Improving Neural-Network-Based Speech Enhancement for Noise Reduction and Intelligibility Boosting

by

Li Haoyu

Dissertation

submitted to the Department of Informatics

in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

The Graduate University for Advanced Studies, SOKENDAI

September 2022

Acknowledgments

First and foremost, I would like to thank my supervisor Prof. Junichi Yamagishi, who accepted me as a Ph.D. student and provided me excellent research environment and continued advice. I feel so lucky to be supervised by Yamagishi-sensei and I think this three-year experience will be cherished for a lifetime.

I want to thank the examination committee members, Prof. Isao Echizen, Prof. Yi Yu, Prof. Tomoki Toda, and Prof. Hirokazu Kameoka, for their valuable comments to this thesis. I would also like to express my gratitude to Prof. Yu Tsao and Dr. Szu-Wei Fu for their discussion on the work reported in Chapter 5.

Thanks to all the members in Yamagishi-Lab for their support. Especially thanks to Dr. Xin Wang for his hard work in maintaining lab server. I also greatly appreciate Ms. Makiko Kuwahara, who used to be the lab secretary, for her kind help.

Thanks to MEXT (Ministry of Education, Culture, Sports, Science and Technology, Japan), SOKENDAI (The Graduate University for Advanced Studies), and NII (National Institute of Informatics, Japan) for financial support, without which I couldn't complete my Ph.D. course.

I also want to thank all my friends. Thanks for making my life colorful.

Finally, heartfelt thanks to my loving parents.

Abstract

Speech is the main media used by humans to communicate in daily life. However, environmental factors such as noise, reverberation, and device characteristics inevitably degrade speech signals. In application scenarios such as telephony and teleconferencing, speakers and listeners are often in different physical places with different kinds of ambient noises, which further degrades the quality of the voice communication experience. Speech enhancement techniques, which aim to improve the quality and intelligibility of speech, therefore play an indispensable role in real-world speech communication scenarios.

Depending on the usage scenario, this thesis further divides speech enhancement into two sub-tasks: (1) noise reduction (NR) and (2) intelligibility boosting (IB). Specifically, NR is designed to work on the speaker side, where the microphone receives a mixture signal containing the speaker's voice and environmental noise (e.g., additive background noise or multiplicative reverberant noise). The goal of NR is thus to suppress noise and recover clean speech from noisy mixtures. IB is designed to work on the listener side, where NR techniques cannot be used since noise sources are physically present. Instead of suppressing noise, the goal of IB is to modify speech signals only to improve their intelligibility when exposed to noisy environments.

This thesis focuses on neural network-based speech enhancement. First, this thesis tackles the limitations of noise reduction systems from three aspects: (1) incorporating dynamic neural noise embedding to improve the model's generalization to unseen noise, (2) using a neural vocoder for waveform generation to improve the speech quality, and (3) disentangling the channel (i.e., recording conditions) factor to better enhance the low-quality device recordings. Experiments

find that the proposed method well suppresses noise and produces high-quality speech waveforms.

Next, this thesis proposes a novel neural network-based system for intelligibility boosting, in which a neural surrogate model is introduced to jointly optimize multiple intelligibility and quality metrics. Experimental results indicate that this new system can lead to significant intelligibility gains and perform much better than the state-of-the-art signal processing-based baseline method.

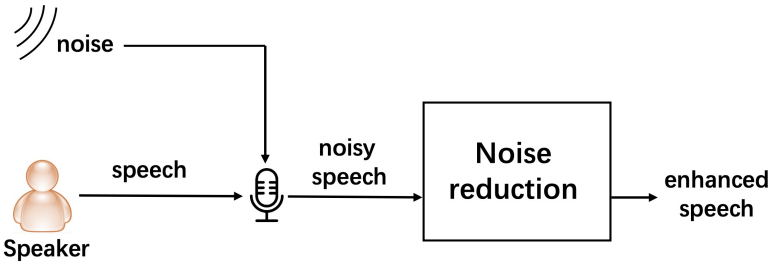
Finally, to address the full-end speech enhancement task where both speaker and listener environments are noisy, this thesis investigates a joint model integrating noise reduction with intelligibility boosting. Such a model can fully benefit from the powerful modeling capabilities of neural networks and get rid of unnecessary assumptions. Experiments show that the joint model significantly improves speech quality and intelligibility and clearly outperforms disjoint pipeline methods.

Contents

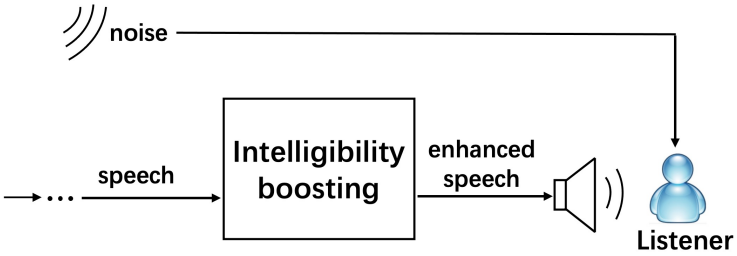
1 Introduction

Speech, an innate human capability, is the most natural form of communication for us. However, in real-world speech applications, the quality and intelligibility of speech is inevitably degraded due to adverse noisy environments.

Noise interference is present everywhere. A comprehensive analysis and measurement of speech and noise levels in real-world environments were done in [1]. According to the report, normal speech levels during face-to-face talking (assumed to be one meter) range between sound pressure levels (SPL) of 60 and 70 dB, and the level is reduced by 6 dB for every doubling of the distance. In addition, the range in noise levels varies across different places. For instance, noise levels are relatively low in a hospital, classroom, and inside the home. In these places, noise levels range between 50 and 55 dB SPL, suggesting that the effective signal-to-noise ratio (SNR) ranges between 5 and 15 dB; noise levels are particularly high, averaging about 70∼75 dB SPL, in trains and airplanes, suggesting that the SNR may approach 0 dB. Obviously, noise degrades speech quality and intelligibility, thus affecting the listening experience to a large extent.



(a) Noise reduction



(b) Intelligibility boosting

Figure 1.1: Application scenarios of speech enhancement sub-tasks.

This thesis is about speech enhancement techniques that compensate for the degradation in quality and intelligibility. Specifically, this thesis treats speech enhancement as a sequential modeling task, with the goal of outputting enhanced speech given input signals by using neural-network-based machine learning techniques.

As an introduction, this chapter briefly explains the background of this thesis in Section 1.1. An overview of this thesis is then given in Section 1.2, listing the potential issues with conventional methods and models and the proposed solutions. An outline of the thesis is given in Section 1.2.3.

1.1 Background

Speech enhancement plays an indispensable role in real-world speech communication such as telephony and teleconferencing. Depending on the usage scenario, we further divide speech enhancement into two sub-tasks: (1) noise reduction (NR) and (2) intelligibility boosting (IB).

Figure 1.1 plots the application scenarios of each sub-task. As we can see, NR

aims to suppress noise and recover enhanced clean speech from noisy input. It is most widely deployed as a front-end processing module on the speaker side, and it can be used in applications such as hearing aids [2], mobile telephony [3], and robust automatic speech recognition (ASR) [4]. In comparison, the IB system is designed to work on the listener side, with the goal to pre-process the (far-end transmitted) speech signals before playback to improve their intelligibility under background noise. The IB system can be widely used in real-world communications such as mobile telephony [5] and public-address announcements [6].

1.2 Thesis overview

This thesis focuses on improving speech enhancement with deep neural network (DNN)-based techniques. Benefiting from DNNs, speech enhancement has made impressive progress. However, there still remains a number of issues.

1.2.1 Issues to be addressed

We first identify three potential limitations for conventional NR systems:

- Issue 1: As a data-driven model, DNN's generalization ability to unseen noise (i.e., noise types not included in training) is not satisfactory.
- Issue 2: Conventional NR systems operate mainly in the spectrogram magnitude domain. Noisy phases are directly incorporated into waveform generation via inverse STFT, which degrades the speech quality [7].
- Issue 3: Most systems focus on additive or reverberant noises, but they lack consideration on device degradation (e.g., the bad frequency characteristics of common consumer-grade recording devices).

We also point out a challenging issue for the IB task:

- Issue 4: Unlike NR in which clean speech is ground-truth labels, in intelligibility boosting, there is no definition of what *perfectly intelligible speech* is. Therefore, DNN techniques cannot be directly used for supervised training.

Can we overcome this obstruction and introduce an effective DNN model into the IB task?

Last, on the basis of the above explorations, we offer an additional problem:

- Issue 5: Can we integrate noise reduction with intelligibility boosting for the scenario where noise exists in both speaker and listener environments?

1.2.2 Contribution

Regarding the issues above, we propose novel methods and conduct experiments:

- For issue 1:
  - Noise tokens, which are a set of trainable neural noise templates, are proposed to dynamically encode noise information and enable the DNN-based NR model to better handle various noise environments.
  - Experiments show that this method consistently improves the generalization ability of NR systems across different DNN architectures. It also significantly outperforms the conventional noise-aware training method [8].
- For issue 2:
  - A neural vocoder is used to generate speech waveforms instead of using conventional inverse short-time Fourier transformation (STFT), which avoids introducing noisy phases.
  - Experiments find that the vocoder-based model improves the listening quality of the generated speech.
- For issue 3:
  - Device degradation, including noise, reverberation, microphone characteristics, and audio effects, are jointly considered, which we collectively refer to as the *channel factor*.

  - An encoder-decoder neural network is proposed to automatically transform low-quality device recordings to professional high-quality ones by disentangling the channel factor via adversarial training.
  - Experiments show that the network can transform an input recording into not only one with professional studio quality but also with other arbitrary acoustic characteristics on the basis of the target channel factor designated.

- For issue 4:

  - A novel DNN-based intelligibility boosting system is proposed. To overcome the lack of ground-truth labels, a neural surrogate is introduced to approximate and mimic the behavior of speech intelligibility metrics. The system then modifies speech signals in such a way as to optimize speech metrics under the guidance of learned surrogate.
  - The new system can not only work in non-real-time mode for offline audio playback but also support practical real-time speech applications.
  - Experimental results using both objective measurements and subjective listening tests indicate that the proposed system significantly outperforms state-of-the-art baselines under various noisy and reverberant listening conditions.

- For issue 5:

  - A DNN-based joint framework integrating noise reduction with intelligibility boosting is proposed, in which the NR module first suppresses noise, and the IB module then modifies the denoised speech, i.e., the output of the NR module, to further improve speech intelligibility.
  - A noise token module, which encodes speaker-side noise information, is further inserted into the framework. The encoded noise embedding is regarded as additional noise knowledge and fed into both NR and LE modules.
  - As experiments demonstrate, the enhanced speech can be less noisy and more intelligible. The proposed framework achieves promising results

and significantly outperforms the disjoint processing methods in terms of various speech evaluation metrics.

1.2.3 Outline of thesis

The thesis is organized in accordance with the roadmap in Figure 1.2.

Chapter 2 will review the basic speech enhancement techniques, including noise reduction, intelligibility boosting, and their evaluation metrics. Chapter 3 will look into noise reduction for speech under additive noise. New models will be proposed to address **issues 1** and **2**. Chapter 4 will focus on noise reduction for low-quality device recordings (**issue 3**) and propose a model to simultaneously remove noise, reverberation, and device acoustic characteristics. Chapter 5 will investigate an intelligibility boosting task. A novel DNN-based model will be introduced to address **issue 4**. Chapter 6 will address **issue 5** by combining the noise reduction and intelligibility boosting methods explored above.

Chapter 7 will conclude this thesis and list potential directions for future improvements.

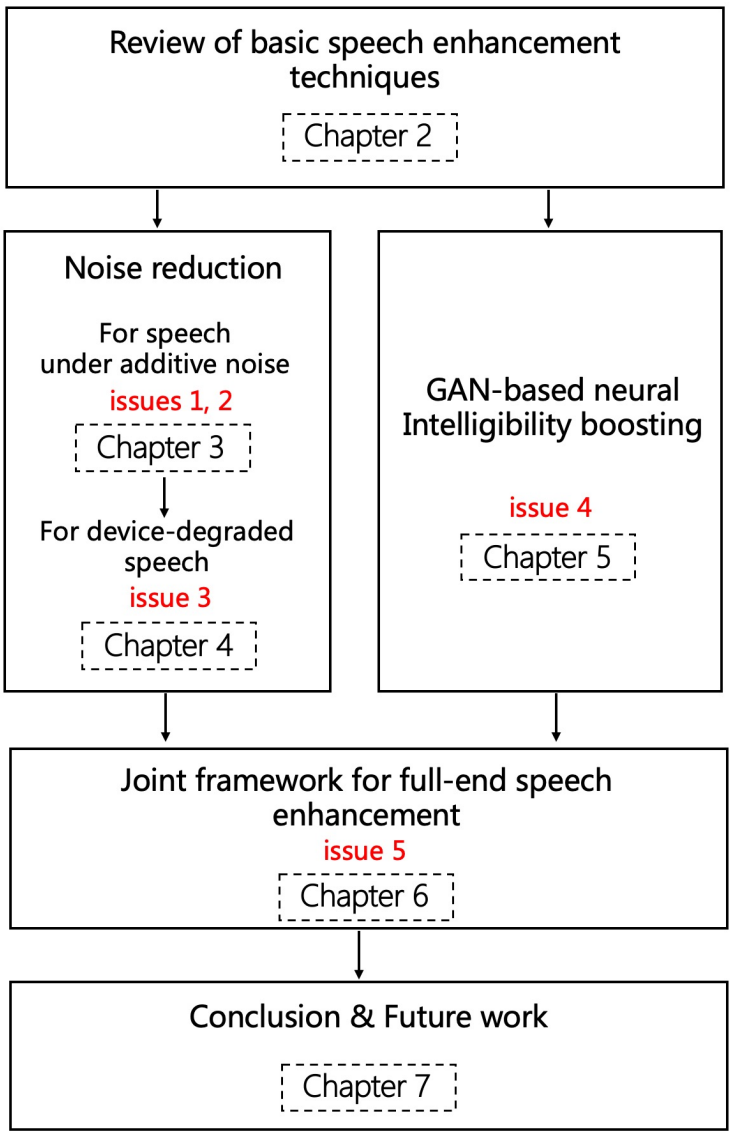


Figure 1.2: Thesis outline.

