $\underline{\mathbf{s}}^1$ (11.11). For completion, $\underline{\tilde{\mathbf{a}}}_{\text{te},n} \in \mathbb{R}^{72}$ and $\tilde{\mathbf{m}}_{\text{te},n} \in \mathbb{R}^3$ denote the feature vectors associated with the nth observation of $\underline{\mathbf{s}}_{\text{te}}$, extracted according to (11.8) and (11.10), respectively.

Let $\left(e^i_{\text{en}}(n), e^i_{\text{de}}(n)\right)$ represent the two-dimensional coordinate generated by the propagation of $\underline{\tilde{\mathbf{a}}}_{\text{te},n}$ through DED^i_{tr}. $e^i_{\text{en}}(n)$ and $e^i_{\text{de}}(n)$ are produced according to (11.13)

and (11.14), respectively. For the sake of clarity, we neglect the time index n and address $e^i_{\text{en}}(n)$ and $e^i_{\text{de}}(n)$ as a two-dimensional coordinate $(e^i_{\text{en}}, e^i_{\text{de}})$. As stated earlier, $(e^0_{\text{en}}, e^0_{\text{de}})$ and $(e^1_{\text{en}}, e^1_{\text{de}})$ are concatenated and projected into a four-dimensional error map. Let R_j stand for region j created by the devision $\mathcal{C}$ applied to the error map, where $j \in \{0, 1\}$. Ultimately, the following decision rule is applied by the classifier $\mathcal{C}$ on the input feature vector $\underline{\tilde{\mathbf{a}}}_{\text{te},n}$:

$$\mathcal{C}\{\underline{\tilde{\mathbf{a}}}_{\text{te},n}\} = \begin{Bmatrix} \mathcal{H}^0, & (e^0_{\text{en}}, e^0_{\text{de}}, e^1_{\text{en}}, e^1_{\text{de}}) \in \text{R}_0 \\ \mathcal{H}^1, & (e^0_{\text{en}}, e^0_{\text{de}}, e^1_{\text{en}}, e^1_{\text{de}}) \in \text{R}_1 \end{Bmatrix}. \tag{11.15}$$

Real-Time Mode Testing Process

Since immediate prediction is often required in many audio-based applications, real-time mode is considered as the main branch of this study. Compared with the batch mode, the low-dimensional error is now unavailable. Meaning, the high-dimensional error becomes the single measure to distinguish between audio frames of different hypotheses.

Let $e^i_{\text{de}}(n)$ denote the error produced by propagating $\underline{\tilde{\mathbf{a}}}_{\text{te},n}$ through DED^i_{tr}. In a similar manner to the batch mode, $e^0_{\text{de}}(n)$ and $e^1_{\text{de}}(n)$ are joined and projected into a two-dimensional error map. For sake of clarity, we again address these two measures as $(e^0_{\text{de}}, e^1_{\text{de}})$. Let R_j stand for region j created by the devision $\mathcal{C}$ applied to the two-dimensional error map, where $j \in \{0, 1\}$. As a result, the following decision rule is considered by the classifier $\mathcal{C}$, regarding input feature vector $\underline{\tilde{\mathbf{a}}}_{\text{te},n}$:

$$\mathcal{C}\{\underline{\tilde{\mathbf{a}}}_{\text{te},n}\} = \begin{Bmatrix} \mathcal{H}^0, & (e^0_{\text{de}}, e^1_{\text{de}}) \in \text{R}_0 \\ \mathcal{H}^1, & (e^0_{\text{de}}, e^1_{\text{de}}) \in \text{R}_1 \end{Bmatrix}. \tag{11.16}$$

11.6 Experimental Results

In each of the experiments described in this section, comparisons are made between our proposed approach and several competing voice-activity detectors. In order to

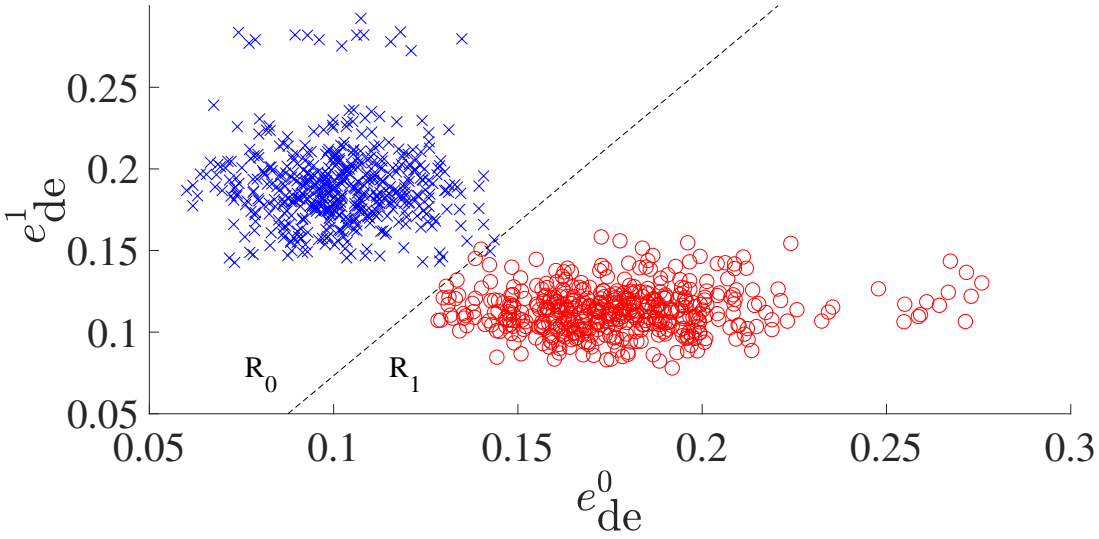

Figure 11.2: Two-dimensional error map, generated by the real-time mode. Red circles denote speech presence and blue 'x' marks denote speech absence. The trained linear SVM classifier is represented by the dashed line and the decision regions it generates are notated by R_0 and R_1.

avoid skewness and unfair imbalance, performances were generated by using identical experimental conditions. Specifically, the same test set, acoustic setup and optimization measure, i.e., TN + TP (true positive + true negative), are uniformly employed. To allow appropriate assessment of performances, two measures are used: The optimized TP+TN measure, and the relation between TP and TN measures.

11.6.1 Performance of Proposed Approach

Accuracy

Primarily, the proposed method is applied using 100% of the DED training data set in a batch mode, as detailed in Section 11.5.4. The accuracy rate is 99.1%. In this mode, voice activity is detected by using both low and high-dimensional numerical measures. This performance gives rise to the main assumption of this research. Namely, that speech can be distinguished from transients based on their underlying geometric structures. Real-time voice-activity detection is performed according to Section 11.5.4. In this mode, the accuracy rate reaches up to 98.1% when 100% of the DED training data set is used. Visualization of the error map is given in Fig. 11.2. It should be highlighted that similar visualization is not given for the batch mode, since the corre-

	B. 10 K.	M. 10 H.	C. 5 H.	M. 0 K.	B. 15 S.	std
Tamura	73.6	83.8	83.9	73.8	81.2	5.2
Dov - Audio	87.7	89.9	87.8	86.5	90.2	1.6
Dov - Video	89.6	89.6	89.6	89.6	89.6	0
Dov - AV	92.9	94.5	92.8	92.9	94.6	0.9
Ariav - AV	95.8	95.4	95.9	95.1	97.2	0.8
Proposed Real Time	**98.4**	**98.3**	**98.3**	**98.3**	**98.5**	**0.1**
Proposed Batch	**99.3**	**99.6**	**99.3**	**99.3**	**99.5**	**0.1**

Table 11.1: Comparison between voice-activity detection methods in terms of accuracy rate, with respect to the TP+TN measure.

sponding error map lays in $\mathbb{R}^4$. These results reflect on the strong relation between low and high-dimensional information. Namely, even though low-dimensional measures are not integrated into the decision rule, the separation in the diffusion space is implicitly expressed through the inverse mapping of the decoder. Therefore, the reconstructed high-dimensional information in the feature space is a sufficient measure to tell apart speech from non-speech frames. By examining the results, high robustness can be concluded. Namely, despite the variety of stationary and non-stationary noises included in the database, the intrinsic structure of speech is still well detected.

Generalization

Generalization and sensitivity of the proposed method are analyzed by performing an additional experiment in the real-time mode. These properties are examined with respect to two parameters; the corpus size of the DED training set and the ratio of speech observations in the latter. In this experiment, 5 different fractions of the full amount of the DED training set are considered. For each fraction, 5 different ratios between speech and non-speech observations are inspected. Results are demonstrated in Fig. 11.3. It can be observed that the accuracy rate surpasses 95%, even when merely 50% of the training data is available, which projects on the low sensitivity of the proposed algorithm to this measure. Also, the maximal accuracy is achieved when the speech

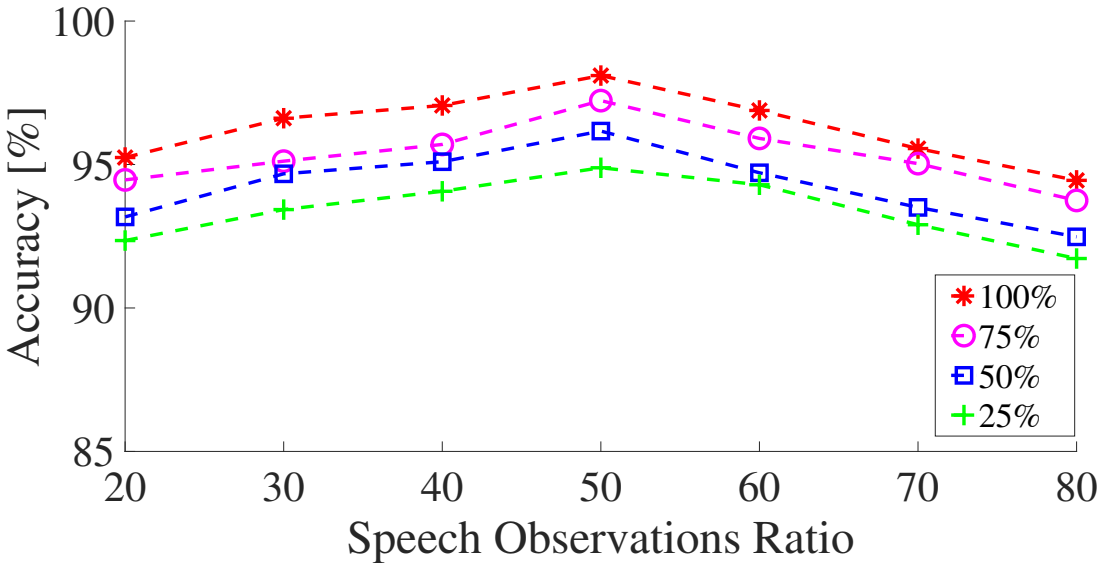

Figure 11.3: Accuracy rate percentage (TP+TN) of the proposed method using the real-time mode. Different fractions of the full DED training set (25,50,75,100[%]) are considered along a grid of speech observations ratios.

observations ratio is equal to 50%, i.e., when there is an equal amount of speech and non-speech observations in the DED training set. This optimal ratio allows the network to learn two separate manifolds with minimal bias. This bias, if exists, can come to surface during testing, when one mapping is more robust than the other. In this case, relying on Euclidean distance between manifolds as done in this research may be harmful for classification. It can also be inferred that the performance has low sensitivity to changes in the speech observations ratio parameter. For example, let us consider the results achieved by exploiting 100% of the training corpus. Then, speech observations ratios of 20%, 50% and 80% yield accuracies of 95.2%, 98.1% and 94.4%, respectively. It is interesting to note that the degradation in performance is not symmetric around the ratio of 50%. i.e., degradation is more noticeable when the amount of noise observations is lower than those of speech in the training process. This can be related to the high varying nature of non-stationary noises in comparison to speech. Meaning, larger corpus of transients is needed to construct a robust low-dimensional structure with the DM method.

As mentioned in Section 11.4.1, the constructed database comprises 42 different combinations of stationary and non-stationary noises. Thus, a fundamental question is concerned with the ability of the proposed detection system, and specifically DED^0, to generalize well to other types of noise. In order to increase the generalization ability

of the suggested detector to noises of various kinds, we performed several actions that regard both the architecture of the system and the feature extraction process. The way the architecture is built puts emphasis on both the difference between speech and noise, and on the similarity of noise to previously trained noises. As a result, the decision mechanism of the system relies on a combination of two learning systems. The features that are extracted from the time domain are constructed to exploit this form of architecture. During training, not only temporal and spectral features are derived, as traditionally done in state-of-the-art methods, but also the informative spatial diffusion map features. This reveals the unique intrinsic geometric structure of speech utterances. Ultimately, when feeding the system with unseen noise, its intrinsic structure is evaluated by the system and compared against speech and non-speech frames separately. Therefore, the performance of the system is not sensitive to unseen noises, in comparison to competing methods, as shown through the experimental setup detailed earlier in this section.

11.6.2 Comparison to Competing Methods

In order to assert the performance of our architecture in a global scale, it is compared to 5 voice-activity detectors. The competing methods are presented in [DTC15, ADC18a, TIH$^+$10] and are denoted "Ariav", "Dov" and "Tamura", respectively. Table 11.1 presents the performance of each method in 5 different acoustic environments that compose of transients (keyboard, hammering, scissors) and stationary noises (babble, musical, colored Gaussian noise) with different SNR values (0, 5, 10, 15 [dB]). The explicit abbreviations used in the table are as follows; "B. 10 K." is babble noise with 10 dB SNR and keyboard tapping, "M. 10 H." is musical noise with 10 dB SNR and hammering, "C. 5 H." is colored noise with 5 dB SNR and hammering, "M. 0 K." is musical noise with 0 dB SNR and keyboard tapping, and "B. 15 S." is babble noise with 15 dB SNR and scissors. The real-time and batch modes are notated by 'Proposed Real-Time' and 'Proposed Batch', respectively.

It can be observed that the proposed algorithm, even in real-time mode, achieves the best accuracy rate through all varied setups. It should be highlighted that the proposed solution exploits only audio signals, while competing methods rely on integration

of both audio and video data.

By observing the std measure in Table 11.1 it is shown that, unlike competing methods, the performance of the proposed approach is barely affected by the change in the acoustic environments. This high robustness can be related to the construction of intrinsic representations of the audio frames. These representations do not consider the contents of transients or background noises, but merely their intrinsic geometric patterns. These patterns are unique for speech and non-speech audio frames, which allows enhanced performance regardless of the setup. The results presented in Table 11.1 show slight improvement in comparison to the results presented in Section 11.6.1. While in the former, 5 specific setups are inspected, 37 additional setups are considered in the latter. This indicates the existence of specific combinations of speech, stationary and non-stationary noises that are harder to comprehend. Deeper analysis of this phenomenon should be addressed in future work.

To allow further evaluation, we employ the receiver operating characteristic (ROC) curve. Three acoustic setups presented in Table 11.1 are considered in Figs. 11.4 – 11.6. In each ROC curve, the real time and batch proposed approaches are compared against four competing voice-activity detectors. Since the test set is identical and balanced across all methods, a constructive comparison is made by the ROC curves. The latter allows analysis of the relation between TP and TN, thus delivering information about the trade-off between the two. It is worth noting that TN can be derived from the false positive (FP) measure, held by the x axis, by simply applying the relation TN = 1-FP. It can be observed that our voice-activity detector outperforms the competing methods in a wide range of operating points.

11.6.3 Performance Analysis

This study presents a voice-activity detection method that reaches substantially higher accuracy results in comparison to other state-of-the-art methods. This improvement can be attributed to several novelties, where two of them are considered the most influential. First, the integration of the DM method, forced at the end of the encoder. Second, construction of two separate DEDs, one trained with speech presence observations and the second with speech absence observations. This section is divided into two

main parts. Initially, the differences between two competing methods and the proposed approach are analyzed and theoretical explanations of the gap in accuracies are given. Then, two experiments are conducted to establish these explanations.

First, the method proposed in [DTC15] is considered. In this method, low-dimensional embedding is built with the DM method, as done in our study. This embedding is constructed by considering joint relations between speech and non-speech features. However, our approach employs the DM method by considering relations between features of the same hypothesis only. In order to evaluate the influence of this difference on the degradation in performance, the algorithm proposed in [DTC15] has been implemented. Consequently, high overlap of speech and non-speech embeddings in the diffusion space has been observed. This method performs voice-activity detection mainly by modeling two low-dimensional Gaussian mixture models. Meaning, this approach aims to separate speech from non-speech coordinates by constructing a separator from a sum of weighted exponential kernels. As a result, overlapped coordinates are highly at risk to be misclassified.