2

Basic Speech Enhancement Techniques

This chapter reviews the basic speech enhancement techniques. Section 2.1 introduces techniques for noise reduction, including traditional signal processing-based and recent neural solutions. Section 2.2 then introduces basic techniques for intelligibility boosting. It also explains the difficulties of applying neural models to this task. Last, Section 2.3 lists several metrics for evaluating the performance of speech enhancement.

2.1 Review on noise reduction

Consider the signal model in Equation (2.1), where $s(n)$ is a voice signal, $u(n)$ is environmental noise, and $x(n)$ is a noisy signal received by a microphone.

$$x(n) = s(n) + u(n) \tag{2.1}$$

The goal of noise reduction is to construct estimated clean speech $\tilde{s}(n)$ from noisy input $x(n)$.

Specifically, noise reduction problems are usually addressed in the time-frequency domain. By applying short-time Fourier transformation (STFT) to both sides of Equation (2.1), we have

$$X(m,k) = S(m,k) + U(m,k), \tag{2.2}$$

where m denotes a frame index, and k denotes the index of a frequency bin. Since the human auditory system is insensitive to phase information [9], the core step of noise reduction is to estimate a clean spectrogram $\tilde{S}(m,k)$ from a noisy observation $X(m,k)$. After that, we can apply inverse STFT (ISTFT) to $\tilde{S}(m,k)$ to reconstruct a time-domain speech signal $\tilde{s}(n)$.

2.1.1 Signal processing-based methods

In this section, we briefly review signal processing-based methods. Note that we will not cover noise estimation methods for estimating noise spectral density, although they are very essential for implementing signal processing-based noise reduction. More detailed discussions on this scope can be found in [9].

Spectral subtraction

Spectral subtraction [10] produces a clean spectrogram magnitude $|\tilde{S}(m,k)|$ by subtracting noise magnitudes from the noisy counterparts:

$$|\tilde{S}(m,k)|^p = |X(m,k)|^p - |\tilde{U}(m,k)|^p, \tag{2.3}$$

where p is an empirically chosen exponent value. The noise magnitude $|\tilde{U}(m,k)|$ can be actively estimated from non-speech segments or by using noise estimation methods [11, 12, 13, 14]. Next, ISTFT is used to reconstruct time-domain speech signals by combining $|\tilde{S}(m,k)|$ with noisy phases,

$$\tilde{s}(n) = ISTFT(|\tilde{S}(m,k)|e^{\Phi_x}), \tag{2.4}$$

where Φ_x is the phase of $X(m,k)$.

Wiener filter

The Wiener filter [15] is another classic method for noise reduction, in which the optimal (in terms of mean-square error) complex spectrogram of clean speech is given as $\tilde{S}(m,k) = \mathbb{E}[S(m,k)|X(m,k)]$. To derive the Wiener filter, three assumptions are required:

- Speech $S(m,k)$ and noise $U(m,k)$ are independent.
- $S(m,k)$ has a complex Gaussian distribution with zero mean value, i.e., $S(m,k) \sim \mathcal{N}_{\mathbb{C}}(0,\, \lambda_S(m,k))$.
- $U(m,k)$ has a complex Gaussian distribution with zero mean value, i.e., $U(m,k) \sim \mathcal{N}_{\mathbb{C}}(0,\, \lambda_U(m,k))$.

$\lambda_S(m,k)$ and $\lambda_U(m,k)$ are power spectral densities of speech and noise, respectively.

On the basis of the above assumptions, the Wiener filter can be represented by Equation (2.5).

$$\mathbb{E}[S(m,k)|X(m,k)] = \frac{\xi_{m,k}}{\xi_{m,k}+1} X(m,k), \tag{2.5}$$

where $\xi_{m,k}$ is defined as a *priori* SNR:

$$\xi_{m,k} = \frac{\lambda_S(m,k)}{\lambda_U(m,k)}. \tag{2.6}$$

A *posteriori* SNR $\gamma_{m,k}$ is also introduced in Equation (2.7)

$$\gamma_{m,k} = \frac{|X(m,k)|^2}{\lambda_U(m,k)} \approx \frac{|X(m,k)|^2}{|\tilde{U}(m,k)|^2}, \tag{2.7}$$

where $|\tilde{U}(m,k)|$ can be also estimated from non-speech segments or by using noise estimation methods [11, 12, 13]. Then, $\xi_{m,k}$ is represented by $\gamma_{m,k}$. For instance, it can be updated in a recursive way with Equation (2.8) taking a decision-directed approach [16].

$$\xi_{m,k} = \alpha \xi_{m-1,k} + (1-\alpha)\max(\gamma_{m,k} - 1, 0), \tag{2.8}$$

where α is a weighting factor and it is commonly set to 0.98.

Statistical spectral magnitude estimator

The Wiener filter is considered to be the optimal complex spectral estimator, but it is not the optimal in a spectral magnitude sense. Since the spectral magnitude plays an important role in speech perception, researchers have studied obtaining the spectral magnitude from noisy observations.

One example is the MMSE estimator [16]. It was proposed to minimize the mean-square error between the estimated and true spectral magnitudes:

$$e = \mathbb{E}\{(|\tilde{S}(m,k)| - |S(m,k)|)^2\}. \tag{2.9}$$

To address this problem, the authors made two assumptions:

- The Fourier transform coefficients, including both real and imaginary parts, have a Gaussian probability distribution with zero mean values.
- The Fourier transform coefficients are independent and, hence, uncorrelated.

On the basis of these assumptions, the optimal MMSE estimator is given by:

$$|\tilde{S}(m,k)| = \frac{\sqrt{v_{m,k}}}{\gamma_{m,k}}\Gamma(1.5)\Phi(-0.5,1;-v_{m,k})|X(m,k)|, \tag{2.10}$$

where $\Gamma(\cdot)$ denotes a gamma function, $\Phi(a,b;c)$ denotes a confluent hypergeometric function, and $v_{m,k}$ is defined as:

$$v_{m,k} = \frac{\xi_{m,k}}{1+\xi_{m,k}}\gamma_{m,k}, \tag{2.11}$$

where $\xi_{m,k}$ and $\gamma_{m,k}$ are priori and posteriori SNRs given in Equations (2.6) and (2.7), respectively. Same as Equation (2.4), the noisy phase is then combined with $|\tilde{S}(m,k)|$ to reconstruct the speech waveform $\tilde{s}(n)$.

Another example is the log-MMSE estimator [17], which extends the MMSE estimator by minimizing the mean-square error of the log-magnitude spectra:

$$e = \mathbb{E}\{(\log(|\tilde{S}(m,k)|) - \log(|S(m,k)|))^2\}. \tag{2.12}$$

Such a metric based on the squared error of log-magnitude spectra has been considered to be more suitable for subjective speech perception [17, 18]. The optimal solution for the log-MMSE estimator is then given as:

$$|\tilde{S}(m,k)| = \frac{\xi_{m,k}}{1+\xi_{m,k}} \exp\left\{\frac{1}{2}\int_{v_{m,k}}^{\infty} \frac{e^{-t}}{t} dt\right\} |X(m,k)|, \tag{2.13}$$

where $\xi_{m,k}$ and $v_{m,k}$ were defined in Equations (2.6) and (2.11), respectively.

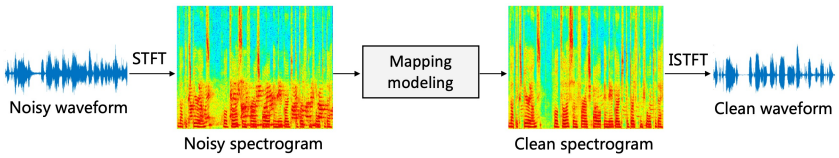
Limitations

Signal processing-based methods have been extensively studied over the decades and widely used in real-time communication systems due to their robustness and low computational cost. However, their performance is still far from satisfactory. The above-mentioned methods rely heavily on statistical noise estimation, thereby the performance severely degrades under non-stationary noise conditions (e.g., restaurant babble noise), where the noise property changes fast and constantly.

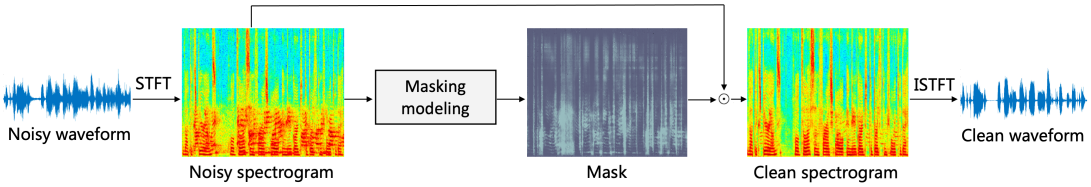
2.1.2 Neural network-based methods

Recently, neural network-based noise reduction methods have become the mainstream and outperformed the signal processing-based ones by a large margin [19, 4, 20, 21]. Neural networks have a strong modeling capability from learning from large data. Common neural architectures used in the speech processing field include: (1) the feed-forwarding neural network (FNN) [19, 20], (2) the recurrent neural network (RNN) [4, 22], (3) the convolutional neural network (CNN) [23, 24], (4) Transformer [25], and (5) various combinations of basic architecture units, e.g., CNN+RNN [26, 27] and CNN+Transformer [28]. It would be impractical to list all architecture details in such a short section. A general introduction on neural networks can be found in other literature [29].

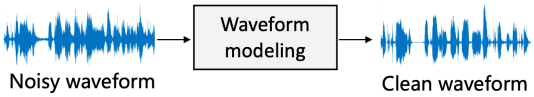
On the basis of modeling targets, we categorize neural noise reduction models into three groups. Figure 2.1 gives the overall diagrams for each modeling target.



(a) Mapping-based modeling



(b) Masking-based modeling



(c) Waveform modeling

Figure 2.1: Diagrams for different modeling targets.

Mapping-based modeling

This approach operates in the time-frequency (T-F) domain. As shown in Figure 2.1(a), it directly maps a noisy spectrogram into a clean spectral magnitude. For instance, Xu *et al.* [19] proposed using FNNs to predict a clean log power spectrogram (LPS), which is defined as follows.

$$LPS(m,k) = \log(|S(m,k)|^2) \tag{2.14}$$

The input feature for FNNs is a noisy LPS of speech. As can be seen, it is a straightforward approach exploiting the modeling capability of neural networks.

Masking-based modeling

The LPS target is an unbounded value; therefore, it is not very stable. Researchers have proposed predicting a *mask* instead of LPS [30, 31, 32]. Some widely-used masks include the ideal binary mask (IBM), ideal ratio mask (IRM), and spectral

magnitude mask (SMM):

$$IBM(m,k) = \begin{cases} 1, & if \;\; SNR(m,k) > LC \\ 0, & else \end{cases} \tag{2.15}$$

$$IRM(m,k) = \left(\frac{|S(m,k)|^2}{|S(m,k)|^2 + |U(m,k)|^2} \right)^{\beta} \tag{2.16}$$

$$SMM(m,k) = \frac{|S(m,k)|}{|X(m,k)|} \tag{2.17}$$

where IBM assigns the value 1 to a T-F unit if the SNR within that unit exceeds the local criterion (LC, commonly set to $-5 \sim 5$ dB), and otherwise 0. IRM is a ratio mask ranging from 0 to 1, and β is a hyper-parameter scaling the mask that is commonly set to 0.5. When $\beta = 1$, the IRM is similar to the Wiener filter. Unlike IBM and IRM, SMM is not bounded. However, in practice, SMM is usually clipped by an upper bound, e.g., 10. Compared with the LPS target, it was found that predicting such a mask target is more stable and easier for neural models [31]. As shown in Figure 2.1(b), an estimated clean spectrogram $\tilde{S}(m,k)$ can then be obtained by multiplying the mask with noisy input:

$$\tilde{S}(m,k) = Mask \odot X(m,k) \tag{2.18}$$

Waveform modeling

Unlike mapping-based and masking-based approaches operating in the T-F domain with ISTFT reconstruction, this approach directly models waveform samples (see Figure 2.1(c)). For example, the WaveNet [33] architecture, which has large receptive fields owing to multiple dilated convolutions, was revised and adapted to model clean waveform samples in [34, 35, 36]. Another example is TasNet [37, 38]. By replacing STFT and ISTFT with trainable neural modules, TasNet can directly output estimated clean waveforms.

Compared with T-F operations, waveform modeling avoids introducing noisy phases. It can model output waveforms in an end-to-end manner.

Limitations

Although neural noise reduction has shown impressive performance and become the mainstream approach [32], there remains a lot of room for improvement.