Next, the method proposed in [ADC18a] is analyzed. In this approach, a single auto-encoder attempts to learn the low-dimensional embedding of both speech and non-speech frames. As a result, joint embedding is shown to lead to high overlap in the low-dimension, much like in the research conducted in [DTC15]. Additionally, this architecture does not consider the DM method as a constraint on the embedded data, so dimensionality reduction is done automatically. This leads to a lack of spatial information in the low-dimension and absence of geometric insight. Ultimately, this causes significant overlap between low-dimensional representations and to deterioration in performance. The high accuracy shown in [ADC18a] can be related to high exploitation of temporal relations, carried by the RNN, and integration of visual features in the classification process. To explore the performance of the network without video, the authors of this work implemented audio-only version of the method presented in [ADC18a]. The outcome shows severe degradation in performance, as the average accuracy is 83% with respect to all 5 setups considered in Table 11.1.

Two experiments are conducted in order to validate the above notions. First, the algorithms proposed in [DTC15] and [ADC18a] are implemented with merely audio

data, as demonstrated in Fig. 11.7. Accuracy rates of these methods are calculated by employing different fractions of the full DED training set. For this particular experiment, the ratio of speech observations was fixed to 50%, to achieve optimal results. Several interesting insights can be obtained based on these outcomes. Primarily, there is a substantial gap between performances when considering only the audio data and neglecting visual features. Moreover, it is noticeable that the method proposed in [ADC18a] is not affected as much by the change in the amount of training observations. As previously stated, the latter does not consider any geometric or structural constraint on the embedded data. Therefore, as long as the training observations are divided roughly equal between hypotheses, their amount has lower significance. On the other hand, the study presented in [DTC15] highly relies on the intrinsic structure of the data. i.e., the more training observations are available, the better the joint relations between speech and non-speech features are modeled. In this case, larger training set leads to a more robust manifold construction.

In order to further explore the core of the advantages of the proposed approach, another experiment is conducted. This time, the studies in [DTC15] and [ADC18a] are implemented by integration of several principles of this study. It should be noted that the detection algorithm presented in each of these studies remains the same. In [DTC15], the algorithm was altered such that the low-dimensional coordinates are learned separately for speech and non-speech frames before applying the Gaussian mixture model on the generated manifolds. In [ADC18a], two separate auto-encoders were implemented. Each auto-encoder learned the low-dimensional mapping of speech and non-speech audio frames independently. Also, the DM method was applied in a similar manner to the proposed method in order to integrate spatial information. The output of each encoder was inserted into a separate RNN. The output of each RNN represents the probability that a test observation is taken from a speech audio frame. Ultimately, the probabilities of the two RNNs are intersected and a prediction is made by a constructed decision rule.

The results of this experiment are given in Fig. 11.8. For each method, the accuracy is calculated along a grid of fractions of the full DED training set, while the speech observations ratio is once again set to 50%. Moreover, the performance of each method

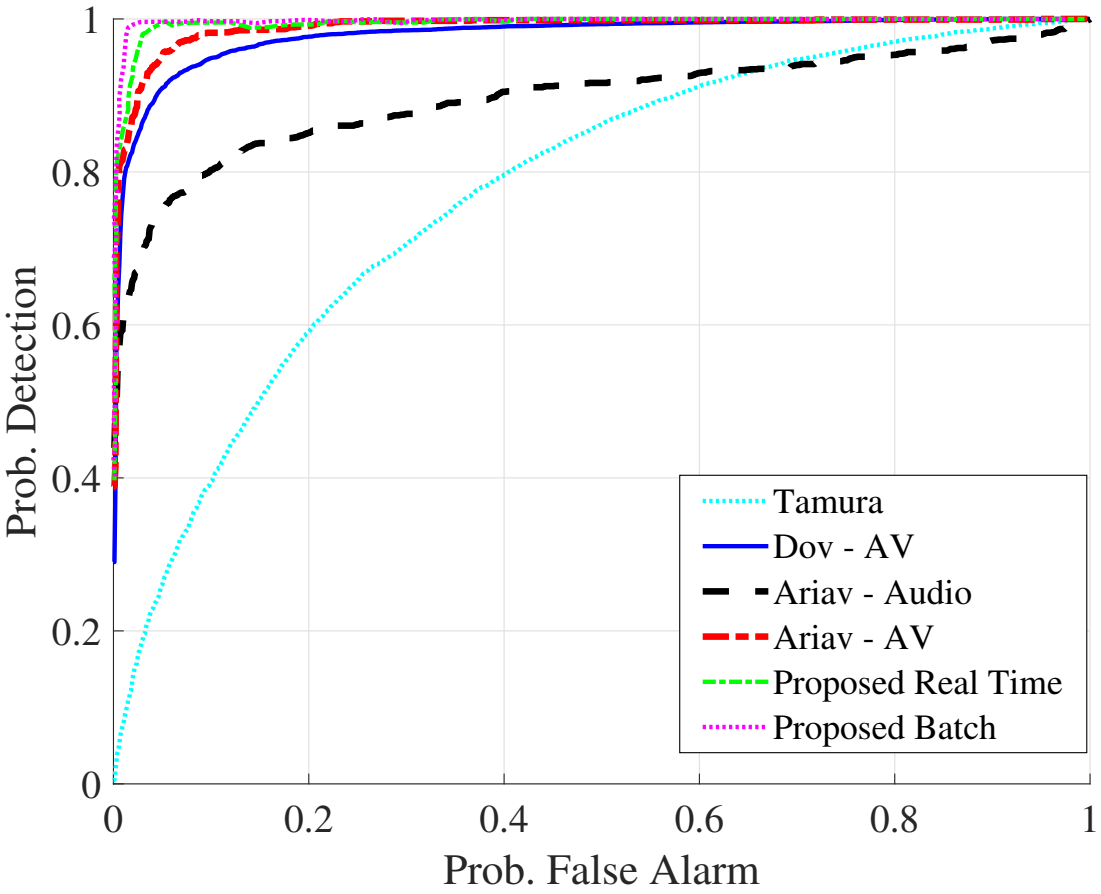

Figure 11.4: Probability of detection versus probability of false alarm in an acoustic environment of babble noise with 10 dB SNR and keyboard transient interferences.

is given once with its original implementation and once with the improved implementation that combines principles from our method. Regarding the studies presented in [DTC15, ADC18a], the accuracies of the two new implementations significantly improve. Also, these models are less sensitive to changes in the size of the DED training corpus.

Even though an increase in performance can be observed, the studies presented in [DTC15] and [ADC18a] still do not reach the results of the proposed method. The core classification algorithm of each of the three discussed methods remains unchanged through all the comparative experiments conducted in this study. Therefore, the core classification algorithm proposed in our study may be responsible for the observed gap.

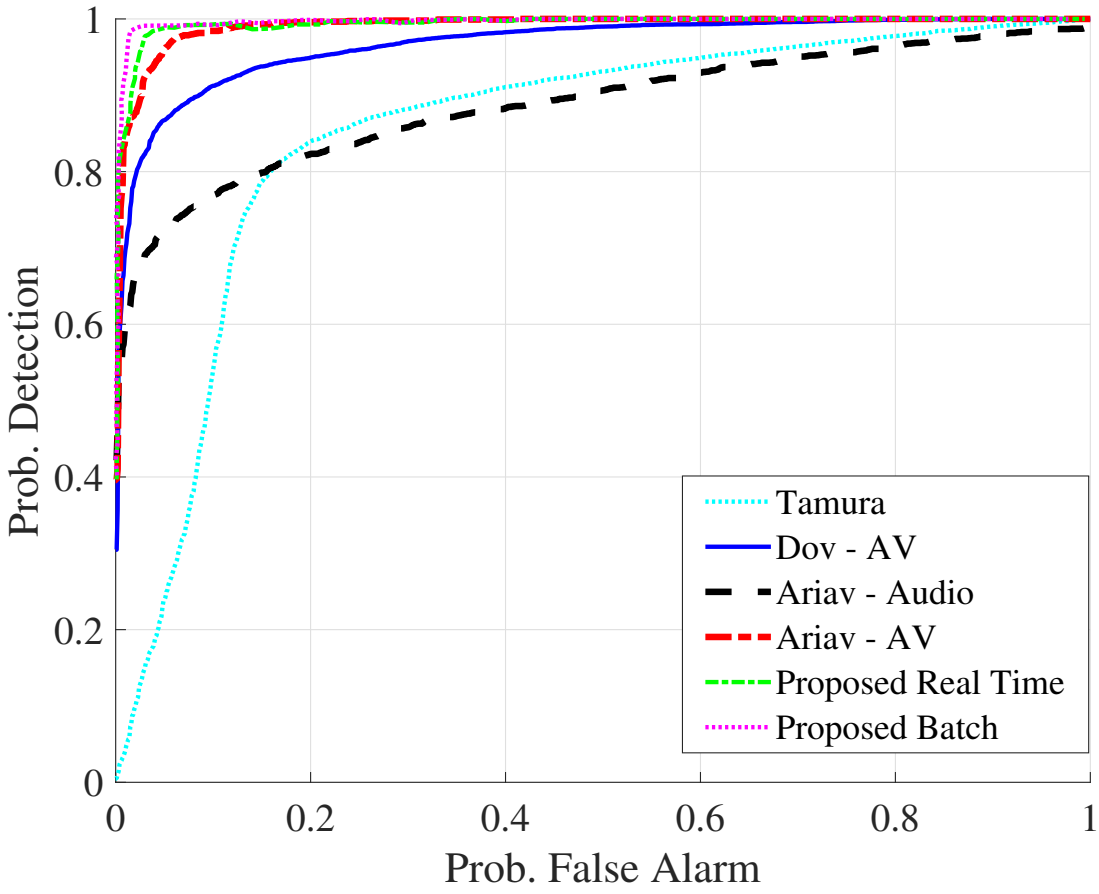

Figure 11.5: Probability of detection versus probability of false alarm in an acoustic environment of colored noise with 5 dB SNR and hammering transient interferences.

11.7 Conclusions

In this work we have performed voice-activity detection with audio-based features. We separately represented the low-dimensional geometric structures of speech and non-speech frames by integrating the diffusion maps method with two independent, encoder-decoder based, deep neural networks. This separation of speech from stationary noises and transients during the training process of the two networks also led to high robustness and generalization abilities, as well as low sensitivity to the amount of available training data. The proposed method has shown state-of-the-art results in a real time mode, and can be integrated into dedicated communication systems. Nonetheless, non stationary noises are still the main cause of false detection in this research, due to their high varying nature. This challenge may be addressed by employment of more distinctive geometric features as well as assimilation of joint constraints between the

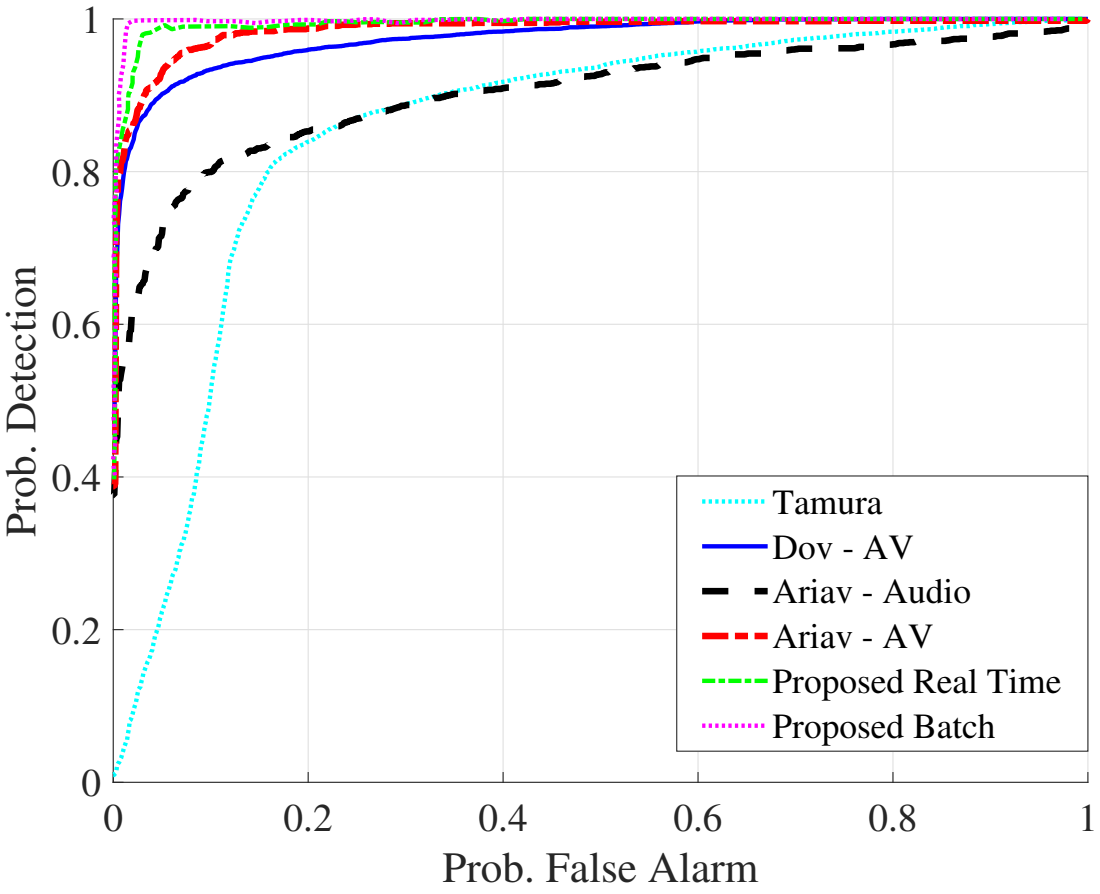

Figure 11.6: Probability of detection versus probability of false alarm in an acoustic environment of musical noise with 10 dB SNR and hammering transient interferences.

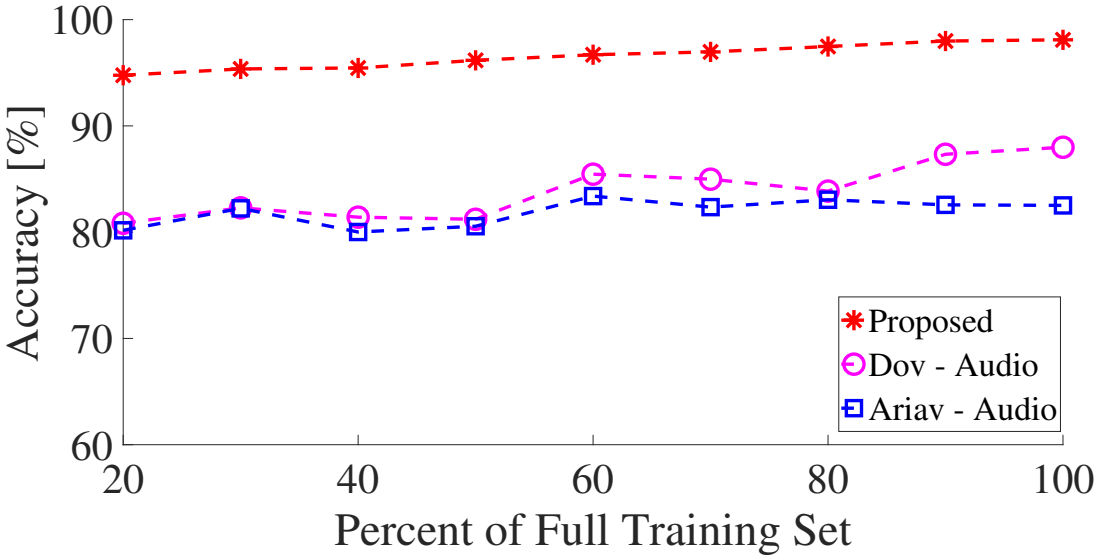

Figure 11.7: Accuracy rate percentage (TP+TN) of the proposed method using the real-time mode. Performance is presented along a grid of different fractions of the full DED training set, while the speech observations ratio is fixed to 50%.