First, since a neural network is a typical data-driven model, it usually fails to generalize well to unseen noise that is not included in the training. Second, many noise reduction methods operate on spectrogram magnitude and by default disregard phases. Although new methods based on complex IRM [39] and waveform modeling have been proposed to overcome this problem, the quality of the generated speech is not satisfactory. Third, though the training data (i.e., pairs of clean speech and noisy counterparts) can be easily generated by artificially adding noise segments into clean speech, complicated real-world recording conditions (e.g., device-recorded speech) have not been fully considered and analyzed. In the latter chapters, we will propose improved methods to address these limitations.

2.2 Review on intelligibility boosting

Unlike noise reduction, since noise sources are physically present in the near-end listener environment, intelligibility boosting aims to modify speech signals only to improve their intelligibility when exposed to noise.

Numerous methods have been studied over the past decade (e.g., [40, 41, 42, 43, 44, 45, 46]). In particular, the 1st and 2nd Hurricane Challenges [47, 48] featured many effective methods and conducted comprehensive comparisons for each, providing remarkable reference value for researchers. In this section, we categorize intelligibility boosting methods into mianly three groups.

2.2.1 Knowledge-based approach

On the basis of an analysis of clear (intelligible) speech, two acoustic cues contributing to higher intelligibility were reported:

- Speech signals with more spectral flattening [49, 50], i.e., higher energy distributed in the mid-frequency region, are more intelligible.

- Enhancing the transient components of speech (e.g., vocalic onsets and offsets, nasal, fricatives, and stops) [51] improves the intelligibility of speech in noise conditions.

To increase intelligibility, many speech modification methods were artificially designed on the basis of expert knowledge. We list two examples.

SSDRC

As a two-step modification, SSDRC (spectral shaping and dynamic range compression) [40] first sharpens formant information and reduces spectral tilt by using pre-emphasis filters. This step was designed to redistribute more energy into the mid-frequency region. Second, it uses dynamic range compression (DRC) to decrease the loudness of the most sonorant parts of speech (vowels) and increase the loudness of the less sonorant parts like consonants. Experiments have demonstrated that SSDRC leads to significant intelligibility boosting under various noise conditions [40], achieving the top performance in Hurricane Challenge 1 [47].

ASE

Another example is a method called ASE (Automatic Sound Engineer) [41], which maximizes intelligibility through audio manipulations, including multi-band and broadband DRC, equalization, and limiting. ASE is designed by professional audio engineers. Specifically, it consists of four modifications: (1) decomposing the signal into six bands, (2) applying DRC to each band, (3) scaling bands in accordance with a power scheme, and (4) reconstructing the signal and using broadband DRC. Experiments found that ASE increases intelligibility while preserving quality well, achieving the top performance in Hurricane Challenge 2 [48].

Limitations

Although knowledge-based methods clearly improve speech intelligibility, they are dependent on domain experts' subjective experiences, thus still leaving room for improvement. Such methods also consist only of non-parametric speech modifications; therefore, they cannot adapt well to changing environments.

2.2.2 Lombard-style conversion

Speakers tend to increase their vocal effort when speaking in the presence of loud noise to enhance the intelligibility of their speech. This is known as the Lombard effect [49]. Inspired by such speech production characteristics, some methods (e.g., [52, 53, 54]) aim to convert normal speech to Lombard-style speech. To achieve speaking style conversion, most methods rely on vocoder-based analysis-and-synthesis techniques, where vocoder features are transformed to fit in the Lombard style. Depending on whether parallel data is used[1], we divide conversion methods into two categories.

Parallel learning

The straightforward approach is to transform acoustic features from normal to Lombard style using parallel data-driven mapping models. Specifically, the acoustic features are first produced by vocoder analysis. The parallel training data, i.e., the alignment of normal and Lombard speech frames, is obtained using dynamic time warping (DTW). The model is then trained to learn the mapping relationship between the normal and Lombard-style speech. Finally, the converted features are transformed to Lombard speech by vocoder synthesis.

Seshadr *et al.* [55] comprehensively compared different vocoders [GlottDNN [56], STRAIGHT [57], and pulse model in log-domain (PML) [58])] and models [Gaussian mixture model (GMM) and DNN] for parallel normal-to-Lombard mapping. According to their experiments, GlottDNN and PML stand out as the best vocoders in terms of quality and Lombardness, respectively, and DNN is the best mapping method in terms of Lombardness.

Non-parallel learning

Since parallel data is quite limited, non-parallel learning has been studied to better exploit data resources. CycleGAN is a typical framework for non-parallel speech conversion problems that uses cycle-consistent generative adversarial networks. It was used in [59, 54] and showed its effectiveness in experiments in terms of intelligibility and quality of converted speech.

[1] The talker makes utterances in both normal and Lombard styles.

Limitations

For most Lombard-style conversion methods, using parametric vocoders inevitably degrades the converted speech quality. It was also found that even natural Lombard speech could produce only very limited intelligibility gains under low SNR conditions [47]. Consequently, the performance of such Lombard-inspired methods is still far from satisfactory.

2.2.3 Metric-oriented optimization

The third group of methods was developed through the optimization of certain objective intelligibility metrics. The basic concept is to modify input speech in such a way as to maximize a target intelligibility metric. It would be impractical to list all related target metrics in this section, so we simply give several example methods with three widely-used metrics.

Optimizing speech intelligibility index

In [44], a linear filter was proposed to maximize the speech intelligibility index (SII) [60] by redistributing speech energy over time and frequency. Specifically, SII can be obtained by (1) estimating the long-term average spectra of the speech and noise within critical bands, (2) calculating the within-band SNR, clipping it between –15 and 15 dB, and normalizing the range between 0 and 1, and (3) calculating the SII as the weighted average of the normalized within-band SNRs.

By solving a constrained optimization problem, the optimal linear filter can be given in a closed-form solution. SII predictions and intelligibility listening test experiments have shown considerable intelligibility improvements with this linear filter method.

Optimizing mutual information

Kleijn *et al.* [61] proposed optimizing the mutual information (MI) rate between unmodified speech and the received (at the listener side) speech. The signal power is redistributed by multiplying the power with the gain at each frequency bands, and the gain value can be obtained by solving an optimization problem

under certain approximations. The method is simple but effective; it enhances the intelligibility of speech rendered in a noisy environment.

Optimizing glimpse proportion

Glimpse proportion (GP) [62] has also been used as an optimization target. The GP score is the percentage of time-frequency (T-F) regions in modeled auditory bands whose local SNR exceeds a certain threshold LC in dB:

$$GP = \frac{100}{MK} \sum_{m=1}^{M} \sum_{k=1}^{K} \mathcal{C}(S_{m,k} - (V_{m,k} + LC)), \tag{2.19}$$

where M and K are the numbers of time frames and frequency channels, $S_{t,f}$ and $V_{t,f}$ denote the T-F excitation in dB of speech and near-end noise at time m and frequency k and the $\mathcal{C}(\cdot)$ operator counts the number of *glimpses* that exceeds an audibility criterion. Although this metric is simple, it can well quantify the audibility of speech in the presence of noise.

Similarly, spectral weighting (in terms of T-F gains) is used to redistribute speech energy over time and frequency. However, unlike SII and MI, there is no closed-form solution for GP optimization. Therefore, numerical methods have been explored. Tang *et al.* used a genetic algorithm [63] to find the best spectral weighting, whereas Valentini-Botin *et al.* used a gradient descent algorithm. Experiments found that GP-oriented optimization achieved a significant intelligibility gain in both objective and subjective evaluations.

Limitations

Although metric optimization-based intelligibility boosting shows promising results and does not rely on expert knowledge, its performance still falls behind state-of-the-art algorithms such as SSDRC in subjective tests, as previously reported in [47]. This is because the objective metrics (e.g., SII) optimized within the above methods are relatively simple and inaccurate, i.e., they are not highly correlated with subjective intelligibility across different types of noise and other signal degradations [64]. Besides, optimizing only a single target sometimes causes sub-optimality in another metrics, therefore limiting performance.

Very recently, several advanced intelligibility metrics have been proposed and shown good results [65, 66, 67]. However, it is still difficult to find closed-form solutions for optimizing these metrics due to mathematical complexities. Though numerical methods, such as gradient descent and the genetic algorithm, can simultaneously optimize multiple complex metrics, their optimization schemes are based on offline iterative updates and are thus not suitable for real-time online applications.

2.2.4 Toward neural intelligibility boosting

Despite the impressive success of neural noise reduction (NR), DNNs have not been extensively used in intelligibility boosting (IB). One big challenge for neural IB is that there is no ground truth label that can be provided for supervised training.

Specifically, given unmodified plain speech, there is no standard that explicitly defines what perfectly intelligible speech is, and thus, no ground truth label can be prepared. In contrast, in a NR task, clean speech without mixed noise can be easily collected and regarded as a training label. In Chapter 5, we will propose a novel DNN-based solution to overcome this obstruction.

2.3 Speech evaluation metrics

In this section, we briefly introduce the speech metrics used in this thesis. These metrics are used to properly evaluate processed speech in terms of quality and intelligibility. Note that we will not explain the mechanism of these metrics and how they were designed as this is beyond the scope of this thesis.

2.3.1 Speech quality

Speech quality is highly subjective. It assesses the *naturalness* of a speech signal. There are too many factors affecting quality. In this thesis, we consider only environmental degradation, including noise, reverberation, and poor acoustic characteristics of recording device.

Objective metrics

We describe the objective quality metrics used in the thesis as follows.

- *CSIG*: This composite metric was designed in such a way as to correlate well with listening test results on signal distortion (SIG). The SIG is rated by listeners on a five-point scale [68]:
  - 5 - very natural, no degradation;
  - 4 - fairly natural, little degradation;
  - 3 - somewhat natural, somewhat degraded;
  - 2 - fairly unnatural, fairly degraded;
  - 1 - very unnatural, very degraded.
- *CBAK*: This composite metric was designed in such a way as to correlate well with listening test results on background intrusiveness (BAK). The BAK is rated by listeners on a five-point scale [68]:
  - 5 - not noticeable;
  - 4 - somewhat noticeable;
  - 3 - noticeable but not intrusive;
  - 2 - fairly conspicuous, somewhat intrusive;
  - 1 - very conspicuous, very intrusive.
- *COVL*: This composite metric was designed in such a way as to correlate well with listening test results on overall quality (OVL) considering both SIG and BAK. The OVL is rated by listeners on a five-point scale [68]: 5 - excellent; 4 - good; 3 - fair; 2 - poor; and 1 - bad.
- *PESQ*: Perceptual evaluation of speech quality (PESQ) is a metric defined in ITU-T recommendation P.862 [69] for automated assessment of speech quality. The PESQ score ranges from -0.5 to 4.5.
- *ViSQOL*: Virtual speech quality objective listener (ViSQOL) is an objective metric released by Google [70] for perceived audio quality. The ViSQOL score ranges from 1 to 5.

- *HASQI*: The hearing-aid speech quality index (HASQI) [66, 65] is a measure of speech quality originally designed for the evaluation of speech quality for those with hearing impairments. It has also been shown to be able to evaluate quality for listeners without hearing loss [71]. The HASQI score ranges from 0 to 1.

All above-mentioned metrics are intrusive (or full-reference) models, which require a clean speech signal as a reference to predict an intelligibility or quality score for distorted speech (with or without noise). For these metrics, higher scores indicate better quality.

Subjective metrics

We can also use subjective metrics. For example, we evaluate enhanced speech samples through the Mean-Opinion-Score (MOS) test. In this test, participants are instructed to rate a sample from 1 (bad) to 5 (excellent) in terms of the perceived quality. Another common test is the preference test, in which the participants listen to a pair of samples and choose the better one. Details on subjective test design will be given in the later chapters of the thesis.

2.3.2 Speech intelligibility

Speech intelligibility measures how comprehensible speech is, i.e., the content of the spoken words, under given conditions. Intelligibility is also affected by numerous factors, e.g., accent, speaking style, and environmental degradation, while we focus only on environmental degradation in this thesis. Unlike speech quality, intelligibility is not subjective and can be quantified by counting the number of words identified correctly in listening tests.

It is worth noting that speech intelligibility and quality are not synonymous terms. Speech can sometimes be highly intelligible but poor in quality, and vice versa. The relationship between intelligibility and quality is still not fully understood [9].

Objective metrics

We describe the objective intelligibility metrics used in the later chapters.

- *SIIB*: Speech intelligibility in bits (SIIB) [67] computes an estimation of the information shared between clean and distorted speech signals in bits per second. Since an SIIB score is related to the signal duration, all stimuli are either repeated or truncated to have a consistent duration of 20 seconds when using SIIB, producing scores in the range of $[0, +\infty)$.

- *HASPI*: The hearing-aid speech perception index (HASPI) [66] estimates the intelligibility loss through an analysis of cepstral correlation and auditory coherence within an auditory model. We use a modified variant proposed for its recently improved version [65], where the final score, within the range of $[0, +\infty)$, is calculated as a weighted sum of the modulation filter outputs.

- *STOI*: Short-time objective intelligibility (STOI) [72] measures intelligibility by computing the correlation between the spectra of clean and distorted speech. The STOI score ranges from 0 to 1.

- *ESTOI*: Extended STOI [73] also measures the spectral correlation between clean and distorted speech signals. Unlike STOI, ESTOI does not assume mutual independence between frequency bands, making it able to accurately predict the intelligibility of speech under temporally highly modulated noise sources. The ESTOI score also ranges from 0 to 1.