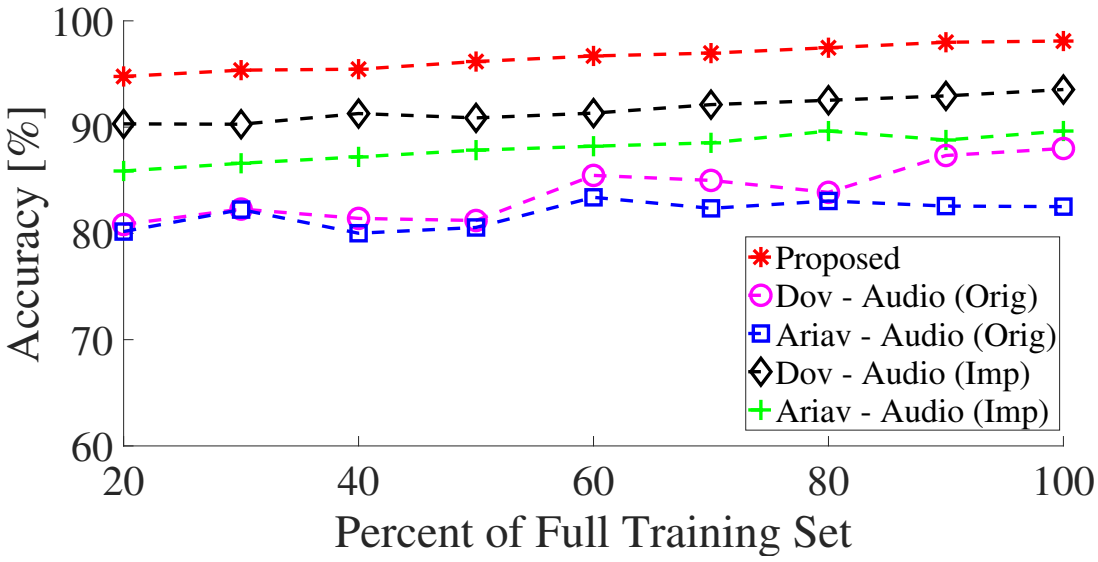

Figure 11.8: Accuracy rate percentage (TP+TN) of competing methods, along with the performance of the proposed algorithm in the real-time mode. Accuracy is presented along a grid of different fractions of the full DED training set, while the speech observations ratio is fixed to 50%. Each of the two competing methods is implemented once in the original form (marked 'Orig') and once with integration of concepts from the proposed method (marked 'Imp').

encoder and decoder. It would be instructive to further factorize the proposed approach and analyze the improvement. Moreover, a heuristic explanation regarding the relation between diffusion maps and the presented method can be meaningful for further understanding. One hypothesis, for instance, links between transition in time and on the transition map. Another theory suggests that the corresponding Markov chain is a sequence of phonemes, and the diffusion rate in the diffusion map corresponds to the velocity of phonemes pronunciation. Additionally, the performance of the proposed detection method in reverberant and noisy acoustic environments with signal-to-noise ratios lower than 0 dB, should be explored.

Chapter 12

Evaluation of Deep-learning-based Voice-Activity Detectors and Room Impulse Response Models in Reverberant Environments

12.1 Introduction

VAD aims to determine the boundaries in which speech exists in an observed audio signal. State-of-the-art deep-learning-based VADs are often trained with anechoic data. However, real-life acoustic environments are reverberant, which deteriorates VAD performance in practical scenarios. In this study, we mitigate the mismatch between training data and real data by generating an augmented training set that integrates anechoic and reverberant audio signals. The reverberant training corpus is generated by convolving anechoic utterances with a simulated RIRs. Enhanced VAD in reverberant environments may benefit a variety of audio-based applications such as speech enhancement [KDG+16, KDJJ16, ZWMZ16], dereverberation [HWW+15, SK15] and speech and speaker recognition [KPP+17, GSDY15].

Deep-learning-based VADs have attained leading performances during recent years, due to the ability of neural networks to learn non-linear relations and complex patterns of audio signals. To detect voice activity, Ariav and Cohen [ADC18b] encoded spectral audio features via an auto-encoder that fed a recurrent neural network. Wagner et al. [WSSA18] introduced automatic feature engineering through the convolutional layers of a deep neural network. Leading performance was obtained by Kim and Hahn [KH18] that integrated an attention model to weight context information into existing deep learning architectures. Combined end-to-end VAD system was introduced by Ariav et al. [AC19], that comprised of WaveNet for feature extraction and a deep residual network for speech detection. Ivry et al. [IBC19] applied ensemble learning with two deep encoder-decoder structures to learn the unique temporal and spatial patterns of speech through the diffusion maps method.

In latest decades, several RIR models were proposed to produce reverberant utterances via simulations. An extension of the known image method [AB79] to arbitrary polyhedra was first introduced by Borish [Bor84]. Vorländer [Vor89] suggested a combined modeling that considers both the image method and ray-tracing techniques. Rindel [Rin93] employed reflection coefficients that are incidence angle-dependent in the frequency domain, to offer a more accurate characterization of a room response. A similar model was implemented by Lam [Lam05], but it focused on low frequencies for more realistic boundary conditions. Valeaua et al. [VPH06] applied the diffusion equation to predict room acoustics.

We consider the aforementioned five deep-learning-based VADs [ADC18b, WSSA18, KH18, AC19, IBC19] and five RIR models [Bor84, Vor89, Rin93, Lam05, VPH06]. First, we show that training these detectors with solely anechoic corpus and testing them in real reverberant rooms and spaces leads to a significantly impeded detection capability. To include unique acoustic patterns of reverberant data during training, we generated an augmented training set of nearly five million utterances. This extended corpus comprises of anechoic and reverberant signals, where the latter is generated by convolving the anechoic signals with a variety of RIRs, generated using a fixed RIR model. Then, all five VADs are independently trained with this augmented training set. This experiment is repeated for each of the five RIR models. All trained detection systems

are tested in three real reverberant spaces of a classroom, a large concert hall, and an octagon shaped library. Experimental results demonstrate that the performance of all detectors is enhanced in each of the tested reverberant environments, regardless of the RIR model employed during training. Evaluation measures such as accuracy, precision and recall increase by 20% on average, compared to non-reverberant training. An interesting outcome shows that the leading accuracy of each detector was consistently achieved by the Valeaua RIR model [VPH06]. In a similar manner, the detector introduced in Ivry [IBC19] prevailed competing VADs across all experiments.

The remainder of this chapter is organized as follows. In Section 12.2, we describe the database generation. In Section 12.3, we introduce the experimental results. Finally, in Section 12.4, we draw conclusions.

12.2 Database Generation

In this section, we detail the construction of two disjoint datasets: An augmented training set and a test set. The training set contains both anechoic and reverberant utterances, that are generated by simulating a fixed RIR model and convolving the anechoic data with it. In contrast, the test set is constructed with real reverberant conditions, not simulations.

For the training stage, we employ the TIMIT [GLF$^+$88] training dataset that contains 4620 anechoic utterances, sampled at 16 kHz. Since this corpus is imbalanced and does not comprise of noises, we perform several preprocessing steps. Initially, since in TIMIT there are more speech frames than silence frames, we manually add 2 s of silence for each existing recording in the corpus. Next, we acquire recordings of stationary noises such as white and colored Gaussian noise, musical instruments and babble. These noises are randomly added to both speech and silence frames in SNRs that are distributed uniformly between $[10, 20]$ dB relative to clean anechoic speech.

We perform augmentation of this anechoic training set, so it holds both anechoic and simulated reverberant data. To simulate varied reverberant environments, 50 rectangular spaces are considered, such that the length, width and height are uniformly chosen from the range $3 - 20$ m. This permits both small, medium and large spaces.

To cover various scenarios, each of the 50 spaces is simulated 20 times, with different locations of the speaker and the receiver. To obtain a realistic setting, the speaker and the microphone are limited to height range of $1 - 2$ m, and a distance of at least 0.5 m from each other. Each room is simulated with a reverberation time (RT60) that is chosen uniformly from the interval $0.1 - 1$ s, such that both low and high reflective surfaces are accounted for.

Given an RIR model, we simulate 50×20 RIR signals. Each of these responses is convolved with the anechoic utterances in [GLF+88], which results in a reverberant training set. The augmented training set is simply a composition of the original anechoic signals with their aforementioned reverberant modifications. Ultimately, for a given RIR model, the training set comprises of 4620×1001 utterances.

In the test stage, we use 100 anechoic utterances from the TIMIT test dataset. To obtain the reverberant test set, convolution is applied between this corpus and real recordings of room responses. These RIRs are taken from three reverberant environments [SS10] of a classroom, a large concert hall and an octagon shaped library. For each environment, 130 recordings are available, from various locations in the room. Thus, three test sets are formed, each comprises of 100×130 reverberant utterances.

12.3 Experimental Results

In the following experiments, voice-activity detection performance is evaluated by several measures. The ROC curve is used to present a trade-off between speech detection and false-alarm rates in various operation points. The robustness of the VAD and the sensitivity of its classifier to noises is derived by the AUC measure. Accuracy, precision, recall and F1-score [Pow20] are also employed in this study. When combined, all measures strongly indicate on the accuracy, generalization and robustness abilities of the detector.

In this study, we consider five VADs [ADC18b, WSSA18, KH18, AC19, IBC19] and address them as Ariav-R, Wagner, Kim, Ariav-W and Ivry, respectively. Also, we employ five RIR models [Bor84, Vor89, Rin93, Lam05, VPH06], and refer them as Borish, Vorlnder, Randel, Lam and Valeaua, correspondingly. We perform the following

experiment, comprises of two-stages; training stage and test stage. In the first part, a fixed RIR model is simulated. Then, the steps described in Section 12.2 are implemented with respect to the chosen RIR model. As a result, an augmented training set is obtained. Next, a VAD system is chosen and trained with the derived training set. We repeat this experiment for each VAD system and for each RIR model. Ultimately, this stage yields 5×5 trained VAD systems. In the second stage, we test each trained detector on three test sets, generated in three reverberant environments of a classroom, a large concert hall and an octagon shaped library, as detailed in Section 12.2. An experiment conducted by the authors of this study showed that these three acoustic spaces are characterized by long (1 s), medium (0.8 s) and short (0.6 s) reverberation time, in correspondence.

By observing Figures 12.1–12.5, several conclusions can be derived. First and foremost, if the training set contains merely anechoic data, then the performance of all VADs is significantly degraded when tested in real reverberant conditions. Respectively, employing the suggested augmented training set that comprises of reverberant utterances consistently enhances VAD performance in practical scenarios. The reason is that acoustic patterns and features highly differ between reverberant and anechoic environments, and this mismatch between the training data and real data is mitigated by the reverberant augmented training set. Another interesting derivation is that the RIR model introduced by Valeaua [VPH06] consistently leads to the highest performance, relative to competing RIR models, for all VADs and in all tested acoustics. One explanation is that the model proposed in [VPH06] predicts room acoustics better than the remaining models. It should be noticed that training with Valeaua RIR model leads to rapid convergence of the ROC curves and leading AUC values. These results indicate that detectors trained with Valeaua impulse response achieve wide margins of separation between speech and silence. Therefore, these detectors experience high robustness from noises and interferences that might shift the classifier.

Further derivations can be made based on Figures 12.6–12.8. The reported results reaffirm that augmentation of the training set with respect to Valeaua RIR model leads to enhanced VAD performance in reverberant conditions, compared to training that merely considers anechoic data. This enhancement can be quantified by approximately

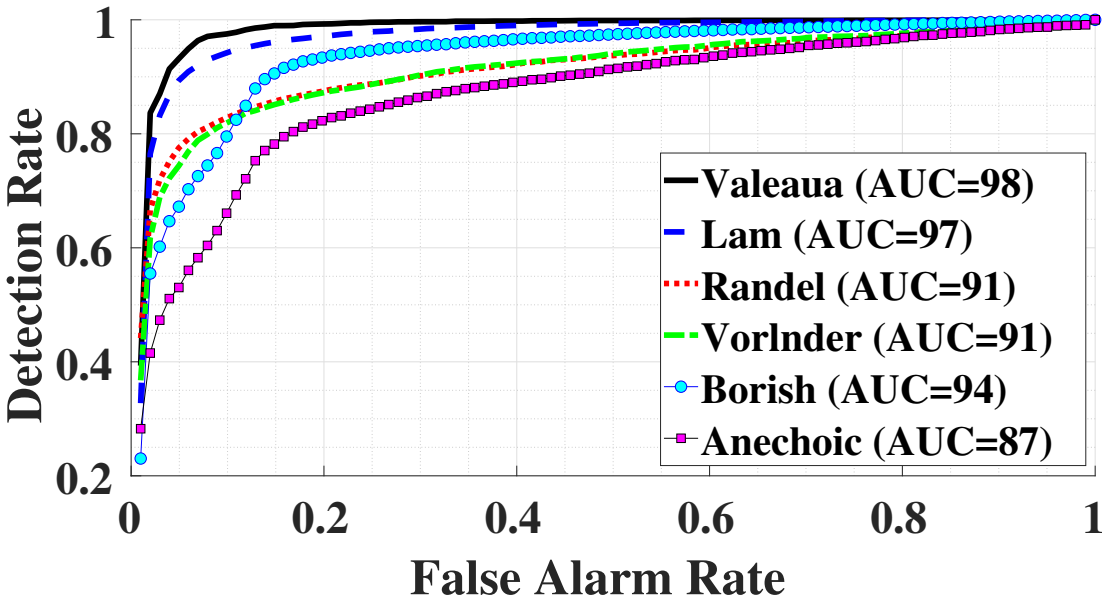

Figure 12.1: Detection rate versus false alarm rate in a reverberant setup of a classroom. Comparison is made between the five different training RIR models with VAD by Ivry.

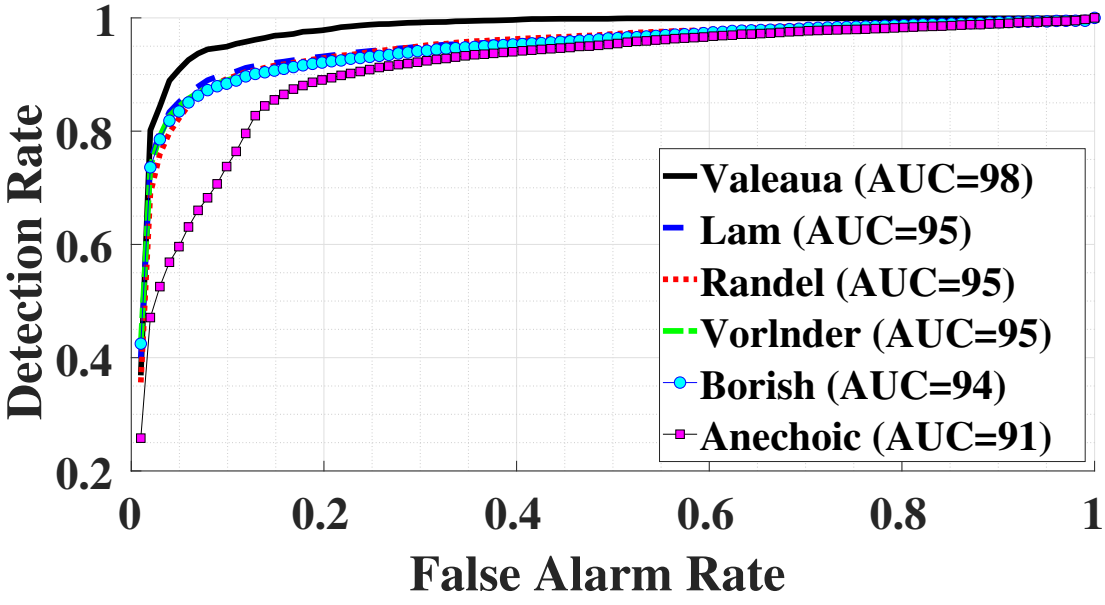

Figure 12.2: Detection rate versus false alarm rate in a reverberant setup of a classroom. Comparison is made between the five different training RIR models with VAD by Ariav-W.

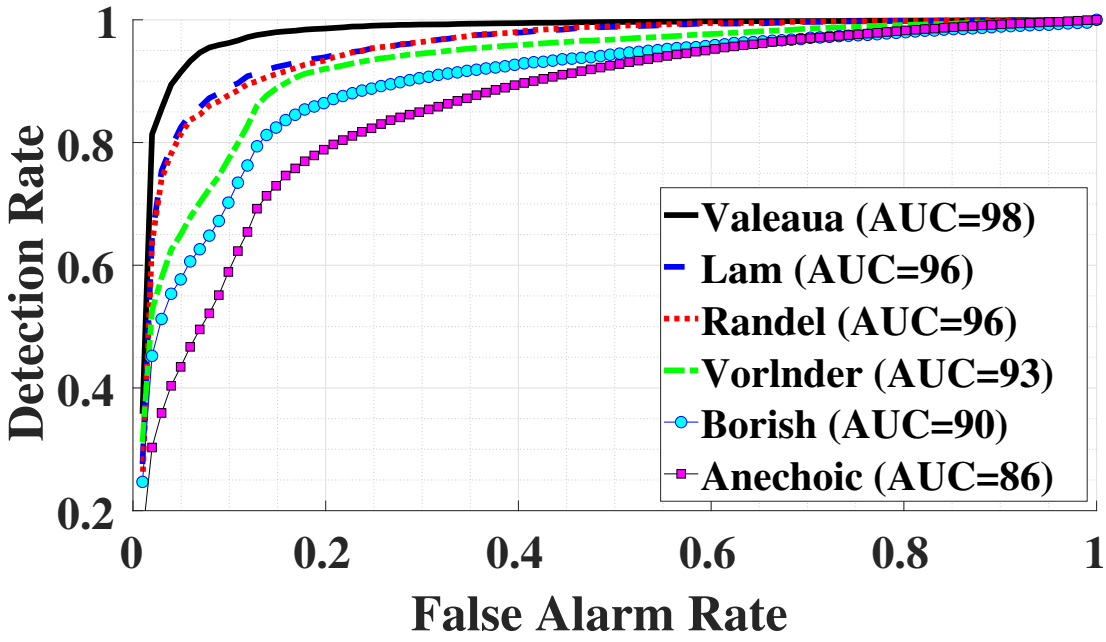

Figure 12.3: Detection rate versus false alarm rate in a reverberant setup of a classroom. Comparison is made between the five different training RIR models with VAD by Kim.

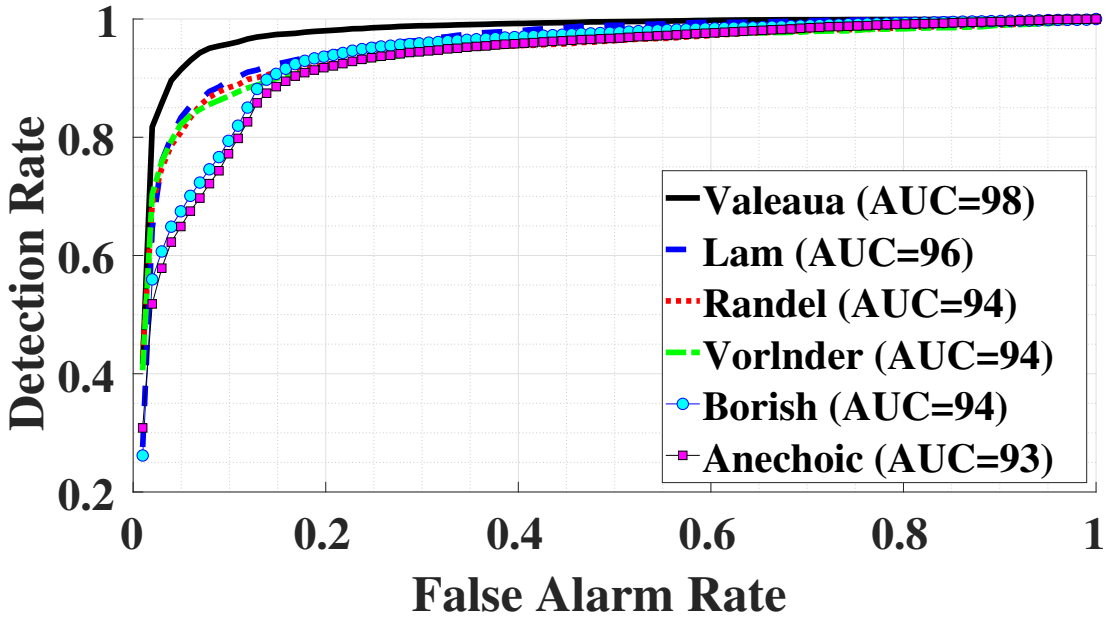

Figure 12.4: Detection rate versus false alarm rate in a reverberant setup of a classroom. Comparison is made between the five different training RIR models with VAD by Wagner.

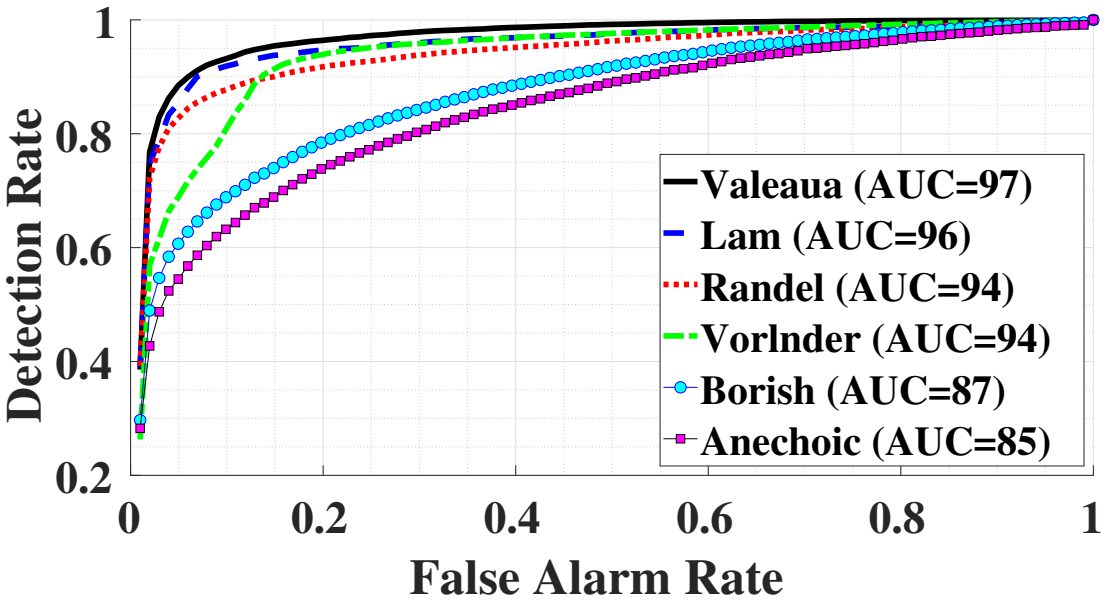

Figure 12.5: Detection rate versus false alarm rate in a reverberant setup of a classroom. Comparison is made between the five different training RIR models with VAD by Ariav-R.

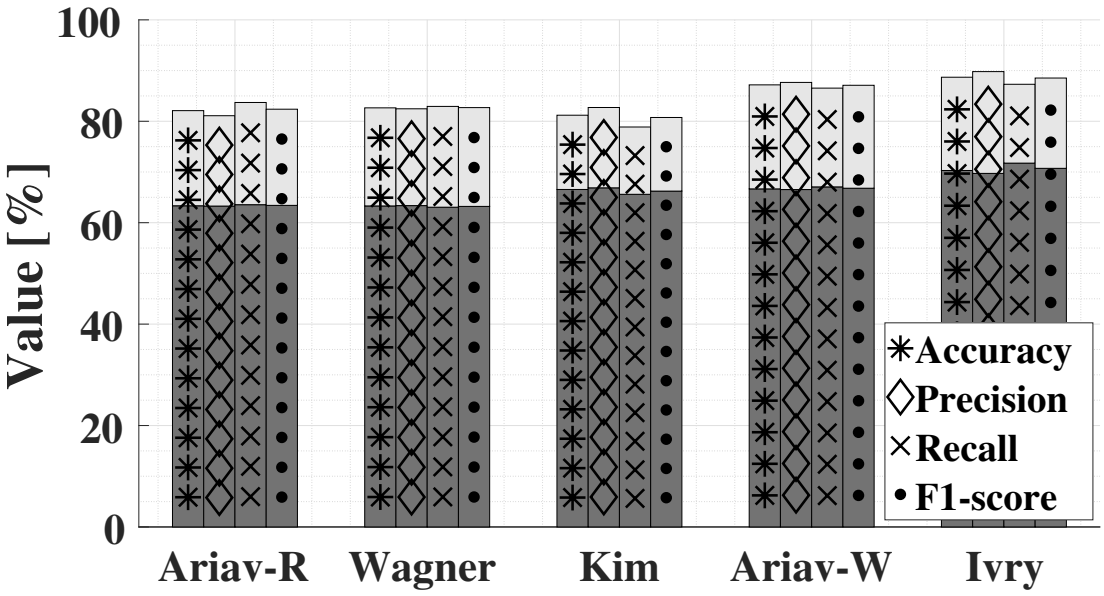

Figure 12.6: Performance of the five VADs in real reverberant conditions of classroom. Comparison is made between employing anechoic training (dark) and augmented training with Valeaua RIR model (light).

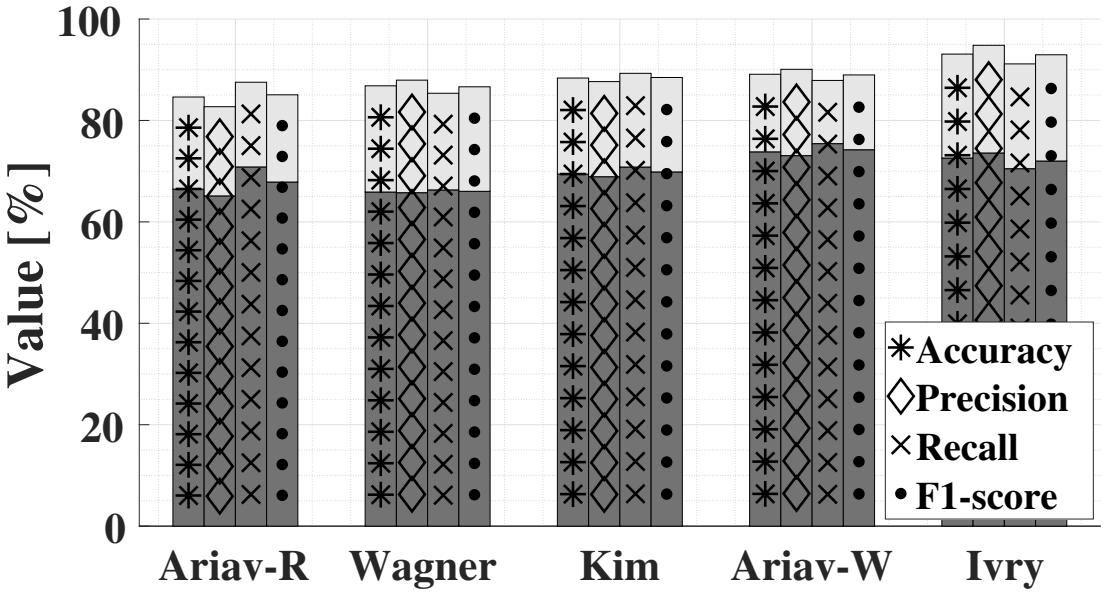

Figure 12.7: Performance of the five VADs in real reverberant conditions of large concert hall. Comparison is made between employing anechoic training (dark) and augmented training with Valeaua RIR model (light).

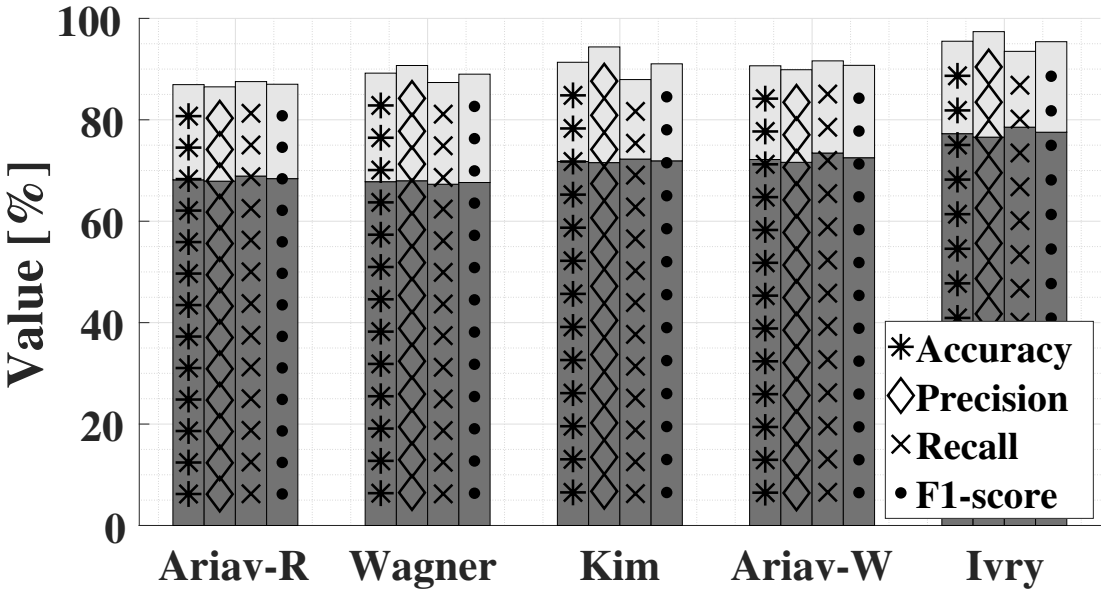

Figure 12.8: Performance of the five VADs in real reverberant conditions of octagon library. Comparison is made between employing anechoic training (dark) and augmented training with Valeaua RIR model (light).

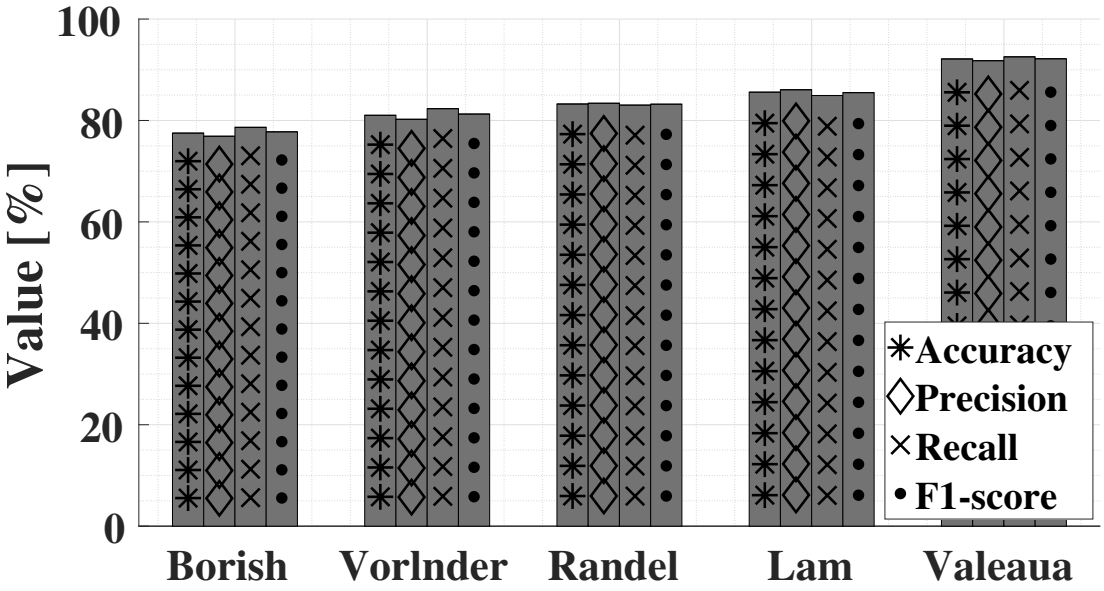

Figure 12.9: Performance of Ivry VAD in real reverberant conditions of classroom. Comparison is made between the five RIR training models.

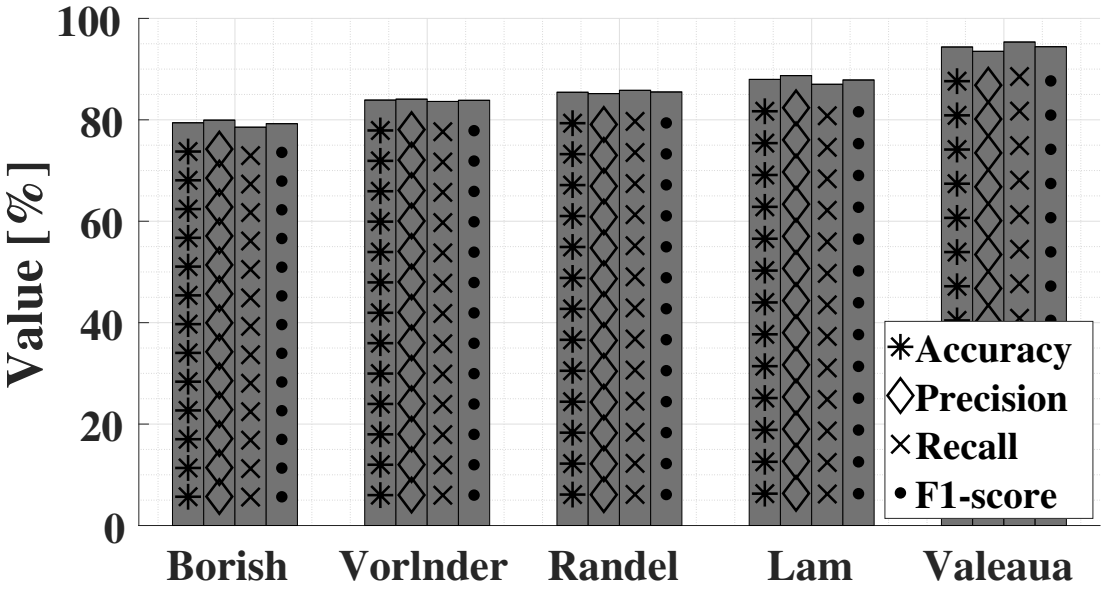

Figure 12.10: Performance of Ivry VAD in real reverberant conditions of large concert hall. Comparison is made between the five RIR training models.