- *sEPSM*: An improved intelligibility prediction metric [74] based on the speech envelope-power spectrum model [75]. The sEPSM score is in the range of $[0, +\infty)$.

Same with quality metrics, all intelligibility metrics we used in this thesis are intrusive models.

Subjective metrics

The golden rule for assessing speech intelligibility is subjective listening tests, in which participants are instructed to listen to a sample and type the words they

can hear. The word accuracy rate can then be calculated as an intelligibility measure. Phonetically-balanced speech material, such as Harvard sentences [76], is commonly used in such listening tests. We can also conduct a preference test, in which the participants listen to a pair of samples and select the one that sounds clearer or the one that they can hear with less listening effort. We will give experimental details in the later chapters of the thesis.

3

Improved Noise Reduction for Speech under Additive Noise

In this chapter, we investigate on noise reduction for additive background noise. As mentioned in Section 2.1.2, the performance of many neural noise reduction models is limited by noise generalization capability and synthetic waveform quality. To improve it, we propose a new noise reduction framework as shown in Figure 3.1. Our framework includes two novel modules: (1) an environment-aware STFT enhancement module plugged with noise tokens (NT) and (2) a neural vocoder-based waveform generation module. The NT can dynamically capture the environment variability and thus enable the DNN to handle various environmental noises to produce STFT magnitude with higher quality. The subsequent waveform generation module can further suppress the residual noise through mel-spectrogram correction and vocoder-based waveform synthesis.

The chapter is structured as follows. Section 3.1 will introduce the details of NT, and Section 3.2 will introduce waveform generation module. Experimental

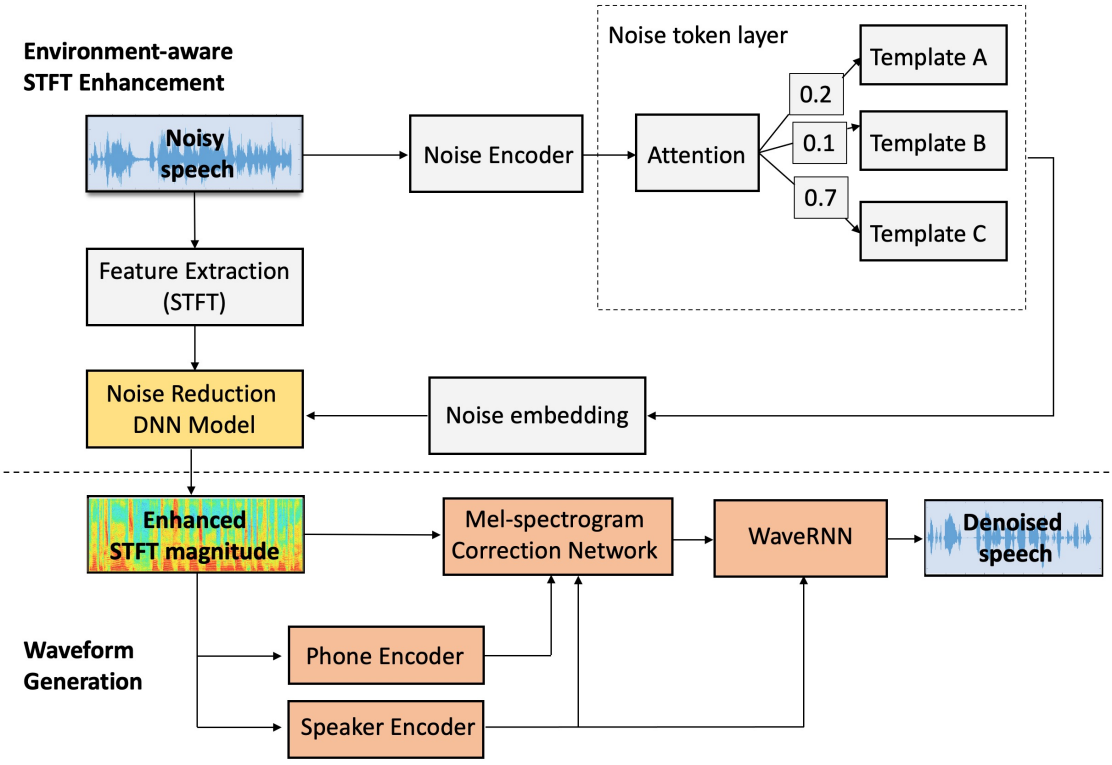


Figure 3.1: System diagram of the proposed improved noise reduction model.

setup and results are given in Section 3.3.

3.1 Noise tokens for improved generalization

This section introduces the environment-aware STFT enhancement module plugged with noise tokens, which is depicted as the upper part of Figure 3.1.

3.1.1 Background

As a data-driven approach, the DNN model is inevitably limited by its generalization ability to unseen noises. Real-world environment variability is much more complex, and it is hard for the DNN to model it sufficiently well. The mismatch between training and real-world environments leads to serious performance degradation.

To address this problem, we introduce “noise tokens” (NT) into the noise

reduction system. The NT module is inspired by recent progress [77] in expressive speech synthesis, where conceptually-similar "style tokens" were proposed to model acoustic expressiveness to control speaking style. We adapt and revise the original style tokens module to the noise reduction task. In particular, the NT module projects the noisy speech onto a *noise subspace* by assigning weights to each noise template. Noise embedding is obtained with the weighted sum of these templates and then jointly trained with the noise reduction system. By factorizing unseen noises into a linear combination of learned templates, we expect that the DNN model can handle various environments in a more efficient way.

The similar idea can be also found in [8], where the noise embedding was performed in advance by using either a conventional noise estimation or IBM-based estimation [30, 31]. In this context, however, the noise embedding generated with a separate noise estimation module might be suboptimal and inefficient. In our work, the noise embedding produced by NT is jointly optimized with the noise reduction DNN model, which facilitates flexibility and the effectiveness of the whole system. The comparison results is given in Section 3.3.

3.1.2 Environment-aware STFT enhancement with noise tokens

As shown in Figure 3.1, environment-aware STFT enhancement module produces the enhanced STFT magnitude from the noisy speech. The noise reduction part is a typical masking-based model (see Section 2.1.2). It predicts a real-value soft mask $Mask$, which is then element-wise multiplied with the noisy spectrogram magnitude $|X|$ to obtain the enhanced magnitude $|\tilde{S}| = Mask \odot |X|$. As suggested in [78], we adopt the following mean-squared error (MSE) as the loss function in the training:

$$L = \left\| |\tilde{S}|^{0.3} - |S|^{0.3} \right\|^2 + \lambda \left\| \tilde{S}^{0.3} - S^{0.3} \right\|^2 \tag{3.1}$$

where the MSE of both magnitude and complex spectra are taken into account, with a weight parameter $\lambda = 0.1$. Although our target is the enhanced magnitude, i.e., $|\tilde{S}|$, the MSE of the complex spectrogram is integrated into the loss function with the aim of somewhat reducing the phase distortion. All spectral in Equation (3.1)

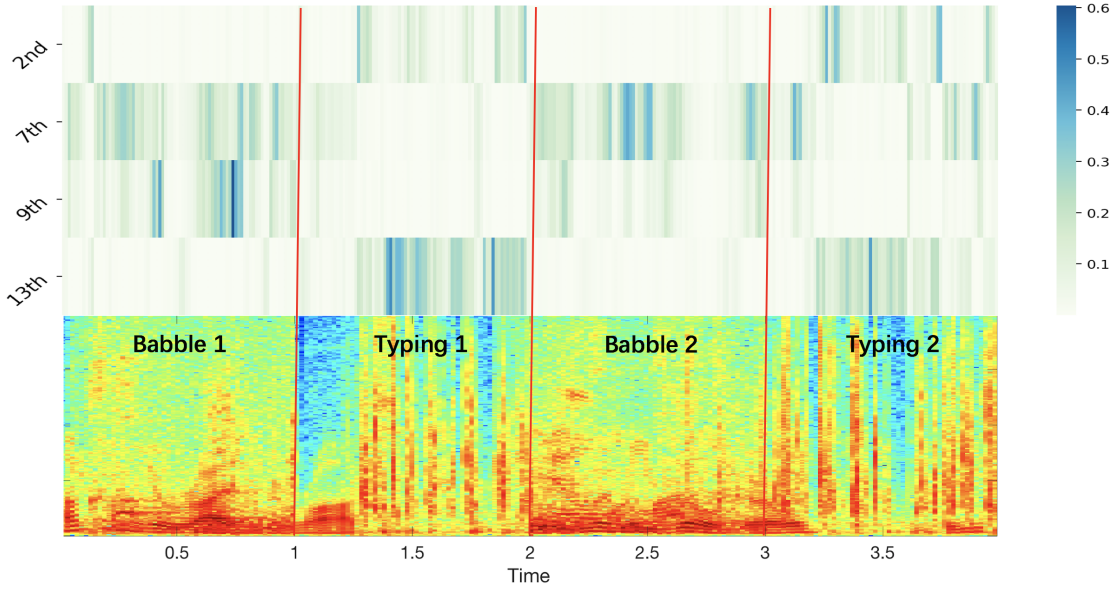


Figure 3.2: A visualization example of template weights for noisy speech where *Babble* and *Typing* noises alternatively appear. There are 16 noise templates for each of the 8 attention heads. For clear visualization, we only list four templates (2^{nd}, 7^{th}, 9^{th}, and 13^{th}) on the first attention head branch. This model is trained with BLSTM architecture on a 50-hour noisy speech data set. The detailed configurations will be discussed in Section 3.3.

are power-law compressed with a power of 0.3.

Compared to conventional masking-based baseline, noise tokens are further introduced to extract the noise embedding and inform the DNN model of environment information. The NT consists of a noise encoder and a noise token layer (NTL). The noise encoder takes as input the spectrogram magnitude of noisy speech. It is composed of 6 layers of 2-D CNN each with 3 (along the time axis)×3 (along the frequency axis) kernel, 1×2 stride, batch normalization, and ReLU activation. The output channels are set to 32, 32, 64, 64, 128, and 128, respectively. A bi-directional GRU with 128 nodes is followed by the last CNN layer, resulting in a 256-dimensional (128×2) feature for each time step. The output of the GRU is regarded as an encoded environment representation, which is then passed to the NTL. The NTL is composed of 16 trainable noise templates (tokens) and a multi-head attention module [79]. Each template has 256 dimensions, and the number of attention heads is set to 8. The representation produced by the

previous noise encoder is served as the *query* vector here, and the attention module calculates the similarity (weight) between the encoded representation and each template. The noise embedding is then generated as the weighted sum of the noise templates and fed as an additional input into the enhancement model.

Unlike [77] where only a global embedding was considered, we generate noise embedding in a dynamic frame-by-frame manner aiming to fully capture the non-stationary environment information. In addition, since noise templates are jointly trained with the whole system in an unsupervised way, we expect that the learned templates can be representative enough to model various environmental noises. Figure 3.2 gives a visualization example of learned template weights for noisy speech. We can clearly see that the 7^{th} and 9^{th} noise templates are activated during *Babble* segments, while the 2^{nd} and 13^{th} templates are active during *Typing* segments. This shows that the proposed noise tokens do capture and adapt well to varying environments.

3.2 Neural vocoder-based waveform generation

We also propose a neural vocoder-based waveform generation module (the bottom part of Figure 3.1) to generate waveform instead of using conventional inverse STFT (ISTFT). To alleviate the phase distortion and further suppress the residual noise, a mel-spectrogram correction network (MCN) is first used to predict the clean mel-spectrogram from the enhanced STFT magnitude. WaveRNN [80] vocoder is then applied to generate the final waveform with significantly better noise reduction compared to the ISTFT-based counterpart.

3.2.1 Why neural vocoder

Researchers have found that ISTFT conversion with noisy phase degrades speech quality, especially under low SNR condition [9, 81]. To address this problem, we choose to use neural vocoder.

Specifically, we choose WaveRNN for its efficiency. As shown in Figure 3.3, WaveRNN converts compact acoustic feature (e.g., mel-spectrogram) into waveform samples in an auto-regressive manner. Acoustic feature c is first up-sampled by a

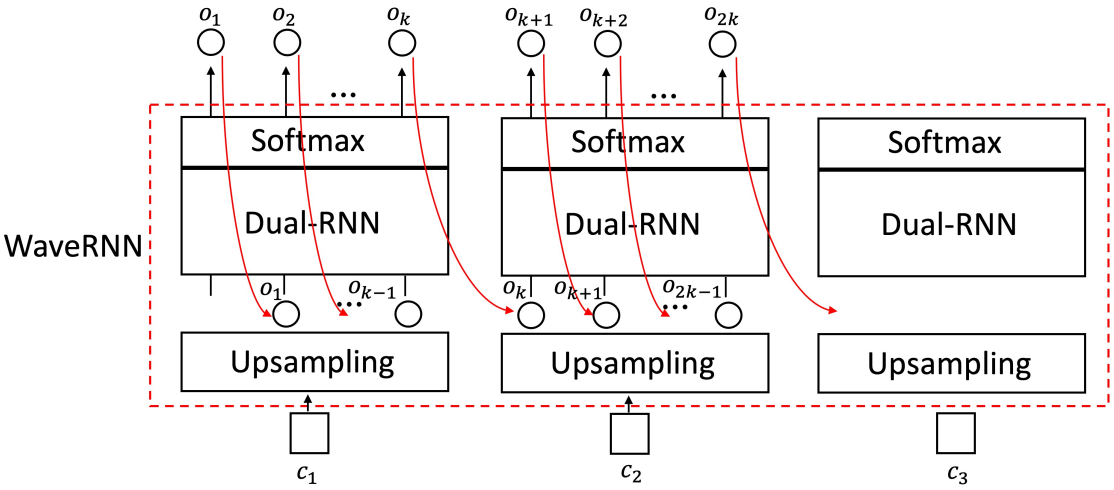


Figure 3.3: Diagram of WaveRNN vocoder.