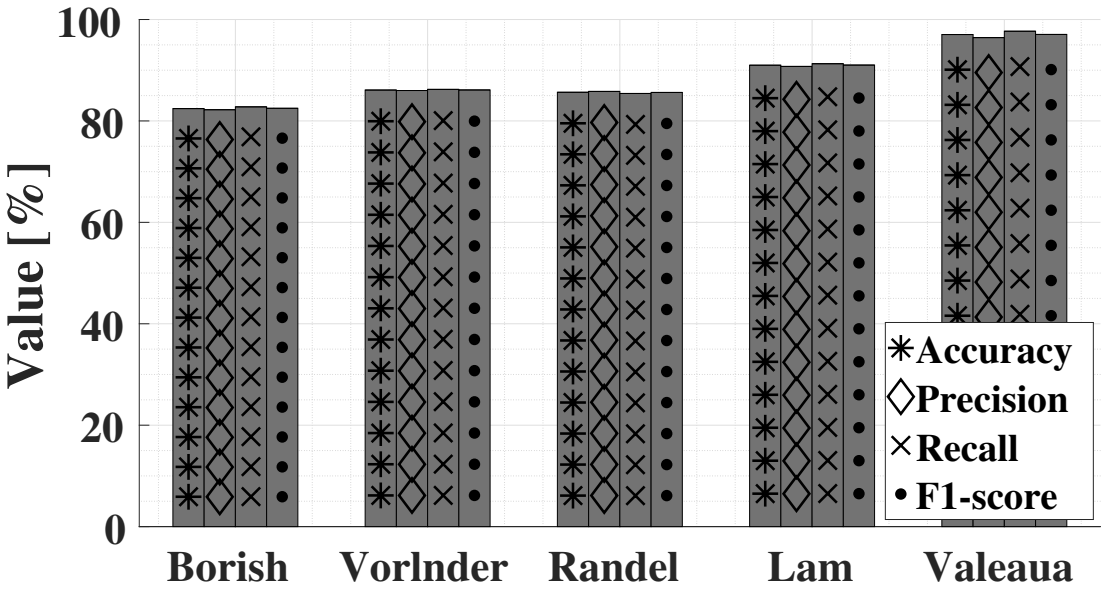

Figure 12.11: Performance of Ivry VAD in real reverberant conditions of octagon library. Comparison is made between the five RIR training models.

20% gap across all performance measures of accuracy, precision, recall and F1-score. This conclusion also implies high generalization ability of all VADs that are trained with [VPH06], since they consistently achieve enhanced performance for all measures and in all three acoustic environments. Next, let us focus on the interpretation of the accuracy, precision and recall measures. Since the training and test sets are balanced, these values strongly characterize the capabilities of the detector. The accuracy measure confirms that the Valeaua model leads to accurate detection in frames of both speech and silence. Also, the enhanced precision measure correspondingly lowers the false-positive value, i.e., non-speech frames has lower probability of being classified as speech. This result highly benefits applications such as speech enhancement, in which interferences may lead to severe degradation in practical performance. In a similar manner, the increase in recall decreases the false-negative measure. Thus, loss of information that typically lies in speech frames is obviated with higher probability.

Additional results are depicted in Figures 12.9–12.11. Here, we focus on Ivry VAD [IBC19] that achieved leading performance in all previous experimental results of this study. It can be deduced that this detector obtains a state-of-the-art performance of 95% in all reported measures when trained with Valeaua RIR model, which prevails competing VAD methods. Also, this detection system obtains leading accuracy, preci-

sion and recall measures across all tested reverberant setups. This outcome points on high generalization ability, robustness for noises and interferences, and prime accuracy in correctly distinguishing speech from silence.

12.4 Conclusions

In this study, we have considered five different state-of-the-art deep-learning-based VADs. We have shown that these detectors, when trained with merely anechoic data, experience substantial degradation in performance when tested in reverberant conditions. To mitigate this mismatch, we simulated an augmented training set that contains both an anechoic corpus and its reverberant transformation, where the latter was generated using a fixed room impulse response model. This extension permitted detectors to learn unique patterns and audio-based features that represent reverberant settings. The experiment was performed independently with five different room impulse response models. The training augmentation led to enhanced performance of all VAD systems when tested in three different real-life reverberant spaces. Improvement was obtained in terms of both accuracy, generalization and robustness abilities. Also, an average increase of 20% was held in accuracy, precision and recall measures with respect to non-reverberant training corpus. This study has also shown that the response model introduced by Valeaua [VPH06] consistently leads to the best performance, regardless of the detector and the tested acoustic environment. That and more, the VAD introduced by Ivry [IBC19] has achieved leading performance across all experiments. In future work, additional aspects such as feature engineering and dedicated architecture will be addressed in order to further enhance Ivry detector and adjust it for practical and reverberant acoustic scenarios.

Chapter 13

Discussion and Conclusions

13.1 Discussion and Conclusions

This research thesis has introduced state-of-the-art deep learning-based AEC systems, which are also adequate for embedding into practical hands-free speech communication platforms. As organizations shift to remote communication, and specifically to remote conferencing in office environments, there is also an increased need for AEC systems to perform reliably in real-life conditions. Often, the goal of existing studies was to improve the average benchmark performance and offer new neural network architectures. When tested in real-life conditions, however, these methods under-performed. Thus, a solution that can maintain high performance in various acoustic setups and in high echo and noise levels became essential. This thesis focuses on the different challenges of the AEC pipeline and offers solutions that highly perform in practical setups, including for non-linear AEC, linear AEC, RES, and objective performance assessment.

The non-linearity between the echo captured in the microphone and its origin in the far-end, which is becoming more dominant as hands-free communication devices undergo miniaturization. Existing solutions have often assumed complete linearity of the hardware that plays the far-end speech inside the near-end, which has constantly shown degraded performance when tested in real-life conditions. To handle this, our work allows to learn the non-linearities that power amplifiers and loudspeakers insert, and to estimate them with real-time tracking. This does not remove the non-linear effects from the AEC pipeline, but rather takes a novel approach of providing the

linear AEC system a reference of non-linear far-end speech estimate, which has linear relations with the echo captured in the microphone array. We have shown that this allows the linear AEC system to be utilized efficiently and to handle challenging real-life acoustic conditions while remaining lean in terms of resources.

A prominent challenge in AEC is the difficulty in tracking and estimating the linear echo path using adaptive filtering in real environments of frequent echo path changes and double-talk periods. To accommodate, we harnessed the power of deep learning to create a tracking model for the echo path that does not assume any acoustic setup or heuristic parameterization. A deep learning model learned the optimal adaptation step-size that promotes most rapid convergence of the misalignment between the true and estimated echo paths to its minimum. By design, this approach outperformed the competition in real-life scenarios and allowed to maintain low computational resources.

In practice, even enhanced performance for non-linear AEC and for linear adaption control are not sufficient for sustainably removing all acoustic echo. This is caused due to the mismatch between the lengths of real and estimated echo paths that create inherent linear adaptation error, due to the imperfection of non-linear AEC and linear AEC algorithms that tend to struggle when acoustic variety increases, and due to the inevitable convergence time of the adaptive filter that leads to degraded performance during this period. To that end, we constructed a deep RES system that maps the output of the linear AEC stage directly to the desired near-end speech. By utilized a larger amount of resources, this time we managed to build an RES model that also allows for dynamic tuning between the contracting RES demands of echo suppression and speech distortion. Using modern neural processors, however, have made our RES approach feasible for hands-free speech communication platforms on-edge.

In AEC systems, objective evaluation is often ambiguous and objective performance measures do not comply with human ratings in double-talk periods. This mismatch has brought inefficient progress to the field of AEC when inspecting real-life setups, due to broken compassing and biased results. To help resolve this issue, we developed objective performance measures that have high correlation with subjective human evaluation of AEC systems quality, i.e. the RESL and DSML, in contrast to the very low correlation exhibited by previously used measures like the SDR. By relating our new proposed

measures with modern needs of AEC systems, we also created a user-centric framework that allows users to choose the values of the RESL and DSML they desire, and thus to balance between echo suppression levels and speech distortion levels, while maintaining high subjective speech quality at the output of the AEC system.

SAEC is just as relevant as monophonic AEC for conferencing, but is vastly more challenging compared to its monophonic counterpart. We managed to successfully project concepts from our aforementioned monophonic approach to the stereophonic case, and to witness significant performance improvement in real-life conditions, while preserving low computational load that complies with hands-free communication standards. In one instance, we managed to improve adaptation control in the linear SAEC stage. In a second instance, we proposed an extension to the DSML and RESL performance measures to the stereophonic case, and showed that these also correlate well with subjective human evaluations compared to competing objective measures. In this process, we still managed the computational load according to on-edge requirements.

Another principal component to build a successful communication pipeline for modern conferencing is VAD. In reality, acoustic conditions often contain reverberations, transients, and stationary noises. We proposed a unique paradigm to distinguish speech segments in which raw data undergoes geometric analysis in which its underlying geometric structures are extracted and compared to ones of non-speech segments. Across practical acoustic setups, this approach has helped to separate speech from non-speech segments and to achieving extremely powerful performance in practice, with a modest amount of computational resources and a short system latency. This VAD system will propel succeeding hands-free speech communication system such as AEC, but also speech and speaker recognition, and speech diarization.

13.2 Future Research Directions

This thesis has introduced efficient implementations to improve each of the parts of an AEC system pipeline, including partially the stereophonic case, and of VAD systems. These advancements now allow for even further progress, from several research directions:

1) **Examine real-valued speech signal representations.** Decomposing the speech waveform signal into its frequency sub-bands using a real-valued transform, in contrast to the common complex-valued representation applied by the STFT and its modifications, can be efficient and powerful. This transform can enable a utilization of waveform-based deep learning models, and in certain cases lead to improved performance compared to their STFT-based counterparts. For instance, feedback-based neural networks that are specifically built for time sequence analysis can be applied efficiently using this representation, e.g. GRU and LSTM-based systems. Also, preservation of phase information is achieved, in contrast to STFT-based methods that usually introduce mismatch between the reconstructed amplitude and original phase information. In addition, every sub-band is associated with a lower sample frequency than the original signal, which may reduce the computational complexity and lower the inference time of the system.

2) **Develop a framework for real-time waveform-based speech processing.** Equipped with a sub-band decomposition of the speech signal, one can respectively decompose existing speech-based systems into smaller and more efficient sub-systems. Nowadays, waveform architectures are fed with the complete spectrum of speech signals that often demands high-resources consumption for high-quality modeling, which is not optimal for real-time usage. This direction aims to process each sub-band representation of the speech signal separately and independently by a smaller waveform-based architecture, and merge their outcomes. Hopefully, each sub-system will require a small computational load that is reasonable for embedding on real-time mobile communication platforms.

It should be noted that our work has utilized waveform-based processing in the context of non-linear acoustic-echo cancellation. This system managed to achieve leading results compared to competing methods that utilized time-frequency analysis, while operating with end-to-end system latency that coincides with real-time mobile communication standards. In addition, it consumes low amount of resources that renders it adequate for on-device integration in hands-free communication platforms. This success can also aim to explore additional speech-based applications, such as speech enhancement in noisy transient environments, speech intelligibility enhancement for

eavesdropping in unknown acoustic environments, and acoustic fencing.

3) **Extend the framework to additional speech processing applications.** We believe that the proposed waveform-based speech processing framework has the potential to constitute an essential part in the field of real-time hands-free communication. Specifically, we direct towards three future appliances; speech enhancement in noisy transient environments, speech intelligibility enhancement for eavesdropping using hidden microphone recordings, and acoustic fencing in a multi-participant conversation.

Speech enhancement in noisy transient environment regards the problem of recovering desired-speech signal from measurements that contain undesired noise and transient interference in difficult acoustic environments. Today, speech enhancement systems are highly desirable for various low-power hands-free communications platforms, such as smartphones, smart speakers, wearable devices, smart homes, IoT endpoints, and more. For instance, the Amazon Alexa speech inference is still impeded in real-life noisy environments, and construction noise outdoor still degrades speech intelligibility indoors, e.g. during a mobile phone conversation. Speech enhancement resembles our previous studies of residual-echo suppression and non-linear acoustic-echo cancellation, since in both cases speech should be recovered from degraded measurements. Speech enhancement also draws similarity to our VAD study that detected speech in transient noisy and reverberant environments. Thus, we propose to project the concepts we already successfully applied in previous systems to speech enhancement.

We also suggest to address speech intelligibility enhancement for eavesdropping using hidden microphone recordings in unknown acoustic environments. These may include ones with strong reverberations, echoes, and interference, and speech not directed at the microphone reception area. This solution should comprise a low-power, low-resources, on-device system that receives raw audio data, applies deep learning enhancement algorithms to it, and compresses it before transmission. Today, deep learning-based speech enhancement is mostly applied to speakers that want to be heard, intentionally recorded in improved conditions. These methods present leading performance in suppressing noise and interference by feeding time-frequency signal representations to complex and computationally-heavy models. However, they do not

address speech intelligibility enhancement in acoustic setups that characterize hidden microphone recordings and eavesdropping, and are inadequate for low-power, on-device applications. Achieving success in the proposed research is valuable in several aspects. First, creating a low-power speech intelligibility enhancement system and embedding it into stand-alone eavesdropping devices. Second, exploiting narrow-band transmission that is cheap, long-range, and low-power consuming, and extending the longevity of the battery-supplied device. Third, allowing more rapid and improved data inference by the end-user, i.e. the listener. Forth, reducing cost spent on human trainings that include big data collection. And fifth, achieving enhanced performance of following speech recognition algorithms.

Acoustic fencing aims at separating speakers by their physical locations in a room using a microphone array. Achieving success in this research can benefit many speech-based applications. For instance, it may improve speech enhancement of a speaker located in a certain region by attenuating speech sources that are located in other regions in the room. That and more, it may enhance succeeding speech-based systems, e.g. direction estimation, speaker recognition, and speech recognition. Another on-demand application nowadays is automatic transcription of conference meetings. By setting acoustic fences that isolate speakers located in different regions in the room, more accurate transcription results can be obtained compared with existing methods. The acoustic fencing system presents a challenging optimization between suppressing all sources that do not reside inside a certain region, and preserving the information from that region without distortions. Even though this system can operate offline and perform heavy computations on the cloud, it is instructive to allow on-device and real-time applicability, e.g. for automatic translation in a conversation between two remote sides. Essentially, this problem can be formulated as multi-channel speech separation, for which various waveform-based systems have been successfully utilized, however, inadequately in terms of computational resources. Given that we achieve the aforementioned speech enhancement system, we plan to extend it to a multi-channel architecture. Even though this system is originally built to preserve speech contaminated by noisy measurements, an alternative speech separation point of view can be adopted. In this perspective, each microphone now contains a desired-speech signal

and speech inference from several other sources, instead of transient noisy inference.

Bibliography

[AB79] J. B. Allen and D. A. Berkley. Image method for efficiently simulating small-room acoustics. *Journal of Acoustic Society of America*, 65(4):943–950, 1979.

[AC08] Y. Avargel and I. Cohen. Nonlinear acoustic echo cancellation based on a multiplicative transfer function approximation. In *Proc. International Workshop on Acoustic Signal Enhancement (IWAENC)*, pages 1–4. Citeseer, 2008.

[AC09a] Y. Avargel and I. Cohen. Adaptive nonlinear system identification in the short-time Fourier transform domain. *IEEE Transactions on Signal Processing*, 57(10):3891–3904, 2009.

[AC09b] Y. Avargel and I. Cohen. Modeling and identification of nonlinear systems in the short-time Fourier transform domain. *IEEE Transactions on Signal Processing*, 58(1):291–304, 2009.

[AC10] Y. Avargel and I. Cohen. Representation and identification of nonlinear system in the short-time Fourier transform domain. In I. Cohen, J. Benesty, and S. Gannot, editors, *Speech Processing in Modern Communication: Challenges and Perspectives*, chapter 3, pages 49–88. Springer, 2010.

[AC19] I. Ariav and I. Cohen. An end-to-end multimodal voice activity detection using WaveNet encoder and residual networks. *IEEE Journal of Selected Topics in Signal Processing*, 13(2):265–274, 2019.

[ADC18a] I. Ariav, D. Dov, and I. Cohen. A deep architecture for audio-visual voice activity detection in the presence of transients. *Signal Processing*, 142:69–74, 2018.

[ADC18b] I. Ariav, D. Dov, and I. Cohen. A deep architecture for audio-visual voice activity detection in the presence of transients. *Signal Processing*, 142:69–74, 2018.

[AMAZ17] S. Albawi, T. A. Mohammed, and S. Al-Zawi. Understanding of a convolutional neural network. In *Proc. International Conference Engineering Technology*, pages 1–6. IEEE, 2017.

[BCHC09] J. Benesty, J. Chen, Y. Huang, and I. Cohen. Pearson correlation coefficient. In *Noise Reduction in Speech Processing*, pages 1–4. Springer, 2009.

[BG95a] A. Birkett and R. A. Goubran. Limitations of handsfree acoustic echo cancellers due to nonlinear loudspeaker distortion and enclosure vibration effects. In *Proc. Workshop on Applications of Signal Processing to Audio and Accoustics (WASPAA)*, pages 103–106. IEEE, 1995.

[BG95b] A. N. Birkett and R. A. Goubran. Acoustic echo cancellation using NLMS-neural network structures. In *Proc. International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, volume 5, pages 3035–3038. IEEE, 1995.

[BGM+01] J. Benesty, T. Gänsler, D. R. Morgan, M. M. Sondhi, S. L. Gay, et al. *Advances in network and acoustic echo cancellation*. New York: Springer, 2001.

[BMS98] J. Benesty, D. R. Morgan, and M. M. Sondhi. A better understanding and an improved solution to the specific problems of stereophonic acoustic echo cancellation. *IEEE Transactions on Speech and Audio Processing*, 6(2):156–165, 1998.