CNN-based upsampling layer to produce temporal condition vectors for each time step. Dual-RNN with softmax layer then estimates the coarse (e.g., high 8-bits) of the sample and the fine (e.g., low 8-bits) of the sample. Finally, o is sampled from the generated distribution, and acts as a part of input for the next time step. Compared to the deterministic ISTFT, WaveRNN vocoder is a neural module that can be pre-trained with large external speech corpus. Therefore, we expect that WaveRNN is a robust neural waveform model which has better tolerance to the prediction error of spectrogram, and thus is able to generate a higher quality waveform.

3.2.2 Module details

We further elaborate the waveform generation module in this section. The MCN in Figure (3.1) aims to further suppress the residual noise by predicting the clean mel-spectrogram. Similar to [82], it is designed as an auto-regressive model that predicts one frame of target given the data generated in the previous frames. The input features are first processed by a feed-forwarding layer with 768 nodes and a BLSTM layer with 400 nodes. A unidirectional LSTM with 256 nodes then takes as input the mel-spectrogram generated in the previous frame and its previous state. Four feed-forwarding layers are successively added, each with 80 nodes, to produce the 80-dimensional mel-spectrogram of the current time step.

All feed-forwarding layers use LeakyReLU activation with slope = 0.3, except for the output layer, which uses a linear function. For better performance, we also integrate phone and speaker embeddings (both extracted from the enhanced magnitude) with the MCN to emphasize the content information and speaker characteristics of the processed utterance. Both speaker and phone embedding are extracted from the enhanced magnitude which are obtained by the former STFT enhancement module. In particular, the 64-dimensional speaker embedding comes from a pre-trained learnable dictionary encoding (LDE) [83, 84] based speaker verification system. And a phone embedding with 256 dimensions is obtained from the last bottleneck layer of a phone recognition model trained with connectionist temporal classification (CTC) loss [85]. We use the open-sourced implementations for the above speaker and phone encoders[1][2]. These two auxiliary embedding are concatenated with the enhanced STFT magnitude as the input features and help the MCN produce the mel-spectrogram with higher quality.

WaveRNN vocoder then directly synthesizes high quality waveform by avoiding introducing the noisy phase. In addition to the mel-spectrogram, the extracted speaker embedding is also fed into the vocoder as local conditions for higher synthesis quality. We use the open-sourced WaveRNN implementation[3].

3.3 Experiments

3.3.1 Data preparation

The MS-SNSD dataset [86] was used in our experiments. We selected 7 and 4 noise types to prepare the training and test sets, respectively. For the training set, we further added another 14 noise types from Nonspeech sounds database [87] to expand the diversity in noises. For the test set, the 4 selected noise types were: babble, typing, squeaky chair, airport announcements. As we only study the DNN's generalization to unseen noises, none of these 4 types were included in the training. Finally, a 50-hour training set (around 36,000 audio clips) was generated

[1] https://github.com/Diamondfan/CTC_pytorch
[2] https://github.com/jefflai108/pytorch-kaldi-neural-speaker-embeddings
[3] https://github.com/mkotha/WaveRNN

Table 3.1: Average PESQ and STOI scores with different noise embedding across three DNN architectures under test unseen noises.

Architectures	w/o embedding		with DNAT		with NT	
	PESQ	STOI	PESQ	STOI	PESQ	STOI
BLSTM	2.686	0.896	2.692	0.898	**2.858**	**0.914**
VoiceFilter	2.792	0.904	2.771	0.902	**2.907**	**0.916**
T-GSA	2.754	0.906	2.726	0.902	**2.808**	**0.912**

with 21 noise types at 5 SNR levels, i.e., -5, 0, 5, 10, 15 dB. The test set consisted of 2 hours of noisy speech, with 4 unseen noise types at 5 SNR levels, i.e., -2.5, 2.5, 7.5, 12.5, 17.5 dB. In addition, we used the VCTK corpus [88] and TIMIT database [89] to train WaveRNN vocoder and phone encoder, respectively. All audios used in our experiments were resampled at 16 kHz.

3.3.2 Pilot test I: performance analysis with noise tokens

We first examined if the performance can be improved by incorporating noise tokens (NT). We systematically test the effectiveness of NTs with three state-of-the-art noise reduction DNN architectures as follows.

- *BLSTM*: A standard model with 2 BLSTM layers and 1 fully-connected layer.
- *VoiceFilter* [26]: A CNN-BLSTM model that consists of 8 layers of 2-D CNN, followed with 1 BLSTM layer and 2 fully-connected layers.
- *T-GSA* [25]: A Transformer-based model that has 4 Transformer encoder blocks [79] with Gaussian-weighted self-attention.

Each architecture was used as the noise reduction model and trained with or without the noise embedding. Moreover, we also compared our proposed NT method with the dynamic noise aware training (DNAT) method, where the noise power spectral density (PSD) estimated by a noise tracking algorithm [13] was regarded as the noise embedding. Since we only focus on the performance

Table 3.2: Average PESQ score and its relative improvements with different training noise corpora under test unseen noises.

Noise corpus	BLSTM w/o NTs		BLSTM with NTs	
	PESQ	Relative imp.	PESQ	Relative imp.
N7	2.564	0.00%	**2.657**	**0.00%**
N12	2.639	2.94%	**2.786**	**4.86%**
N16	2.672	4.20%	**2.812**	**5.82%**
N21	2.686	4.71%	**2.858**	**7.54%**

improvement by NT, but not on the waveform generation (WG) module in this preliminary test, we simply apply ISTFT to generate the waveform with the noisy phase instead of using the WG module. The PESQ [69] and STOI [72] scores were used as objective measures. The experimental results presented in Table 3.1 showed that the proposed NT is a universal and effective technique that consistently improved the generalization capability of noise reduction across all three tested architectures. Using NT also outperformed the DNAT method, which demonstrated that the neural noise embedding is more efficient than the signal processing-based noise estimation.

3.3.3 Pilot test II: impact of noise diversity

The generalization of noise reduction systems can be improved by feeding a diverse noise corpus with more noise types. In this experiment, we analyzed the impact of noise diversity on performance. The original training noise corpus (with 21 noise types) was divided into three smaller subsets, each with {7, 12, 16} noise types. Thus, we had 4 noise corpora (denoted as N7, N12, N16, and N21) in total, and each was used to generate a 50-hour training set. Note that these four generated training sub-sets shared the same configurations, i.e., the clean speech data set, size of noisy speech data (all were 50 hours in duration), and SNR levels, while they only differed from each other in the number of noise types they were mixed with. We used the standard BLSTM architecture described in Section 3.3.2 as the noise reduction DNN model and simply applied ISTFT for waveform synthesis.

Table 3.3: Average PESQ and STOI scores with different waveform synthesis methods under test unseen noises.

Methods	PESQ	STOI
Noisy	2.021	0.833
NT-ISTFT	**2.858**	**0.914**
NT-WG	2.509	0.867

Systems with and without NTs were trained with 4 noise corpora, respectively, and then tested on the same test set. The PESQ results were given in Table 3.2. We can see that feeding more noise types into training always helped improve the performances of both systems. Compared to the system without using noise embedding, NT clearly brought higher relative improvements on PESQ with increasing noise diversity, which indicated that the proposed NT can effectively exploit multiple noises thanks to its trainable noise templates.

3.3.4 Pilot test III: initial analysis on waveform generation module

Next, we checked if the waveform generation module (the bottum part of Figure 3.1 can synthesize speech with higher quality and less residual noise. The enhanced STFT magnitude was first obtained from the noise reduction DNN model implemented with the NT and BLSTM architecture. The enhanced STFT magnitude was then be converted to the waveform by either ISTFT or the waveform generation (WG) module. We denote the systems using the above two methods as **NT-ISTFT** and **NT-WG**. Noisy speech without any processing was also compared. Table 3.3 gives the objective results and shows that the proposed WG module was much worse than the conventional ISTFT. The probable reason was that the PESQ and STOI are not designed to evaluate neural vocoder, which also explained why these measures are not typically used in the field of speech synthesis. Such unexpected results further encouraged us to conduct the following subjective listening tests.

3.3.5 Subjective listening tests

We conducted crowdsourced listening tests to comprehensively evaluate different systems. Since the WG module can be directly applied to the noisy speech to generate waveform, it was also included as a tested system. We summarize the notations for the evaluated systems in the listening tests:

- **Baseline**: BLSTM model without noise tokens. ISTFT was used to generate waveform.
- **NT-ISTFT**: BLSTM model with noise tokens. ISTFT was used to generate waveform.
- **NT-WG**: BLSTM model with noise tokens. Waveform generation module was then used to generate waveform.
- **WG**: Waveform generation module was directly applied to the input noisy STFT magnitude to generate waveform.
- **Clean**: Clean speech without noise.
- **Noisy**: Noisy speech without any processing.

We chose 96 files from the test set for each system[4]. Subjects were asked to rate the speech quality, noise suppression, and the overall performance of the anonymized file from 1-5 for the mean opinion score (MOS). For reference, the clean and noisy versions of each file were also provided to subjects before rating. Each file was rated ten times in order to avoid human bias, and 521 subjects participated. To reduce the burden on the subjects, the test files that were more than 12 seconds in duration were manually split into smaller segments of at most 5 seconds.

The subjective results are shown in Figure 3.4. The Mann-Whitney U test [90] reveals that the **NT-ISTFT** system outperformed **Baseline** in all three scores with p-values all lower than 0.005, which demonstrates the effectiveness of noise tokens. Compared to **NT-ISTFT**, **NT-WG** showed significantly higher performances, especially on the noise suppression score. This indicated that the waveform generation module successfully improved the quality and further suppressed the residual noise. Furthermore, **NT-WG** outperformed **WG**, which

[4]Audio samples of the tested files are available at: https://nii-yamagishilab.github.io/samples-NTs/

means our proposed two-step noise reduction framework, where the waveform generation module was used as a post-processor, was better than the method that applied only the waveform generation module to the noisy input. Last, examples of enhanced spectrograms of the evaluated systems are given in Figure 3.5. We can clearly see that residual noise was more suppressed for the **NT-ISTFT** system than for **Baseline**. Compared to the clean reference, some acoustic artifacts in the middle frequency part were introduced by the vocoder-based **NT-WG** and **WG** systems. However, these artifacts did not affect human perception. Also, we can find that the residual noise in **NT-ISTFT** was further removed after using a post-processing waveform generation module (**NT-WG**).

From the listening test results, we can see that the improved method that incorporates noise tokens and waveform generation module, i.e., **NT-WG**, performed best in all three aspects: speech quality, noise suppression and overall performance. Interestingly, it showed bad results in terms of objective scores: PESQ and STOI. This indicates that the acoustic artifacts introduced by WaveRNN vocoder degraded the objective performance but did not affect the human subjective evaluations. However, we still find that the performance of the WG module was not perfectly stable. This problem occurred with a limited number of cases, but some vocoder-generated samples (from **NT-WG** and **WG** systems) were seriously distorted and thus had very bad quality, which also explained the unsatisfactory lower whisker of the **NT-WG** system.

To improve the robustness of the vocoder-generated speech is our future work. In addition, we find that the raw **WG** system can even outperform **NT-ISTFT**, which indicates the waveform generation module itself can be used as a powerful noise reduction model. We plan to further study the waveform generation module and integrate it with the noise tokens.

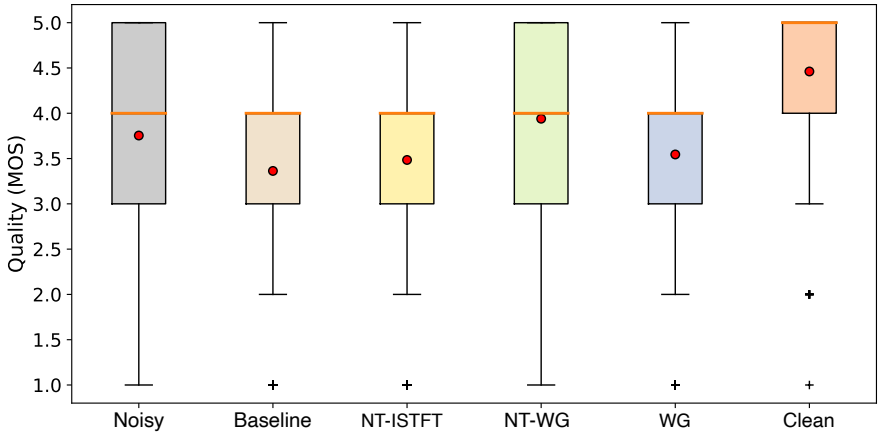
3.4 Summary

This chapter focuses on **issue 1: Limited noise generalization capability of DNN-based noise reduction model** and **issue 2: Speech quality degradation caused by ISTFT with noisy phase**. The proposed improved framework includes noise tokens and WaveRNN-based waveform generation

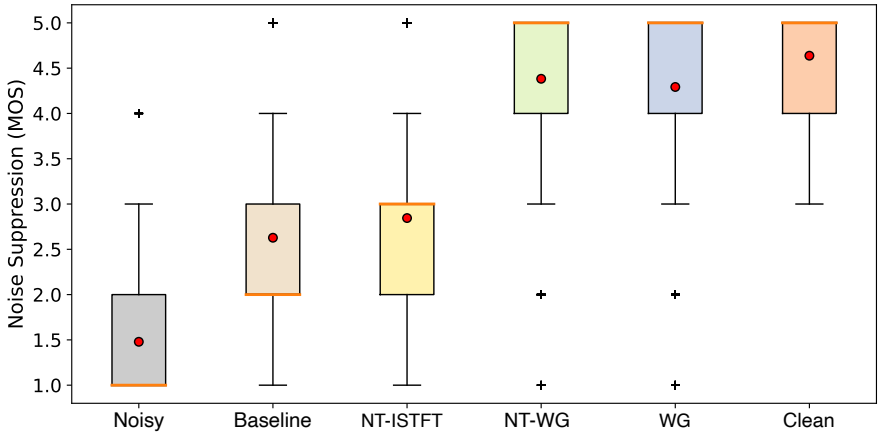
module.

The neural noise embedding, that is made up of trainable noise templates, can dynamically capture the environment information and thus enriches the DNN's generalization. Experimental results show that the noise token module is effective across various DNN architectures and has higher performance growth with increasing noise diversity. Moreover, experiments have found that WaveRNN-based vocoder synthesizes the waveform with higher quality. Subjective listening tests also show that the residual noise can be significantly reduced by the proposed waveform generation module.

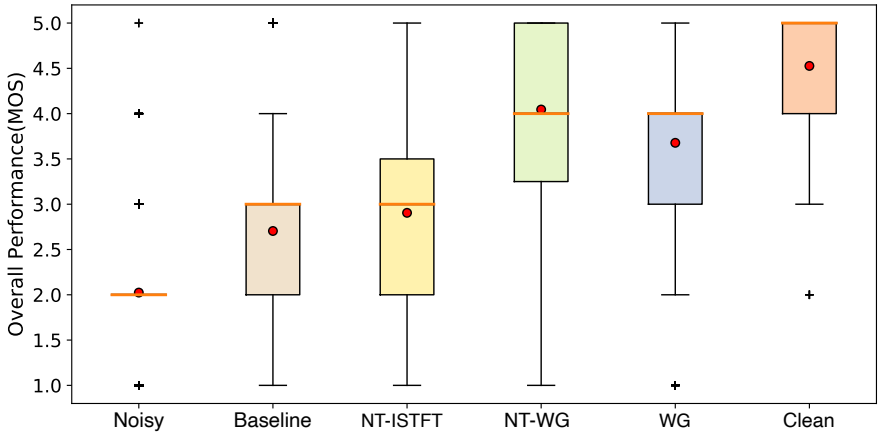
Although the improved framework achieved good performance, it was only evaluated in limited situations, i.e., background additive noise. In the next chapter, we will extend it to adapt to the more complicated real-world noise scenarios, which may take into account not only additive noise but also reverberation and poor acoustic characteristics of recording devices.



(a) Results on speech quality

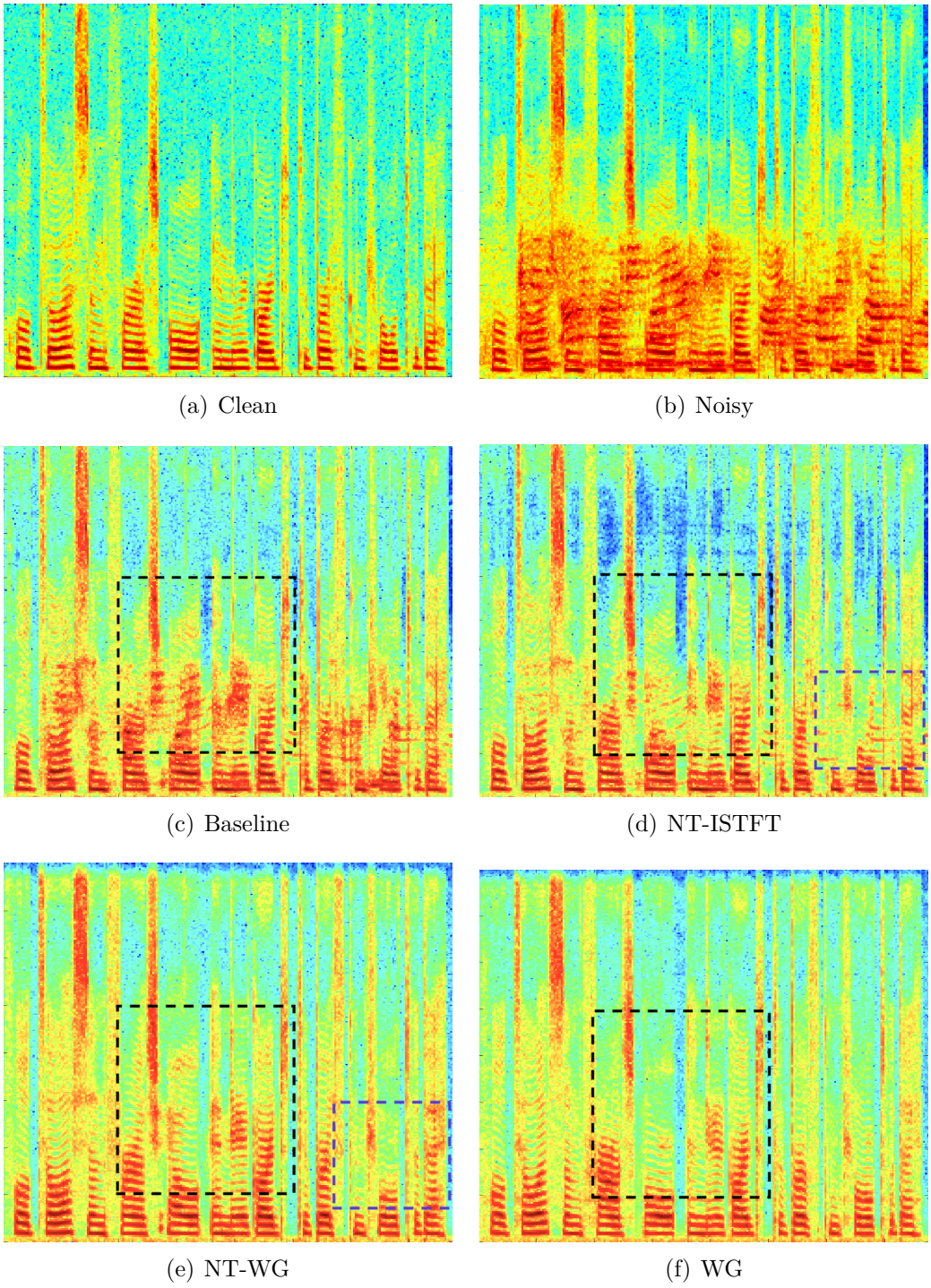


(b) Results on noise suppression



(c) Results on overall performance

Figure 3.4: Box plots on speech quality, noise suppression and overall performance. Red dots represent mean score.



(a) Clean (b) Noisy

(c) Baseline (d) NT-ISTFT

(e) NT-WG (f) WG

Figure 3.5: Examples of spectrograms under airport announcement noise at 2.5 dB for different systems: (a) Clean, (b) Noisy, (c) Baseline, (d) NT-ISTFT, (e) NT-WG, (f) WG.

4

Improved Noise Reduction for Device-degraded Speech

Chapter 3 focuses on noise reduction under additive noise. However, the speech signal in our real world is degraded by not only additive noise but also many other factors, e.g., reverberation and bad microphone response.

In this chapter, we consider a common real-world application scenario – enhancing device-degraded speech that is recorded by consumer-grade device in uncontrolled environment. Recording condition, including noise, reverberation, microphone characteristics, and audio effects, will be jointly considered. Our final goal is to automatically transform such low-quality speech into high-quality speech.

Specifically, we propose an encoder-decoder neural network for this task. To address the variability of recording condition, we first filter out the acoustic characteristics from the original input audio using the encoder network with adversarial training. Next, we extract the recording factor from a reference audio by using a channel token module, which is conceptually similar to the

noise token we introduced in Section 3.1. Conditioned on this disentangled factor, an auto-regressive decoder is then used to predict the target-environment mel-spectrogram. Finally, following the waveform generation module described in Section 3.2, we also apply WaveRNN vocoder to synthesize the speech waveform. Experimental results show that the proposed system can generate a professional high-quality speech waveform when setting high-quality audio as the reference. It also improves noise reduction performance compared with several state-of-the-art baseline systems.

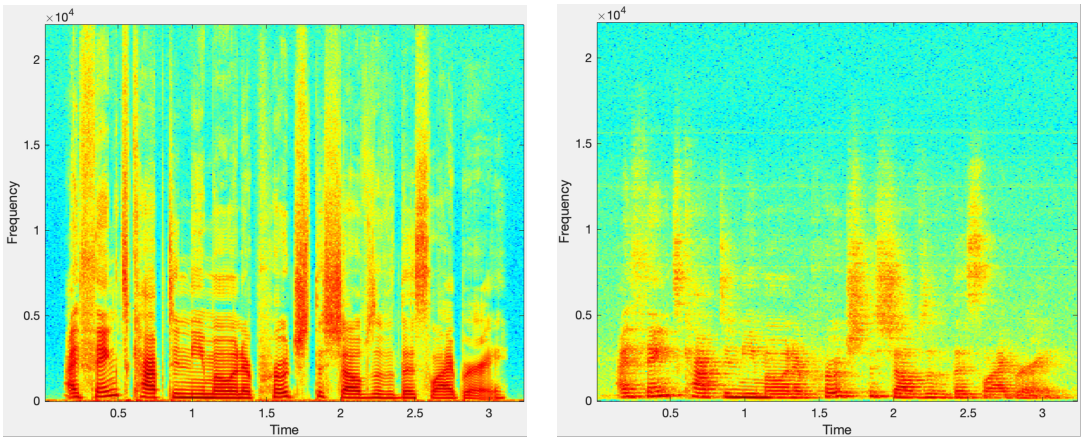
This chapter is organized as follows. Section 4.1 will first introduce the related research background. Section 4.2 will describe our proposed system and give details of each network component. Experimental setup and results are presented in Section 4.3.

4.1 Introduction to device-degraded speech

A large and growing amount of speech content in real-life scenarios is being recorded on consumer-grade devices (e.g., smartphones and laptops) [91] in uncontrolled environments (e.g., homes and offices), where environmental noise, reverberation, and distortion of microphone frequency response degrade the quality of the speech. We refer to speech that has been collected under such uncontrolled recording conditions as *device-degraded* speech. Besides, recording conditions, including noise, reverberation, microphone characteristics, and audio effects, are jointly considered, which we collectively refer to as the *channel factor*.

Figure 4.1 plots the spectrograms of a high-quality recording and its device-degraded version recorded by iPad in an office. In addition to additive background noise, we can see that reverberation (i.e., multiplicative noise) also degrades speech severely, which results in stretched spectrogram and makes speech very muffled and unclear. Besides, the high-frequency components in device-degraded speech (see Figure 4.1(b)) are heavily muffled due to the bad frequency response of recording microphone, making the sound quality worse.

Most existing methods, including both signal processing-based and neural noise reduction, are typically developed for a single application scenario, such as denoising (e.g. [17, 19]), de-reverberation (e.g. [92, 93]), or audio effect adaptation



(a) High-quality recording (b) Device-degraded recording

Figure 4.1: Examples of spectrograms for: (a) a high-quality recording and (b) a device-degraded recording recorded by iPad in an office.

(e.g. equalization [94, 95]). Although one can combine denoising, de-reverberation, and equalization methods to sequentially address each sub-problem, Mysore *et al.* [91] pointed out that such intuitive combination would degrade speech quality due to undesired synergy between processes. For example, a sound equalizer might amplify background noise by wrongly amplifying noisy-frequency components, which causes conflict with the speech denoising process. The performance of such pipeline methods is thus still far from satisfactory.

4.2 Encoder-decoder-based noise reduction

We propose an encoder-decoder network to enhance the device-degraded speech. The framework diagram is illustrated in Figure 4.2. It consists of three main components: an encoder, a channel modeling (CM) network, and a decoder. In addition, a WaveRNN vocoder works separately as the waveform synthesis module.

4.2.1 Component details

In this following section, we will explain each component in details.

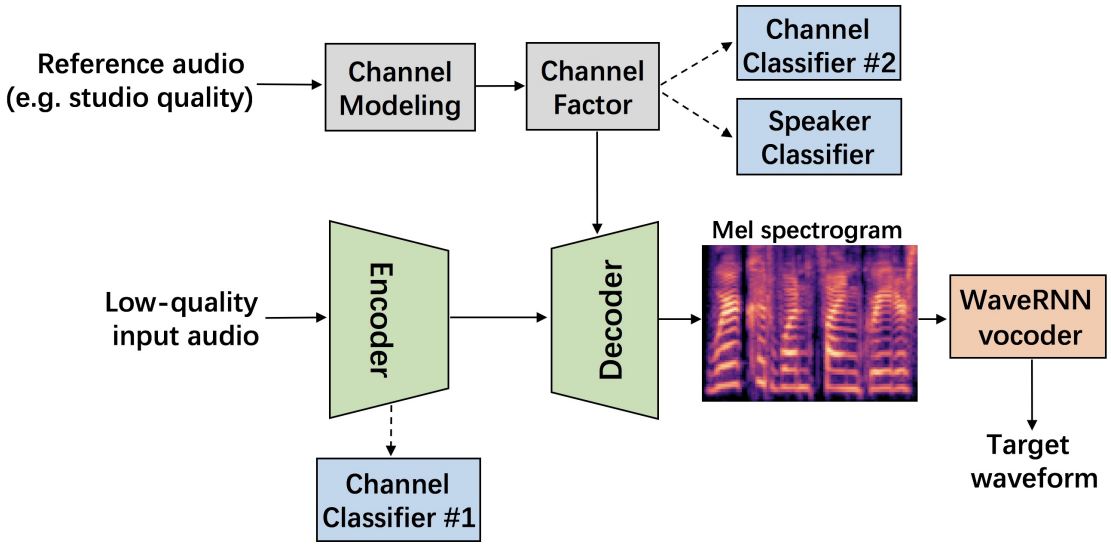


Figure 4.2: Overall diagram of proposed framework.