[BNH18] T. Ben-Nun and T. Hoefler. Demystifying parallel and distributed deep learning: An in-depth concurrency analysis. *preprint arXiv:1802.09941*, 2018.

[Bor84] J. Borish. Extension of the image model to arbitrary polyhedra. *The Journal of the Acoustical Society of America*, 75(6):1827–1836, 1984.

[BPGC11] J. Benesty, C. Paleologu, T. Gänsler, and S. Ciochină. *A Perspective on Stereophonic Acoustic Echo Cancellation*, volume 4. Springer Science & Business Media, 2011.

[Bri90] J. Bridle. Probabilistic interpretation of feedforward classification network outputs, with relationships to statistical pattern recognition. In *Neurocomputing: Algorithms, architectures and applications*, pages 227–236, 1990.

[BRVT06] J. Benesty, H. Rey, L. R. Vega, and S. Tressens. A nonparametric VSS NLMS algorithm. *IEEE Signal Processing Letters*, 13(10):581–584, 2006.

[CB01] I. Cohen and B. Berdugo. Speech enhancement for non-stationary noise environments. *Signal Processing*, 81(11):2403–2418, November 2001.

[CBS22] J. Casebeer, N. J. Bryan, and P. Smaragdis. Meta-AF: Meta-learning for adaptive filters. *IEEE Transactions on Audio, Speech, Language Processing*, 31:355–370, 2022.

[CDX20] M. Chao, L. Dongmei, and J. Xupeng. Optimal scale-invariant signal-to-noise ratio and curriculum learning for monaural multi-speaker speech separation in noisy environment. In *Proc. Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, pages 711–715. IEEE, 2020.

[CGCB14] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling. *preprint arXiv:1412.3555*, 2014.

[CH14] R. R. Coifman and M. J. Hirn. Diffusion maps for changing data. *Applied and Computational Harmonic Analysis*, 36(1):79–107, 2014.

[CK11] N. Cho and E. K. Kim. Enhanced voice activity detection using acoustic event detection and classification. *IEEE Transactions Consumer Electronins*, 57(1):196–202, 2011.

[CKM06] J. H. Chang, N. S. Kim, and S. K. Mitra. Voice activity detection based on multiple statistical models. *IEEE Transactions Signal Processing*, 54(6):1965–1976, 2006.

[CL06a] R. R. Coifman and S. Lafon. Diffusion maps. *Applied and Computational Harmonic Analysis*, 21(1):5–30, 06 2006.

[CL06b] R. R. Coifman and S. Lafon. Geometric harmonics: a novel tool for multiscale out-of-sample extension of empirical functions. *Applied and Computational Harmonic Analysis*, 21(1):31–52, 2006.

[CNL+21] R. Cutler, B. Nadari, M. Loide, S. Sootla, and A. Saabas. Crowdsourcing approach for subjective evaluation of echo impairment. In *Proc. International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 406–410. IEEE, 2021.

[Coh03] I. Cohen. Noise spectrum estimation in adverse environments: Improved minima controlled recursive averaging. *IEEE Transactions Speech and Audio Processing*, 11(5):466–475, September 2003.

[Cro80] R. Crochiere. A weighted overlap-add method of short-time Fourier analysis/synthesis. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 28(1):99–102, 1980.

[CRPP12] S. Cecchi, L. Romoli, P. Peretti, and F. Piazza. Low-complexity implementation of a real-time decorrelation algorithm for stereophonic acoustic echo cancellation. *Signal Processing*, 92(11):2668–2675, 2012.

[CSAR+13] D. Comminiello, M. Scarpiniti, L. A. Azpicueta-Ruiz, J. Arenas-García, and A. Uncini. Functional link adaptive filters for nonlinear acoustic echo cancellation. *IEEE Transactions on Audio, Speech, and Language Processing*, 21(7):1502–1512, 2013.

[CSP+21] R. Cutler, A. Saabas, T. Parnamaa, M. Loida, S. Sootla, et al. Interspeech 2021 Acoustic Echo Cancellation Challenge. In *Proc. Interspeech*, pages 4748–4752, 2021.

[CSP+22] R. Cutler, A. Saabas, T. Parnamaa, M. Purin, H. Gamper, et al. ICASSP 2022 acoustic echo cancellation challenge. In *Proc. International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 9107–9111. IEEE, 2022.

[CSP+23] R. Cutler, A. Saabas, T. Parnamaa, M. Purin, E. Indenbom, et al. ICASSP 2023 acoustic echo cancellation challenge. In *Proc. International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, page to appear, 2023.

[CSVH18] G. Carbajal, R. Serizel, E. Vincent, and E. Humbert. Multiple-input neural network-based residual echo suppression. In *Proc. International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 231–235, 2018.

[CSVH19] G. Carbajal, R. Serizel, E. Vincent, and E. Humbert. Joint DNN-Based Multichannel Reduction of Acoustic Echo, Reverberation and Noise. *preprint arXiv:1911.08934*, 2019.

[CXCL20] H. Chen, T. Xiang, K. Chen, and J. Lu. Nonlinear residual echo suppression based on multi-stream Conv-Tasnet. *preprint arXiv:2005.07631*, 2020.

[DA12] G. David and A. Averbuch. Hierarchical data organization, clustering and denoising via localized diffusion folders. *Applied and Computational Harmonic Analysis*, 33(1):1–23, 2012.

[DC14] D. Dov and I. Cohen. Voice activity detection in presence of transients using the scattering transform. In *Proc. 28th Convention of the Electrical & Electronics Engineers in Israel (IEEEI)*, pages 1–5, 2014.

[DDBW19] N. K. Desiraju, S. Doclo, M. Buck, and T. Wolff. Online estimation of reverberation parameters for late residual echo suppression. *IEEE/ACM Transactions Audio, Speech, Language Processing*, 28:77–91, 2019.

[DDBW22] N. K. Desiraju, S. Doclo, M. Buck, and T. Wolff. Joint online estimation of early and late residual echo PSD for residual echo suppression. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 31:333–344, 2022.

[DM80] S. Davis and P. Mermelstein. Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. *IEEE Transactions on Acoustics, Speech Signal Processing*, 28(4):357–366, 1980.

[Dob11] A. Dobrucki. Nonlinear distortions in electroacoustic devices. *Archives of Acoustics*, 36(2):437–460, 2011.

[DPBC20] L. Dogariu, C. Paleologu, J. Benesty, and S. Ciochină. An efficient kalman filter for the identification of low-rank systems. *Signal Processing*, 166:107239, 2020.

[DSA20] A. Defossez, G. Synnaeve, and Y. Adi. Real time speech enhancement in the waveform domain. *preprint arXiv:2006.12847*, 2020.

[DTC15] D. Dov, R. Talmon, and I. Cohen. Audio-visual voice activity detection using diffusion maps. *IEEE Transactions on Audio, Speech Language Processing*, 23(4):732–745, 2015.

[DTC16] D. Dov, R. Talmon, and I. Cohen. Kernel method for voice activity detection in the presence of transients. *IEEE Transactions Audio, Speech, Language Processing*, 24(12):2313–2326, 2016.

[ETS16] ETSI ES 202 740: Speech and multimedia Transmission Quality (STQ); Transmission requirements for wideband VoIP loudspeaking and handsfree terminals from a QoS perspective as perceived by the user, 2016.

[Fan20a] B. Fang. An integrated system of adaptive echo cancellation and residual echo suppression. In *Proc. International Conference on Computational Communication Information Systems*, pages 19–23, 2020.

[Fan20b] B. Fang. A robust residual echo suppression algorithm even during double talk. In *Proc. International Conference Information Communication Signal Processing (ICICSP)*, pages 6–9. IEEE, 2020.

[FBD+22] I. Fîciu, J. Benesty, L. Dogariu, C. Paleologu, and S. Ciochină. Efficient algorithms for linear system identification with particular symmetric filters. *Applied Sciences*, 12(9):4263, 2022.

[FD93] N. Freire and S. C. Douglas. Adaptive cancellation of geomagnetic background noise using a sign-error normalized LMS algorithm. In *Proc. International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, volume 3, pages 523–526. IEEE, 1993.

[FEKL20] A. Fazel, M. El-Khamy, and J. Lee. CAD-AEC: Context-Aware Deep Acoustic Echo Cancellation. In *Proc. International Conference Acoustics, Speech and Signal Processing (ICASSP)*, pages 6919–6923, 2020.

[FF22] J. Franzen and T. Fingscheidt. Deep residual echo suppression and noise reduction: A multi-input FCRN approach in a hybrid speech enhancement system. In *Proc. International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 666–670. IEEE, 2022.

[FFL10] Z. Farbman, R. Fattal, and D. Lischinski. Diffusion maps for edge-aware image editing. *ACM Transactions Graph*, 29(6):145:1–145:10, 12 2010.

[FZ11] M. M. U. Faiz and A. Zerguine. A steady-state analysis of the ε-normalized sign-error least mean square (NSLMS) adaptive algorithm. In *Proc. Conference Record of the Forty Fifth Asilomar Conference on Signals, Systems and Computers (ASILOMAR)*, pages 538–541. IEEE, 2011.

[Gau01] T. D. Gauthier. Detecting trends using Spearman's rank correlation coefficient. *Environmental Forensics*, 2(4):359–362, 2001.

[GB02] T. Gansler and J. Benesty. New insights into the stereophonic acoustic echo cancellation problem and an adaptive nonlinearity solution. *IEEE Transactions Speech Audio Processing*, 10(5):257–267, 2002.

[GFLBJ03] A. Guérin, G. Faucon, and R. Le Bouquin-Jeannès. Nonlinear acoustic echo cancellation based on Volterra filters. *IEEE Transactions on Speech and Audio Processing*, 11(6):672–683, 2003.

[GHD+17] K. G. Ghasedi, A. Herandi, C. Deng, W. Cai, and H. Huang. Deep clustering via joint convolutional autoencoder embedding and relative entropy minimization. In *Proc. International Conference on Computer Vision (ICCV)*, pages 5736–5745, 2017.

[GK13] S. Gepshtein and Y. Keller. Image completion by diffusion maps and spectral relaxation. *IEEE Transactions Image Processing*, 22(8):2839–2846, 2013.

[GL84] D. Griffin and J. Lim. Signal estimation from modified short-time Fourier transform. *IEEE Transactions Acoustic, Speech, Signal Processing*, 32(2):236–243, 1984.

[GLA+20] P. K. Gadosey, Y. Li, E. A. Agyekum, T. Zhang, Z. Liu, et al. SD-UNET: Stripping down u-net for segmentation of biomedical images on platforms with low computational budgets. *Diagnostics*, 10(2):110, 2020.

[GLF+88] J. S. Garofolo, L. F. Lamel, W. M. Fisher, J. G. Fiscus, and D. S. Pallett. Getting started with the DARPA TIMIT CD-ROM: An acoustic phonetic continuous speech database. *National Institute of Standards and Technology (NIST), Gaithersburgh, MD*, 107:16, 1988.

[GLF+93a] J. S. Garofolo, L. F. Lamel, W. M. Fisher, J. G. Fiscus, and D. S. Pallett. DARPA TIMIT acoustic-phonetic continous speech corpus CD-ROM. NIST speech disc 1-1.1. *NASA STI/Recon technical report*, 93:27403, 1993.

[GLF+93b] J. S. Garofolo, L. F. Lamel, W. M. Fisher, J. G. Fiscus, D. S. Pallett, et al. DARPA TIMIT acoustic-phonetic continous speech corpus CD-

ROM. NIST speech disc 1-1.1. Technical Report LDC93S1, National Institute of Standards Technology, Gaithersburg, MD, USA, 1993.

[GMH13] A. Graves, A. R. Mohamed, and G. Hinton. Speech recognition with deep recurrent neural networks. In *Proc. International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6645–6649, 2013.

[GR96] S. Gold and A. Rangarajan. Softmax to softassign: Nerual network algorithms for combinational optimization. *Journal of Artificial Neural Networks*, 2(4):381–399, 1996.

[GS97] E. B. George and M. Smith. Speech analysis/synthesis and modification using an analysis-by-synthesis/overlap-add sinusoidal model. *IEEE Transactions on Speech and Audio Processing*, 5(5):389–406, Sep. 1997.

[GSDY15] R. Giri, M. L. Seltzer, J. Droppo, and D. Yu. Improving speech recognition in reverberation using a room-aware deep neural network and multi-task learning. In *Proc. International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5014–5018. IEEE, 2015.

[GT98] A. Gilloire and V. Turbin. Using auditory properties to improve the behaviour of stereophonic acoustic echo cancellers. In *Proc. International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, volume 6, pages 3681–3684. IEEE, 1998.

[GV92] A. Gilloire and M. Vetterli. Adaptive filtering in sub-bands with critical sampling: analysis, experiments, and application to Acoust. echo cancellation. *IEEE Transactions Signal Processing*, 40(8):1862–1875, 1992.

[HA16] M. Hamidia and A. Amrouche. Improved variable step-size NLMS adaptive filtering algorithm for acoustic echo cancellation. *Digital Signal Processing*, 49:44–55, 2016.

[HBEK21] T. Haubner, A. Brendel, M. Elminshawi, and W. Kellermann. Noise-robust adaptation control for supervised acoustic system identification

exploiting a noise dictionary. In *Proc. International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 945–949. IEEE, 2021.

[HBK21] T. Haubner, A. Brendel, and W. Kellermann. End-to-end deep learning-based adaptation control for frequency-domain adaptive system identification. *preprint arXiv:2106.01262*, 2021.

[HBK22] T. Haubner, A. Brendel, and W. Kellermann. End-to-end deep learning-based adaptation control for frequency-domain adaptive system identification. In *Proc. International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 766–770. IEEE, 2022.

[HHB+20] T. Haubner, M. M. Halimeh, A. Brendel, W. Kellermann, et al. A synergistic Kalman-and deep postfiltering approach to acoustic echo cancellation. *preprint arXiv:2012.08867*, 2020.

[HHBK21] T. Haubner, M. M. Halimeh, A. Brendel, and W. Kellermann. A synergistic kalman-and deep postfiltering approach to acoustic echo cancellation. In *Proc. European Signal Processing Conference (EUSIPCO)*, pages 990–994. IEEE, 2021.

[HHK19] M. M. Halimeh, C. Huemmer, and W. Kellermann. A neural network-based nonlinear acoustic echo canceller. *IEEE Signal Processing Letters*, 26(12):1827–1831, 2019.

[HK20] M. M. Halimeh and W. Kellermann. Efficient Multichannel Nonlinear Acoustic Echo Cancellation Based on a Cooperative Strategy. In *Proc. International Conference Acoustics, Speech and Signal Processing (ICASSP)*, pages 461–465, 2020.

[HKC14] A. Haddad, D. Kushnir, and R. R. Coifman. Texture separation via a reference set. *Applied and Computational Harmonic Analysis*, 36(2):335–347, 03 2014.

[HL11] H. C. Huang and J. Lee. A new variable step-size NLMS algorithm and its performance analysis. *IEEE Transactions on Signal Processing*, 60(4):2055–2060, 2011.

[HL13] W. T. Hong and C. C. Lee. Voice activity detection based on noise-immunity recurrent neural networks. *International Journal on Advanced Computational Technology (IJACT)*, 5(5):338–345, 2013.

[HM13] T. Hughes and K. Mierle. Recurrent neural networks for voice activity detection. In *Proc. International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7378–7382, 2013.

[HQZ+23] L. Huang, J. Qin, Y. Zhou, F. Zhu, L. Liu, et al. Normalization techniques in training dnns: Methodology, analysis and application. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023.

[HWW+15] K. Han, Y. Wang, D. Wang, W. S. Woods, I. Merks, et al. Learning spectral mapping for speech dereverberation and denoising. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 23(6):982–992, 2015.

[HZ06] W. W. Hager and H. Zhang. A new active set algorithm for box constrained optimization. *SIAM Journal on Optimization*, 17(2):526–557, 2006.

[IBC19] A. Ivry, B. Berdugo, and I. Cohen. Voice activity detection for transient noisy environment based on diffusion nets. *IEEE Journal of Selected Topics in Signal Processing*, 13(2):254–264, 2019.

[IBMH22] G. Imen, A. Benallal, M. Mekarzia, and I. Hassani. The NP-VSS NLMS algorithm with noise power estimation methods for acoustic echo cancellation. In *Proc. International Conference on Advanced Electrical Engineering (ICAEE)*, pages 1–6. IEEE, 2022.

[ICB21a] A. Ivry, I. Cohen, and B. Berdugo. Deep residual echo suppression with a tunable tradeoff between signal distortion and echo suppression. In

Proc. International Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 126–130. IEEE, 2021.

[ICB21b] A. Ivry, I. Cohen, and B. Berdugo. Nonlinear acoustic echo cancellation with deep learning. In *Proc. Interspeech*. IEEE, sep 2021.