Encoder

The encoder is designed to filter out the channel characteristics from input audio. More specifically, the input audio is first transformed to the spectrogram magnitude via STFT, and then passed to the encoder to produce the channel-invariant features. The encoder consists of a six-layer 2D CNN, each layer with batch normalization, ReLU activation, and zero paddings, and one BLSTM layer. The details of the encoder's parameters are listed in Table 4.1.

To encourage the encoder to produce channel-invariant features, inspired by [96] and [97], we introduce a **channel classifier (#1)** as a discriminator for adversarial training. It consists of one uni-directional LSTM (ULSTM) layer with 400 nodes and one fully-connected layer with a softmax layer, which predicts the channel type (recording condition) of the input audio. In the training stage, this classifier is optimized to accurately predict the channel type by minimizing the cross-entropy classification loss. On the other hand, the encoder is optimized oppositely to maximize the classification loss to prevent the produced features from encoding channel information. This adversarial training encourages the encoder to filter out the channel information from its input.

Table 4.1: Parameters of encoder. Kernel shape of 2D CNN layers is represented as [kernel size tuple, stride tuple, output channels]. T and F denote the number of frames and frequency bins, respectively.

Layer	Input shape	Kernel shape / Nodes	Output shape
CNN 1	(T, F)	[(1, 7), (1, 1), 64]	(64, T, F)
CNN 2	(64, T, F)	[(7, 1), (1, 1), 64]	(64, T, F)
CNN 3	(64, T, F)	[(5, 5), (1, 1), 64]	(64, T, F)
CNN 4	(64, T, F)	[(5, 5), (1, 2), 64]	(64, T, F // 2)
CNN 5	(64, T, F // 2)	[(5, 5), (1, 2), 64]	(64, T, F // 4)
CNN 6	(64, T, F // 4)	[(1, 1), (1, 1), 8]	(8, T, F // 4)
BLSTM	(T, F × 2)	256	(T, 512)

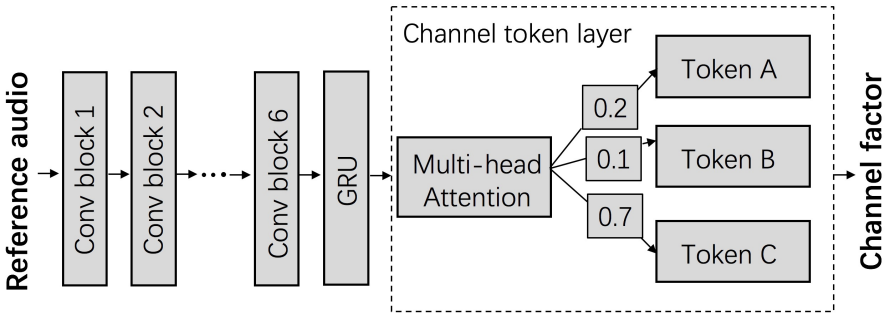


Figure 4.3: Detailed structure of CM network.

Channel Modeling

The channel modeling (CM) network explicitly extracts the channel factor from the reference audio. Its structure is shown in Figure 4.3. As can be seen, CM network is revised from noise token module described in Section 3.1. While the difference lies in the modeling target is not the single "noise" but the complicated "channel", i.e., a joint factor of noise, reverberation and microphone response.

Instead of using a one-hot code, the channel factor can be automatically encoded as a neural code from the reference audio, which enables the system to deal with the unseen channel condition and unlabelled reference audio. Moreover, the CM network can be jointly optimized with other neural components, which further provides better results.

Similar to noise token, the CM network takes as input the spectrogram magnitude computed from the reference. It consists of a six-layer 2D CNN each with a 5×5 kernel, 2×2 stride, batch normalization, and ReLU activation. The output channels are set to 32, 32, 64, 64, 128, and 128, respectively. A uni-directional gated recurrent unit (GRU) layer with 128 nodes follows the last CNN layer, producing an intermediate feature. Next, a channel token layer is added, which consists of 12 trainable channel tokens and a multi-head attention module [79]. Specifically, each token has 256 dimensions, and the number of attention heads is set to 8. The intermediate feature output by the GRU layer is fed to the channel token layer and serves as the *query* vector, then the attention module calculates the similarity (weight) between the query and each token. Finally, the channel factor (vector) is formed as the weighted sum of these channel tokens.

To better disentangle channel and speaker identities from the reference audio, we further introduce two additional classifiers, **channel classifier (#2)** and **speaker classifier**. Both are feed-forwarding networks with one 256-node hidden layer followed by a softmax layer to predict the channel type or speaker identity. Note that different from channel classifier (#1) used in the encoder, classifier (#2) here encourages the channel factor to be more informative about channel information. While speaker classifier still serves as the adversarial discriminator with the aim of filtering out the speaker information from the extracted channel factor.

Decoder

The auto-regressive decoder shown in Figure 4.4 is used to produce the target-environment mel-spectrogram, which is supposed to share similar channel characteristics to those of the reference audio. The extracted channel factor is first repeatedly concatenated to the encoder output in every time frame. The resulting concatenated features are processed by a BLSTM layer with 256 nodes, and then passed to a ULSTM layer with 512 nodes. Four feed-forwarding layers are sequentially added, each with 80 nodes, to produce the 80-dimensional mel-spectrogram. Similar to Tacotron2 [98], we add a 2-layer Pre-Net each with

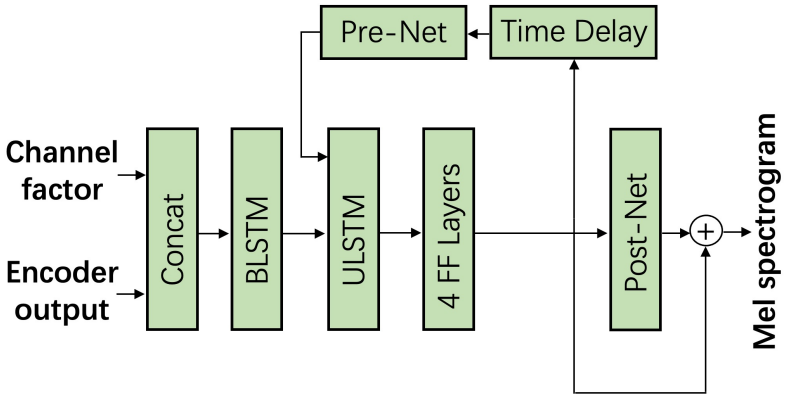


Figure 4.4: Decoder structure. Concat and FF denote concatenation operation and feed-forwarding layer, respectively.

256 nodes for the auto-regressive process. The produced mel-spectrogram from the previous time step is processed through Pre-Net and fed into the ULSTM layer for the prediction of the current time step. A five-layer convolutional Post-Net module used in Tacotron2 is also borrowed to predict the mel-spectrogram residual to improve the overall reconstruction.

WaveRNN vocoder

Last, following the success of vocoder-based waveform generation module used in Section 3, we also choose a WaveRNN vocoder to generate the waveform from the mel-spectrogram. Specifically, we use a speaker-independent WaveRNN, which effectively generalizes to unseen speakers. According to the subjective evaluation results reported in Section 3.3.5, such a neural waveform model can generate speech with high quality.

4.2.2 Training objective

To state the training objective of our proposed framework, we review each component shown in Figure 4.2 and use the following definitions:

$$\boldsymbol{z}_e^{1:T_i} = \boldsymbol{\Phi}_e(\boldsymbol{o}_{in}^{1:T_i}), \tag{4.1}$$

$$\boldsymbol{z}_c = \boldsymbol{\Phi}_c(\boldsymbol{o}_{ref}^{1:T_r}), \tag{4.2}$$

$$\hat{\boldsymbol{m}}^{1:T_i} = \boldsymbol{\Phi}_d(\boldsymbol{z}_e^{1:T_i}, \boldsymbol{z}_c) \tag{4.3}$$

The encoder $\boldsymbol{\Phi}_e$ encodes the input spectrogram of T_i frame length, $\boldsymbol{o}_{in}^{1:T_i} = \{\boldsymbol{o}_{in}^1, \cdots, \boldsymbol{o}_{in}^{T_i}\}$, into a latent sequence $\boldsymbol{z}_e^{1:T_i} = \{\boldsymbol{z}_e^1, \cdots, \boldsymbol{z}_e^{T_i}\}$. The CM network $\boldsymbol{\Phi}_c$ extracts the channel factor (vector) $\boldsymbol{z}_c$ from the reference spectrogram of T_r frame length, $\boldsymbol{o}_{ref}^{1:T_r} = \{\boldsymbol{o}_{ref}^1, \cdots, \boldsymbol{o}_{ref}^{T_r}\}$. Decoder $\boldsymbol{\Phi}_d$ takes as inputs $\boldsymbol{z}_e^{1:T_i}$ and $\boldsymbol{z}_c$ and predicts the mel-spectrogram $\hat{\boldsymbol{m}}^{1:T_i} = \{\hat{\boldsymbol{m}}^1, \cdots, \hat{\boldsymbol{m}}^{T_i}\}$. We jointly optimize three modules, $\boldsymbol{\Phi}_e$, $\boldsymbol{\Phi}_c$, and $\boldsymbol{\Phi}_d$, to minimize the mean-square error (MSE) between the predicted mel-spectrogram $\hat{\boldsymbol{m}}^{1:T_i}$ and ground truth $\boldsymbol{m}^{1:T_i}$, as formulated in Equation (4.4):

$$\mathcal{L}_{MSE}(\boldsymbol{\Phi}_e, \boldsymbol{\Phi}_c, \boldsymbol{\Phi}_d) = \frac{1}{T_i} \sum_{j=1}^{T_i} \|\hat{\boldsymbol{m}}^j - \boldsymbol{m}^j\|_2^2. \tag{4.4}$$

In addition to the main MSE objective, we add the following three objectives:

$$\mathcal{L}_{enc\_ch}(\boldsymbol{\Phi}_e, \boldsymbol{D}_{c1}) = CE(\boldsymbol{D}_{c1}(\boldsymbol{z}_e^{1:T_i}), \boldsymbol{c}_{in}), \tag{4.5}$$

$$\mathcal{L}_{cm\_ch}(\boldsymbol{\Phi}_c, \boldsymbol{D}_{c2}) = CE(\boldsymbol{D}_{c2}(\boldsymbol{z}_c), \boldsymbol{c}_{ref}), \tag{4.6}$$

$$\mathcal{L}_{cm\_spk}(\boldsymbol{\Phi}_c, \boldsymbol{D}_s) = CE(\boldsymbol{D}_s(\boldsymbol{z}_c), \boldsymbol{s}_{ref}) \tag{4.7}$$

where CE denotes the cross-entropy loss, and $\boldsymbol{D}_{c1}$, $\boldsymbol{D}_{c2}$, and $\boldsymbol{D}_s$ denote channel classifier #1, #2, and the speaker classifier, respectively. The channel types of the input and the reference are represented as one-hot labels, i.e., $\boldsymbol{c}_{in}$ and $\boldsymbol{c}_{ref}$, and the speaker label of the reference is denoted as $\boldsymbol{s}_{ref}$. As explained in previous sections, $\mathcal{L}_{enc\_ch}$ is used as the adversarial training objective to filter out the channel information from the encoder output $\boldsymbol{z}_e^{1:T_i}$. We also use $\mathcal{L}_{cm\_ch}$ as an auxiliary objective and $\mathcal{L}_{cm\_spk}$ as an adversarial objective, to encourage the channel factor $\boldsymbol{z}_c$ to encode more channel information but less speaker information. In the training stage, the neural components (i.e. $\boldsymbol{\Phi}_e$, $\boldsymbol{\Phi}_c$, and $\boldsymbol{\Phi}_d$) and classifiers (i.e. $\boldsymbol{D}_{c1}$, $\boldsymbol{D}_{c2}$, and $\boldsymbol{D}_s$) are optimized alternatively. At one training step, we optimize three classifiers individually by minimizing their corresponding cross-entropy objectives, which are $\mathcal{L}_{enc\_ch}$, $\mathcal{L}_{cm\_ch}$, and $\mathcal{L}_{cm\_spk}$. At the next training step, we fix the classifiers and jointly optimize all three neural components with the following training objective:

$$\mathcal{L} = \mathcal{L}_{MSE} + \alpha * \mathcal{L}_{cm\_ch} - \beta * \mathcal{L}_{enc\_ch} - \gamma * \mathcal{L}_{cm\_spk}, \tag{4.8}$$

where α, β, and γ are hyper-parameters controlling the weights of different sub-objectives.