[ICB21c] A. Ivry, I. Cohen, and B. Berdugo. Objective metrics to evaluate residual-echo suppression during double-talk. In *Proc. Workshop on Applications of Signal Processing to Audio and Accoustics (WASPAA)*, pages 101–105. IEEE, 2021.

[ICB22a] A. Ivry, I. Cohen, and B. Berdugo. Deep adaptation control for acoustic echo cancellation. In *Proc. International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 741–745. IEEE, 2022.

[ICB22b] A. Ivry, I. Cohen, and B. Berdugo. Objective metrics to evaluate residual-echo suppression during double-talk in the stereophonic case. *Proc. Interspeech*, pages 5348–5352, 2022.

[ICB22c] A. Ivry, I. Cohen, and B. Berdugo. Off-the-shelf deep integration for residual-echo suppression. In *Proc. International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 746–750. IEEE, 2022.

[ITU01] ITU-T Rec. P.862: Perceptual evaluation of speech quality (PESQ): An objective method for end-to-end speech quality assessment of narrow-band telephone networks and speech codecs, Feb. 2001.

[ITU12] ITU-T Rec. G.168: Digital network echo cancellers, Feb. 2012.

[ITU17] ITU-T Rec. P.862.2: Wideband extension to recommendation P. 862 for the assessment of wideband telephone networks and speech codecs, Oct. 2017.

[Jan04] A. Janczak. *Identification of nonlinear systems using neural networks and polynomial models: a block-oriented approach*, volume 310. Springer Science & Business Media, 2004.

[JMR94] J.C. Junqua, B. Mak, and B. Reaves. A robust algorithm for word boundary detection in the presence of noise. *IEEE Transactions Speech Audio Processing*, 2(3):406–412, 1994.

[KB15] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. In *Proc. International Conference on Learning Representations ICLR*, 2015.

[KDG+16] K. Kinoshita, M. Delcroix, S. Gannot, E. Habets, R. Haeb-Umbach, et al. A summary of the REVERB challenge: state-of-the-art and remaining challenges in reverberant speech processing research. *EURASIP Journal on Advances in Signal Processing*, 2016(1):7, 2016.

[KDJJ16] A. Kuklasiński, S. Doclo, S. H. Jensen, and J. Jensen. Maximum likelihood PSD estimation for speech enhancement in reverberation and noise. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 24(9):1599–1612, 2016.

[KH18] J. Kim and M. Hahn. Voice activity detection using an adaptive context attention model. *IEEE Signal Processing Letters*, 25(8):1181–1185, 2018.

[KJS21] E. Kim, J. Jeon, and H. Seo. U-convolution based residual echo suppression with multiple encoders. In *Proc. International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 925–929. IEEE, 2021.

[Kli05] W. Klippel. Loudspeaker nonlinearities–causes, parameters, symptoms. In *Audio Engineering Society Convention 119*. Audio Engineering Society, 2005.

[KN91] D. A. Krubsack and R. J. Niederjohn. An autocorrelation pitch detector and voicing decision with confidence measures developed for noise-corrupted speech. *IEEE Transactions Signal Processing*, 39(2):319–329, 1991.

[KPP+17] T. Ko, V. Peddinti, D. Povey, M. L. Seltzer, and S. Khudanpur. A study on data augmentation of reverberant speech for robust speech recognition. In

Proc. International Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 5220–5224. IEEE, 2017.

[KS17] A. Kar and M. Swamy. Tap-length optimization of adaptive filters used in stereophonic acoustic echo cancellation. *Signal Processing*, 131:422–433, 2017.

[KTC17] P. Kedar, S. P. Taher, and D. P. Chinmay. A comparative study of categorical variable encoding techniques for neural network classifiers. *International Journal of Computational Applications*, 175(4):7–9, 2017.

[Lam05] Y. W. Lam. Issues for computer modelling of room acoustics in non-concert hall settings. *Acoustical Science and Technology*, 26(2):145–155, 2005.

[LCH+19] Q. Lei, H. Chen, J. Hou, L. Chen, and L. Dai. Deep Neural Network Based Regression Approach for Acoustic Echo Cancellation. In *Proc. International Conference Multimedia Systems and Signal Processing*, pages 94–98, 2019.

[LHB15] S. Leglaive, R. Hennequin, and R. Badeau. Singing voice detection with deep recurrent neural networks. In *Proc. International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 121–125, 2015.

[LKC06] S. Lafon, Y. Keller, and R. R. Coifman. Data fusion and multicue data matching by diffusion maps. *IEEE Transactions Pattern Analysis and Machine Intelligence*, 28(11):1784–1797, 11 2006.

[LM19] Y. Luo and N. Mesgarani. Conv-TasNet: Surpassing ideal time–frequency magnitude masking for speech separation. *IEEE/ACM Transactions Audio, Speech, and Language Processing*, 27(8):1256–1266, 2019.

[Log00] B. Logan. Mel frequency cepstral coefficients for music modeling. In *International Society for Music Information Retrieval (ISMIR)*, 2000.

[LSK15] C. M. Lee, J. W. Shin, and N. S. Kim. DNN-based residual echo suppression. In *Proc. Annual Conference of the International Speech Communication Association*, 2015.

[LTMH13] X. Lu, Y. Tsao, S. Matsuda, and C. Hori. Speech enhancement based on deep denoising autoencoder. In *Proc. Interspeech*, pages 436–440, 08 2013.

[LVRH16] C. Lea, R. Vidal, A. Reiter, and G. D. Hager. Temporal convolutional networks: A unified approach to action segmentation. In *Proc. European Conference on Computer Vision (ECCV)*, pages 47–54. Springer, 2016.

[LWS16] M. Liu, M. J. Wang, and B. Y. Song. An efficient architecture of the sign-error LMS adaptive filter. In *Proc. Solid-State and Integrated Circuits Technology (ICSICT)*, pages 753–755. IEEE, 2016.

[MC13a] G. Mishne and I. Cohen. Multiscale anomaly detection using diffusion maps. *IEEE Journal of Selected Topics in Signal Processing*, 7:111–123, 02 2013.

[MC13b] S. Mousazadeh and I. Cohen. Voice activity detection in presence of transient noise using spectral clustering. *IEEE Transactions on Audio, Speech Language Processing*, 21(6):1261–1271, 2013.

[ME12] S. Malik and G. Enzner. State-space frequency-domain adaptive filtering for nonlinear acoustic echo cancellation. *IEEE Transactions on Audio, Speech, and Language Processing*, 20(7):2065–2079, 2012.

[MEB10] M. I. Mossi, N. W. Evans, and C. Beaugeant. An assessment of linear adaptive filter performance with nonlinear distortions. In *Proc. International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 313–316. IEEE, 2010.

[MEG+16] D. Michal, V. Eugene, C. Gabriel, K. Samuel, and P. Chris. The importance of skip connections in biomedical image segmentation. In *Deep Learning and Data Labeling for Medical Applications*, pages 179–187. Springer, 2016.

[MHB01] D. R. Morgan, J. L. Hall, and J. Benesty. Investigation of several types of nonlinearities for use in stereo acoustic echo cancellation. *IEEE Transactions on Speech Audio Processing*, 9(6):686–696, 2001.

[MHZS20] L. Ma, H. Huang, P. Zhao, and T. Su. Acoustic Echo Cancellation by Combining Adaptive Digital Filter and Recurrent Neural Network. *preprint arXiv:2005.09237*, 2020.

[MK16a] J. Malek and Z. Koldovskỳ. Hammerstein model-based nonlinear echo cancelation using a cascade of neural network and adaptive linear filter. In *Proc. International Workshop on Acoustic Signal Enhancement (IWAENC)*, pages 1–5. IEEE, 2016.

[MK16b] S. Meier and W. Kellermann. Relative impulse response estimation during double-talk with an artificial neural network-based step size control. In *Proc. International Workshop on Acoustic Signal Enhancement (IWAENC)*, pages 1–5. IEEE, 2016.

[MPP15] V. S. Mendelev, T. N. Prisyach, and A. A. Prudnikov. Robust voice activity detection with deep maxout neural networks. *Modern Applied Science*, 9(8):153, 2015.

[MSCC17] G. Mishne, U. Shaham, A. Cloninger, and I. Cohen. Diffusion nets. *Applied and Computational Harmonic Analysis*, 2017.

[NCC16] J. Ni, J. Chen, and X. Chen. Diffusion sign-error LMS algorithm: Formulation and stochastic behavior analysis. *Signal Processing*, 128:142–149, 2016.

[Nic18] A. Nicolae. PLU: The piecewise linear unit activation function. *preprint arXiv:1809.09534*, 2018.

[PBC14] C. Paleologu, J. Benesty, and S. Ciochină. Widely linear general kalman filter for stereophonic acoustic echo cancellation. *Signal Processing*, 94:570–575, 2014.

[PCBG15] C. Paleologu, S. Ciochină, J. Benesty, and S. L. Grant. An overview on optimized NLMS algorithms for acoustic echo cancellation. *EURASIP Journal on Advances in Signal Processing*, 2015(1):1–19, 2015.

[PCPK15] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur. Librispeech: an asr corpus based on public domain audio books. In *Proc. International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5206–5210, 2015.

[PGC+17] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, et al. Automatic differentiation in pytorch. In *Proc. Neural Information Processing Systems (NIPS)*, 2017.

[Pow20] D. Powers. Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation. *preprint arXiv:2010.16061*, 2020.

[PP20] L. Pfeifenberger and F. Pernkopf. Nonlinear residual echo suppression using a recurrent neural network. In *Proc. Interspeech*, pages 3950–3954, 2020.

[PSK+20] S. H. Pauline, D. Samiappan, R. Kumar, A. Anand, and A. Kar. Variable tap-length non-parametric variable step-size NLMS adaptive filtering algorithm for acoustic echo cancellation. *Applied Acoustics*, 159:107074, 2020.

[PSS+22] M. Purin, S. Sootla, M. Sponza, A. Saabas, and R. Cutler. AECMOS: A speech quality assessment metric for echo impairment. In *Proc. International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 901–905. IEEE, 2022.

[RBHH01] A. W. Rix, J. G. Beerends, M. P. Hollier, and A. P. Hekstra. Perceptual evaluation of speech quality (PESQ)-a new method for speech quality assessment of telephone networks and codecs. In *Proc. International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, volume 2, pages 749–752. IEEE, 2001.

[RBP+19] C. Reddy, E. Beyrami, J. Pool, R. Cutler, S. Srinivasan, et al. A scalable noisy speech dataset and online subjective test framework. *preprint arXiv:1909.08050*, 2019.

[RCP+10] L. Romoli, S. Cecchi, L. Palestini, P. Peretti, and F. Piazza. A novel approach to channel decorrelation for stereo acoustic echo cancellation based on missing fundamental theory. In *Proc. International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 329–332. IEEE, 2010.

[RFB15] O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In *Proc. Medical Image Computing and Computer-Assisted Intervention*, pages 234–241. Springer, 2015.

[RGC21] C. K. A. Reddy, V. Gopal, and R. Cutler. DNSMOS: A non-intrusive perceptual objective speech quality metric to evaluate noise suppressors. In *Proc. International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6493–6497. IEEE, 2021.

[Rin93] J. H. Rindel. Modelling the angle-dependent pressure reflection factor. *Applied Acoustics*, 38(2-4):223–234, 1993.

[RLRL10] R. Ravaud, G. Lemarquand, T. Roussel, and V. Lemarquand. Ranking of the nonlinearities of electrodynamic loudspeakers. *Archives of Acoustics*, 35(1):49–66, 2010.

[RPL09] J. D. Rodriguez, A. Perez, and J. A. Lozano. Sensitivity analysis of k-fold cross validation in prediction error estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(3):569–575, 2009.

[RSB+04] J. Ramírez, J. C. Segura, C. Benítez, A. De La Torre, and A. Rubio. Efficient voice activity detection algorithms using long-term speech information. *Speech Communication*, 42(3):271–287, 2004.

[RT98] A. B. Rabaa and R. Tourki. Acoustic echo cancellation based on a recurrent neural network and a fast affine projection algorithm. In *Proc. An-*

nual Conference Industrial Electronics Society (IECON), volume 3, pages 1754–1757. IEEE, 1998.

[Rus11] A. Ruszczynski. *Nonlinear optimization*. Princeton university press, 2011.

[SB99] S. G. Sankaran and A. Beex. Stereophonic acoustic echo cancellation using NLMS with orthogonal correction factors. In *Proc. International Workshop on Acoustic Signal Enhancement (IWAENC)*, pages 40–43. Citeseer, 1999.

[SBP+13] C. Stanciu, J. Benesty, C. Paleologu, T. Gänsler, and S. Ciochină. A widely linear model for stereophonic acoustic echo cancellation. *Signal Processing*, 93(2):511–516, 2013.

[SCK10] J. W. Shin, J. H. Chang, and N. S. Kim. Voice activity detection based on statistical models and machine learning approaches. *Computational Speech Language*, 24(3):515–530, 2010.

[SCPU11] M. Scarpiniti, D. Comminiello, R. Parisi, and A. Uncini. Comparison of Hammerstein and Wiener systems for nonlinear acoustic echo cancelers in reverberant environments. In *Proc. International Conference on Digital Signal Processing*, pages 1–6. IEEE, 2011.

[SCS+21] K. Sridhar, R. Cutler, A. Saabas, T. Parnamaa, M. Loide, et al. ICASSP 2021 acoustic echo cancellation challenge: Datasets, testing framework, and results. In *Proc. International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 151–155. IEEE, 2021.

[SDA22] M. Salah, M. Dessouky, and B. Abdelhamid. Design and implementation of an improved variable step-size NLMS-based algorithm for acoustic noise cancellation. *Circuits, Systems, and Signal Processing*, 41(1):551–578, 2022.

[SDT21] M. Schmidtner, C. Doering, and H. Timinger. Agile working during COVID-19 pandemic. *IEEE Engineering Management Review*, 49(2):18–32, 2021.

[SHK+14a] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15(1):1929–1958, 2014.

[SHK+14b] N. Srivastava, G. E. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15(1):1929–1958, 2014.

[SK15] A. Schwarz and W. Kellermann. Coherent-to-diffuse power ratio estimation for dereverberation. *IEEE Transactions on Audio, Speech, and Language Processing*, 23(6):1006–1018, 2015.

[SKS99] J. Sohn, N. S. Kim, and W. Sung. A statistical model-based voice activity detection. *IEEE Signal Processing Letters*, 6(1):1–3, 1999.

[SMH95] M. M. Sondhi, D. R. Morgan, and J. L. Hall. Stereophonic acoustic echo cancellation-an overview of the fundamental problem. *IEEE Signal Processing Letters*, 2(8):148–151, 1995.

[SRGZ+04] M. Soria-Rodríguez, M. Gabbouj, N. Zacharov, M. S. Hamalainen, and K. Koivuniemi. Modeling and real-time auralization of electrodynamic loudspeaker non-linearities. In *Proc. International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, volume 4, pages 81–84. IEEE, 2004.

[SS10] R. Stewart and M. Sandler. Database of omnidirectional and B-format room impulse responses. In *Proc. International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 165–168. IEEE, 2010.

[SSM+19] I. Szöke, M. Skácel, L. Mošner, J. Paliesek, and J. Černockỳ. Building and evaluation of a real room impulse response dataset. *IEEE Journal of Selected Topics in Signal Processing*, 13(4):863–876, 2019.

[SSN09] A. Singer, Y. Shkolnisky, and B. Nadler. Diffusion interpretation of nonlocal neighborhood filters for signal denoising. *SIAM Journal on Imaging Sciences*, 2(1):118–139, 01 2009.

[Syn21] NDP120 Syntiant™ Neural Processor. `https://www.syntiant.com/ndp120`, 2021.

[TCG12] R. Talmon, I. Cohen, and S. Gannot. Single-channel transient interference suppression with diffusion maps. *IEEE Transactions on Audio, Speech Language Processing*, 21(1):130–142, 04 2012.

[TCGC13] R. Talmon, I. Cohen, S. Gannot, and R. R. Coifman. Diffusion maps for signal processing: A deeper look at manifold-learning techniques based on kernels and graphs. *IEEE Signal Processing Magazine*, 30(4):75–86, 2013.

[TI19] AM5749 Sitara™ Processor. `https://www.ti.com/product/AM5749?qgpn=am5749`, 2019.

[TIH+10] S. Tamura, M. Ishikawa, T. Hashiba, S. Takeuchi, and S. Hayamizu. A robust audio-visual speech recognition using audio-visual voice activity detection. In *Proc. INTERSPEECH*, pages 2694–2697, 2010.

[VGF06] E. Vincent, R. Gribonval, and C. Févotte. Performance measurement in blind audio source separation. *IEEE Transactions on Audio, Speech, and Language Processing*, 14(4):1462–1469, 2006.

[VGX97] S. Van Gerven and F. Xie. A comparative study of speech detection methods. In *Proc. 5th European Conference on Speech Communication and Technology (EUROSPEECH)*, pages 1095–1098, 1997.

[Vir99] N. Virag. Single channel speech enhancement based on masking properties of the human auditory system. *IEEE Transactions on Speech and audio Processing*, 7(2):126–137, 1999.

[Vor89] M. Vorländer. Simulation of the transient and steady-state sound propagation in rooms using a new combined ray-tracing/image-source algorithm. *The Journal of the Acoustical Society of America*, 86(1):172–178, 1989.

[VPH06] V. Valeau, J. Picaut, and M. Hodgson. On the use of a diffusion equation for room-acoustic prediction. *The Journal of the Acoustical Society of America*, 119(3):1504–1513, 2006.

[VVARC16] S. Van Vaerenbergh, L. A. Azpicueta-Ruiz, and D. Comminiello. A split kernel adaptive filtering architecture for nonlinear acoustic echo cancellation. In *Proc. European Signal Processing Conference (EUSIPCO)*, pages 1768–1772. IEEE, 2016.

[WC18] D. Wang and J. Chen. Supervised speech separation based on deep learning: An overview. *IEEE/ACM Transactions Audio, Speech, and Language Processing*, 26(10):1702–1726, 2018.

[WJ11] T. S. Wada and B. Juang. Enhancement of residual echo for robust acoustic echo cancellation. *IEEE Transactions on Audio, Speech, Language Processing*, 20(1):175–189, 2011.

[WQW10] S. Wu, X. Qiu, and M. Wu. Stereo acoustic echo cancellation employing frequency-domain preprocessing and adaptive filter. *IEEE Transactions on Audio, Speech, Language Processing*, 19(3):614–623, 2010.

[WSSA18] J. Wagner, D. Schiller, A. Seiderer, and E. André. Deep learning in paralinguistic recognition tasks: Are hand-crafted features still relevant? In *Proc. Interspeech*, pages 147–151, 2018.

[WZ11] J. Wu and X. L. Zhang. Maximum margin clustering based statistical vad with multiple observation compound feature. *IEEE Signal Processing Letters*, 18(5):283–286, 2011.

[XDDL14] Y. Xu, J. Du, L. R. Dai, and C. H. Lee. A regression approach to speech enhancement based on deep neural networks. *IEEE/ACM Transactions Audio, Speech, and Language Processing*, 23(1):7–19, 2014.

[XYC22] K. Xie, Z. Yang, and J. Chen. Nonlinear residual echo suppression based on gated dual signal transformation LSTM network. In *Proc. Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, pages 1696–1701. IEEE, 2022.

[Yos01] W. A. Yost. Fundamentals of hearing: An introduction, 2001.

[YYC+21] Y. Yu, T. Yang, H. Chen, R. C. de Lamare, and Y. Li. Sparsity-aware SSAF algorithm with individual weighting factors: Performance analysis and improvements in acoustic echo cancellation. *Signal Processing*, 178:107806, 2021.

[ZGZ23] H. Zhao, Y. Gao, and Y. Zhu. Robust subband adaptive filter algorithms-based mixture correntropy and application to acoustic echo cancellation. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 31:1223–1233, 2023.

[Zhi19] H. Zhivomirov. On the development of STFT-analysis and ISTFT-synthesis routines and their practical implementation. *Technology, Education, Management, Informatics (TEM) Journal*, 8(1):56–64, 2019.

[ZL20] X. Zhou and Y. Leng. Residual acoustic echo suppression based on efficient multi-task convolutional neural network. *preprint arXiv:2009.13931*, 2020.

[ZM18] Z. Zhilu and R. S. Mert. Generalized cross entropy loss for training deep neural networks with noisy labels. In *Proc. Neural Information Processing Systems (NIPS)*, page 8792–8802, 2018.

[ZMP05] L. Zelnik-Manor and P. Perona. Self-tuning spectral clustering. In *Proc. Neural Information Processing Systems (NIPS)*, pages 1601–1608, 2005.

[ZTW19] H. Zhang, K. Tan, and D. L. Wang. Deep Learning for Joint Acoustic Echo and Noise Cancellation with Nonlinear Distortions. In *Proc. Interspeech*, pages 4255–4259, 2019.

[ZW18] H. Zhang and D. L. Wang. Deep learning for acoustic echo cancellation in noisy and double-talk scenarios. In *Proc. Interspeech*, pages 3239–3243, 2018.

[ZWMZ16] Y. Zhao, D. Wang, I. Merks, and T. Zhang. DNN-based enhancement of noisy and reverberant speech. In *Proc. International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6525–6529. IEEE, 2016.

[ZWS+22] S. Zhang, Z. Wang, J. Sun, Y. Fu, B. Tian, Q. Fu, and L. Xie. Multi-task deep residual echo suppression with echo-aware loss. In *Proc. International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 9127–9131. IEEE, 2022.

[ZZ17] S. Zhang and W. X. Zheng. Recursive adaptive sparse exponential functional link neural network for nonlinear AEC in impulsive noise environment. *IEEE Transactions on Neural Networks and Learning Systems*, 29(9):4314–4323, 2017.

אפשר לשלוט על ערכי זוג המטריקות הללו בעזרת פרמטר התכנון שהצגנו לעיל - ובכך פיתחנו מערכת מרוכזת-משתמש, שלא הייתה קיימת, המתמקדת בצרכי כל משתמש ספציפי ולא בביצועי מערכת רוחביים.

התרחיש של הד אקוסטי רב-ערוצי מתייחס לקונפיגורציה בה ישנם שני מיקרופונים ושני רמקולים הן בקצה-הקרוב והן בקצה-הרחוק, כפי שנפוץ בתרחישים מציאותיים כגון שיחות ועידה וירטואליות בין שני חדרי דיונים. תרחיש זה אינו נתמך באופן נרחב בעולם המחקרי ואינו נתמך תקציבית על ידי גופים עסקיים פרטיים וציבוריים, דבר אשר גרם לפער משמעותי בין מערכות שהוצעו עד כה לבין הדרישות הביצועיות המציאותיות. על מנת להעצים את הביצועים של קונפיגורציה רב ערוצית זו, השלכנו בהצלחה עקרונות שהצגנו עבור הורדת הד אקוסטי חד ערוצי אל העולם הרב ערוצי. ראשית, הצענו מערכת לשליטה בגודל הצעד של המסנן האדפטיבי על מנת להתמודד עם תרחישי דיבור-כפול ושינויים תכופים במסלול ההד. לאחר מכן, הראינו כי מדדי ביצועים קיימים הינם מוטים גם למקרה הרב ערוצי, והצענו הרחבה של מדדי הביצועים שהצגנו קודם - כעת לעולם הרב ערוצי. מדדי הביצועים שפיתחנו הראו גם הם קורלציה גבוהה עם ההערכה האנושית. כל העת, שמרנו גם בקונפיגורציה הרב ערוצית על סטנדרט תקין של צריכת משאבים ביחס לדרישות של מערכות תקשורת ללא-ידיים.

זיהוי פעילות דיבור הינו רכיב משמעותי בכל מערכת מבוססת-דיבור, ובפרט גם במערכות שמבצעות הורדת הד אקוסטי. על אף שהורדת פעילות דיבור הינו נושא מחקר אשר משך תשומת לב רבה בעשורים האחרונים, הצלחנו לעשות שימוש יעיל בלמידה עמוקה על מנת להשיג ביצועים מובילים בסביבות אקוסטיות אמיתיות, תוך כדי שמירה על אילוצי חישוב וזמני ריצה. שלא כמו מערכות קיימות שהזינו למודלים העמוקים את הייצוג הספקטרלי של אותות הדיבור, אנחנו זיקקנו את הייצוג הגיאומטרי האינהרנטי שמבדיל בין דיבור לתופעות אקוסטיות אחרות, ויישמנו בצורה רזה מערכת למידה עמוקה המבוססת על ארכיטקטורת קידוד-פענוח על מנת לסווג דיבור.

אנחנו הצגנו מערכת שעוקבת בזמן אמת ומשערכת את האפקט הלא לינארי שהרכיבים האלקטרוניים מפעילים על אות הדיבור מהקצה-הרחוק, והזנו את ההערכה הזו כאות ייחוס אל המערכת להורדת הד אקוסטי ליניארי. עקב כך, נוצר קשר לינארי בין אות הייחוס ובין הכניסה השניה למערכת - המיקרופון מהקצה-הקרוב. כך המערכת להורדת הד ליניארי עבדה באופן אפקטיבי והראתה ביצועים משופרים על בעיות מהעולם האמיתי, בעוד שצריכת המשאבים הייתה מועטה.

תרחישי דיבור-כפול ותרחישים בהם המסלול של ההד מהרמקול אל המיקרופון משתנה באופן תדיר קורים לעיתים קרובות בפגישות מציאותיות, וגורמות לחוסר התאמה משמעותי בין מסלול ההד האמיתי למסלול ההד המשוערך. מחקרים קיימים לא התמקדו בתרחישים ספציפיים אלו, אלא בביצועים הממוצעים של מערכות להורדת הד אקוסטי. בפועל, הביצועים על התרחישים הספציפיים המוזכרים לעיל היו ירודים. הסיבה העיקרית לכך היא כפולה - הנחות אקוסטיות לא מציאותיות, והיוריסטיקה של מספר פרמטרים. אנחנו פיתחנו מערכת שלא מצריכה הנחות אקוסטיות ואינה משמתמשת בהיוריסטיקה כלל, והצענו דרך לשלוט בגודל הצעד של המסנן האדפטיבי המוריד את ההד האקוסטי הליניארי עם עקיבה בזמן אמת. תוצאות הראו ביצועים משופרים על תרחישי דיבור-כפול, ובנוסף התכנסות מהירה בהרבה בהשוואה למתחרים כאשר מסלול ההד הליניארי משתנה.

קיום עקבות של הד שיורי היא תופעה נפוצה לאחר הפעלת מערכות להורדת הד אקוסטי לא לינארי ולינארי, עקב אי-שלמות מערכות אלו ועקב חוסר ההתאמה בין אורך המסנן המתאר את מסלול ההד הליניארי במציאות ובמערכת עצמה. על אף שמחקרים קודמים הראו ביצועים ראויים באופן ממוצע, הם גם הציעו מודלים מבוססי למידה עמוקה הצורכים משאבים רבים מדי ודורשים זמן ריצה ארוך מדי, על מנת להיחשב מתאימים למערכות פרקטיות לתקשורת ללא-ידיים. אנחנו הצגנו מערכת רזה לדיכוי הד שיורי המתאימה להטמעה במערכות ללא ידיים, וכללנו בה פרמטר תכנון חדשני המאפשר תכונה נחשקת ממערכות דיכוי הד שיורי - היכולת לשלוט באופן דינמי על ידי המשתמש על רמת דיכוי ההד השיורי לעומת רמת העיוות של האות הרצוי.

מדדי הביצועים אשר משמשים להערכת הביצועים של מערכות להורדת הד אקוסטי הם מוטים וסובלים מדו-משמעות אינהרטית על פי הגדרתן, בעיקר בתרחישי דיבור-כפול. הן מספקות ערכים זהים למצבים אקוסטיים בהם רמת ההד השיורי נמוכה ועיוות האות הרצוי גבוה, ולהיפך. אנחנו הראינו כי מטריקות אלו הן בעלות קורלציה נמוכה מאוד להערכת הביצועים הסובייקטיבית של בני אדם, והצענו שתי מטריקות אלטרנטיביות. מטריקה אחת מאפשרת הערכה של רמת דיכוי ההד השיורי, והמטריקה השניה מעריכה את רמת עיוות האות הרצוי, בתרחישי דיבור-כפול. בתרחישים מציאותיים, הראנו שלמטריקות שלנו קורלציה גבוהה עם דירוג אנושי ובנוסף הצגנו פלטפורמה דרכה

תקציר

תזה זו מציגה מערכות להורדת הד אקוסטי בקונפיגורציה חד-ערוצית ודו-ערוצית המבוססות על למידה עמוקה, בדגש על מימושים יעילים המתאימים להטמעה בפלטפורמות תקשורת ללא-ידיים מבחינת עלות חישובית ומבחינת זמני ריצה. חלק משלים של מחקר זה עוסק בזיהוי פעילות דיבור.

בשנים האחרונות, פגישות וירטואליות תפסו את מקומן של פגישות פרונטליות, בייחוד בסביבות עבודה מקצועיות. כתוצאה מכך, ארגונים מקצועיים ברחבי העולם נאלצו למצוא פתרון לאתגר של הורדת הד אקוסטי. תרחיש המייצג את הצורך במערכות להורדת הד אקוסטי הוא שיחת ועידה וירטואלית בין שני קצוות: חדר דיונים משרדי הנקרא הקצה הקרוב, וסביבה ביתית הנקראת הקצה הרחוק. בתרחיש זה, המשתתפים בקצה הרחוק סובלים לעיתים קרובות משתי תופעות - איכות שמע ירודה של הנאמר בקצה-הקרוב, ושמיעה עקבית של ההד של מה שהם עצמם אמרו. תרחיש זה ותרחישים דומים לו הינם נפוצים בעידן הוירטואלי בו אנחנו נמצאים, ועשויים לגרום לתופעות לא רצויות כגון אובדן מידע, עייפות, והיעדר פרודקטיביות. למרות מחקרים רבים שהציגו מערכות מגוונות להורדת הד אקוסטי, אלו עדיין סובלות מירידה משמעותית בביצועים כאשר הן נבחנות בתרחישים מציאותיים. לפיכך, פתרון פרקטי אשר מוריד את ההד האקוסטי ומשמר את הדיבור הרצוי הינו בעל חשיבות מחקרית ועסקית גבוהה.

תזה זו מתמקדת בעיקר בפתרון לבעיית ההד האקוסטי בקונפיגרוצייה חד-ערוצית ומציעה מערכות עבור הורדת הד אקוסטי לא לינארי, הורדת הד אקוסטי ליניארי, הורדת הד שיורי, ובחינת ביצועים אובייקטיבית עבור מערכות להורדת הד אקוסטי.

ככל שמזעור הרכיבים האלקטרוניים הולך ונהיה דומיננטי במערכות ללא-ידיים, כך גובר האפקט בו מערכות הד אקוסטי נאלצות להתמודד עם תופעות לא לינאריות. לרוב, הקשר בין הדיבור מהקצה-הרחוק לבין ההד הנקלט במיקרופון בקצה-הקרוב הוא לא לינארי. מחקרים קיימים נטו להתעלם או להמעיט בתופעה זו לעיתים קרובות, וביצועיהם על אותות בעולם האמיתי היו לא מספקים.

המחקר בוצע בהנחייתם של פרופסור ישראל כהן ודוקטור ברוך ברדוגו בפקולטה להנדסת חשמל ומחשבים.

מחבר חיבור זה מצהיר כי המחקר, כולל איסוף הנתונים, עיבודם והצגתם, התייחסות והשוואה למחקרים קודמים וכו', נעשה כולו בצורה ישרה, כמצופה ממחקר מדעי המבוצע לפי אמות המידה האתיות של העולם האקדמי. כמו כן, הדיווח על המחקר ותוצאותיו בחיבור זה נעשה בצורה ישרה ומלאה, לפי אותן אמות מידה.

תודות

ברצוני להודות למנחה הראשי שלי, פרופסור ישראל כהן, על כך שהאמין בי ועזר לי להתגבר על מכשולים רבים במהלך המסע שלי לדוקטורט. מומחיותו, הנחייתו, ותמיכתו בי - כולם עשו אותי חוקר טוב בהרבה.

ברצוני להודות גם למנחה הנוסף שלי, דוקטור ברוך ברדוגו, על כך שבאופן עקבי עזר לי כמומחה בתחום בבניית הבנה טכנית עמוקה וייחודית של המחקר שלי, בייחוד בשלביו ההתחלתיים.

לאבא שלי ראובן ז"ל, שזיהה את האהבה שלי למדע ומתמטיקה עוד בצעירותי - והשקיע ללא לאות כדי לתת לי את התנאים הטובים ביותר להצליח. בלעדייך לא הייתי מי שאני היום.

לאמא שלי אווה, שידעה לשמוח על הצלחות ביחד איתי, ולזהות רגעים קשים גם ללא מילים. בזכותך ידעתי לקבל החלטות גדולות וחשובות במהלך המסע שלי.

לאחיות הגדולות שלי יעל ומיכל, על כך שידעו לתת לי מקום לבטא את היכולות שלי כשגדלתי, לתת לי ביטחון במי שאני, ולהיות שם ברגעים החשובים.

יותר מכל, אני רוצה להודות לאשתי נוי ולילד שלי ניתאי. ברגעי שמחה וברגעי משבר, שאבתי כוח מכם כדי לצלוח את האתגרים הגדולים ביותר. ההישג הזה משמעותי הרבה יותר כי אתם כאן איתי.

ביטול הד אקוסטי
בעזרת למידה עמוקה

חיבור על מחקר

לשם מילוי חלקי של הדרישות לקבלת התואר
דוקטור לפילוסופיה

אמיר עברי מרק

הוגש לסנט הטכניון — מכון טכנולוגי לישראל
כסלו תשפ"ד חיפה נובמבר 2023

ביטול הד אקוסטי
בעזרת למידה עמוקה

אמיר עברי מרק