4.3 Experiments

4.3.1 Data preparation

The DAPS (device and produced speech) dataset [91] was used in our experiments. It provides aligned recordings of high-quality speech[1] and a number of versions of low-quality speech[2], which are affected by noise, reverberation, and microphone response. Specifically, it consists of 20 speakers (10 female and 10 male) reading 5 excerpts each from public domain stories.

To prepare the training set, we selected 4 of the 5 excerpts narrated by 18 of the 20 speakers under 7 of the 10 recording conditions then split the corresponding recordings into shorter segments, which resulted in 23,555 audio clips. The remaining 1 excerpt, 2 speakers (1 female and 1 male), and 3 conditions were used to form the test set, which resulted in 228 audio clips. Thus, all the content, speakers, and recording conditions of the tested speech were unseen to the training set. The three tested real-world recording conditions were: (1) *ipad_livingroom*, recording was done by an iPad Air in a living room; (2) *ipadflat_office*, recording was done by an iPad Air placed flat in an office; and (3) *iphone_bedroom*, recording was done by an iPhone 5S in a bedroom.

4.3.2 Implementation details

All audios were resampled at 16kHz. We used STFT to compute the spectrogram with a Hanning window size of 50 ms and a hop size of 12.5 ms, and the spectrogram was power-law compressed [78] with a power of 0.3. For WaveRNN vocoder, we

[1]High-quality speech recordings were collected in a studio environment. Several audio effects were further applied to these recordings by professional sound engineers to make them sound more aesthetically pleasing.

[2]Low-quality speech recordings were produced by replaying high-quality audio through a professional loudspeaker and re-recording them with different consumer devices in different real-world acoustic environments.

used a public speaker-independent model[3], which was pretrained sufficiently with more than 900 speakers selected from the LibriTTS corpus [99]. We slightly fine-tuned the model, with the high-quality studio recordings in the training set, to make the model adapt to the studio audio effect. Note that the speakers and content of the tested recordings were still unseen to the fine-tuned WaveRNN vocoder.

Although the primary target of this work is to enhance low-quality recordings, we implemented audio effect transfer, e.g., transferring the iPhone recording in the bedroom to the iPad recording in the office, within one unified system. As shown in Figure 4.2, the decoder can predict not only the mel-spectrogram in studio quality but also that under other recording conditions, depending on the reference audio[4]. This architecture enables us to augment training data with diverse combinations of input and reference pairs. Since the system learns to disentangle the channel factor and adapt to various recording conditions, we expect that it can reduce overfitting and improve overall performance. Therefore, each audio clip under 7 training recording conditions was combined with 3 different types of references: one high-quality recording (as primary training target) and two recordings that were randomly selected from the other 6 conditions. This extended the original training set and resulted in a total of 70,665 (23,555 $\times$ 3) training examples. For the test set, we set the high-quality recording only as the reference since our ultimate target is to examine if the low-quality input can be enhanced. The Adam optimizer [100] was used for training, with learning rates of 0.0001 and 0.0002 for the model and its classifiers, respectively. Hyper-parameters α, β, and γ in Equation (4.8) were set to 1.0, 0.2, and 0.05, respectively.

4.3.3 Evaluated systems

We conducted an ablation study on the proposed system. Several noise reduction baselines were also re-implemented, making a total of seven systems compared in the experiments. We describe and notate each system as follows:

[3] https://github.com/erogol/WaveRNN

[4] The reference audio was randomly selected, which did not correlate with the input recording in terms of both speaker and content.

- **ED**: A simplified version of our proposed system that is composed only of encoder and decoder modules. The decoder only predicts the high-quality mel-spectrogram as its prediction target.
- **ED+CM**: Another simplified version that is composed of encoder, decoder, and CM modules. No classifiers and corresponding training objectives were used for training. Following the work of [101], we improved this system by conditioning the encoder with the input's channel factor[5].
- **FULL (ED+CM+Classifiers)**: Our complete proposed system shown in Figure 4.2, which consists of an encoder, decoder, CM network, and three classifiers. Auxiliary (in Equation (4.6)) and adversarial (in Equations (4.5) and (4.7)) training objectives were integrated through these three classifiers.
- **Linear-ISTFT**: This system shares the same settings as **FULL**, except the decoder output was changed to linear spectrogram. Instead of WaveRNN, we synthesized the waveform using ISTFT with the noisy phase.
- **Wavenet**: A waveform-to-waveform mapping system based on Wavenet [36]. We reimplemented it with the same model architecture and training objective (i.e., L1 loss on log spectrogram).
- **WPE**: A state-of-the-art speech de-reverberation baseline, which estimates a linear filter to minimize the weighted linear prediction error [92].
- **WPE+L**: An integrated system that sequentially combines **WPE** for de-reverberation and a standard log-MMSE magnitude estimator [17] for denoising.

4.3.4 Objective evaluations

We first evaluated each system with objective measures. We used the short-time objective intelligibility (STOI) score [72] to measure speech intelligibility and three composite scores (CSIG, CBAK, and COVL) [68] to measure enhancement

[5] As an alternative to adversarial training, this additional conditioning was used to encourage the encoder to produce channel-invariant features.

Table 4.2: Objective evaluation results of different systems on test set. For all four measures, higher scores indicate better performance.

System	CSIG	CBAK	COVL	STOI
Raw audio	3.05	2.23	2.60	0.869
WPE	3.16	2.41	2.75	0.888
WPE+L	2.81	2.33	2.52	0.811
Wavenet	3.67	2.42	3.08	0.904
Linear-ISTFT	3.94	**2.61**	**3.37**	0.905
ED	3.89	2.48	3.28	0.906
ED+CM	3.73	2.49	3.16	0.886
FULL	**3.94**	2.52	3.34	**0.906**

quality. As described in Section 2.3.1, CSIG, CBAK, and COVL are mean opinion score (MOS) predictions of speech distortion, noise distortion, and overall quality, respectively. The evaluation results are listed in Table 4.2.

As shown, the **FULL** system consistently improved its two simplified versions (**ED** and **ED+CM**) for all measures, which indicates both the CM network and classifiers played important roles in our proposed system. It also significantly outperformed time-domain **Wavenet** and the two signal processing-based baselines (**WPE** and **WPE+L**). **WPE+L** system performed much worse than **WPE**. This is mostly because the log-MMSE estimator suppressed noise too aggressively even though the noise level of the DAPS dataset was not high, therefore it degraded speech quality. We found that **FULL** system was worse than **Linear-ISTFT** in terms of CBAK and COVL. The probable reason is that the vocoder-generated speech had more artifacts than the ISTFT-generated one. However, most of these artifacts introduced by the neural vocoder did not affect human perception, as has been observed in the previous experiments (see Section 3.3.5). To comprehensively evaluate each system, we further conducted the following subjective listening tests.

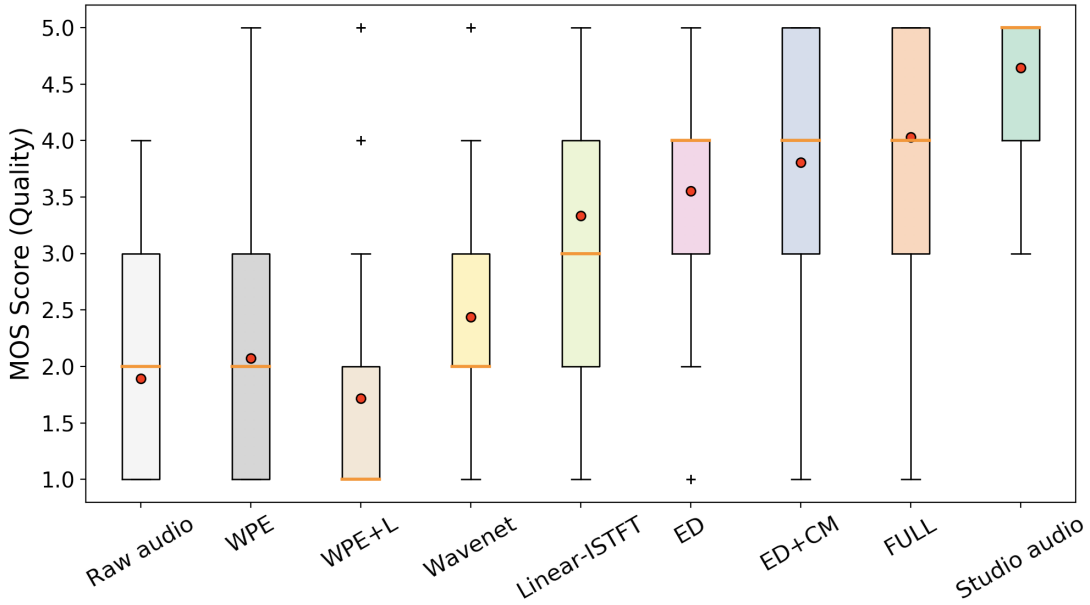


Figure 4.5: Box plot on MOS scores for audio quality. Red dots represent mean score.

4.3.5 Subjective evaluations

We conducted crowdsourced listening tests for the subjective evaluations. Specifically, we chose 120 (20 audios × 3 conditions × 2 genders) of the 228 tested audio clips for each system[6]. Participants (165 individuals) were asked to rate the quality of each anonymized audio from 1–5 (five-point Likert scale) for the mean opinion score (MOS). For reference, the raw (with low quality) and studio versions of each audio were also provided to the participants before rating. Each audio was rated ten times to avoid human bias.

The listening results are given in Figure 4.5. The Mann-Whitney U test reveals that the proposed **FULL** system significantly outperformed the other systems with p-values all lower than 1e-7. It is noteworthy that unlike the objective results, **FULL** system showed a higher score than **Linear-ISTFT**, which means the vocoder-based waveform synthesis (i.e., WaveRNN) module successfully improves the quality of the synthetic waveform. This also indicates that although the artifacts introduced by WaveRNN degraded the objective results, they did not

[6]Audio samples are available at: https://nii-yamagishilab.github.io/hyli666-demos/evr-slt2021

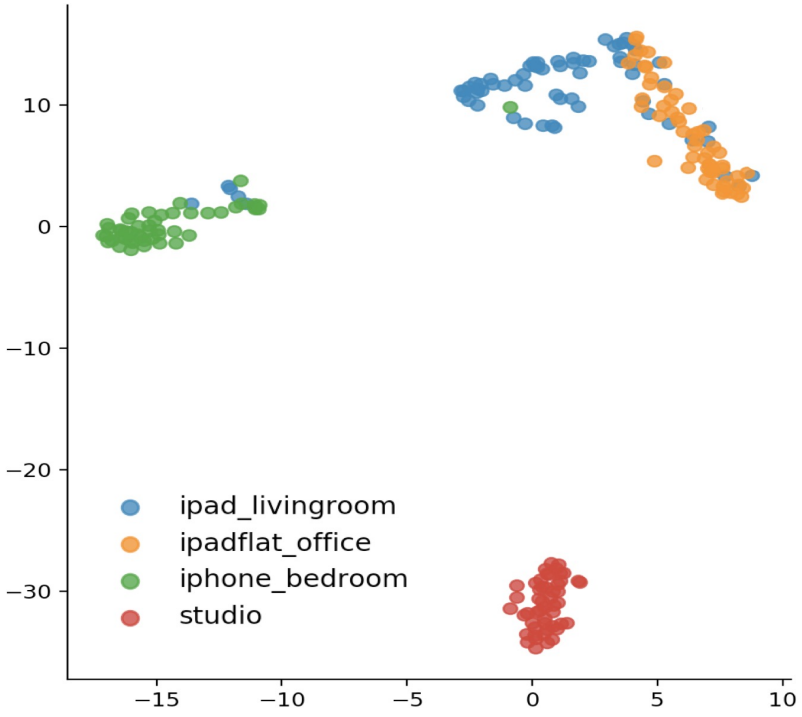


Figure 4.6: Visualization of learned channel factors using t-SNE transformation; color coded by channel condition labels.

affect human subjective evaluations. More interestingly, we can see that the **FULL** system outperforms **ED** in both subjective and objective tests, even though the task of **FULL** system was more challenging: the extra channel factor should be disentangled, and the output mel-spectrogram could be not only in studio quality but also in other acoustic characteristics based on the provided reference. Such additional learned knowledge related to channel information did benefit the **FULL** system and improved its performance.

4.3.6 Beyond noise reduction: audio effect transfer

In addition to noise reduction, the proposed system can also realize audio effect transfer: transferring the input recordings to sound as if they were recorded in another environment. To achieve this, we only need a few or even one reference audio recorded under the corresponding desired channel (or recording) condition.

Instead of using one-hot code, the CM network automatically encodes the

channel factor from the arbitrary reference, then the decoder can predict the target-environment mel-spectrogram conditioned on this factor. Figure 4.6 gives a visualization of learned channel factors for different reference recordings under three tested unseen conditions and one studio condition. We used t-SNE transformation [102] to project the 256-dimensional channel factor into 2 dimensions. We can see that the learned factors are clearly clustered based on their channel conditions[7], which indicates that the CM network can effectively discriminate unseen reference audios and produce representative factors. Therefore, it enables the system to deal with the unlabelled references under unseen channel conditions. With this system, we can further control the transferred effect (e.g. reverberation level) by flexibly scaling the channel factor. Examples of transferred mel-spectrograms are given in Figure 4.7, where we aimed to transfer a studio recording to sound as if it were recorded in the (unseen) *iphone_bedroom* condition. Instead of feeding a reference audio, the applied channel factor $\hat{\boldsymbol{z}}_c$ was pre-computed through linear interpolation of two factors using Equation (4.9):

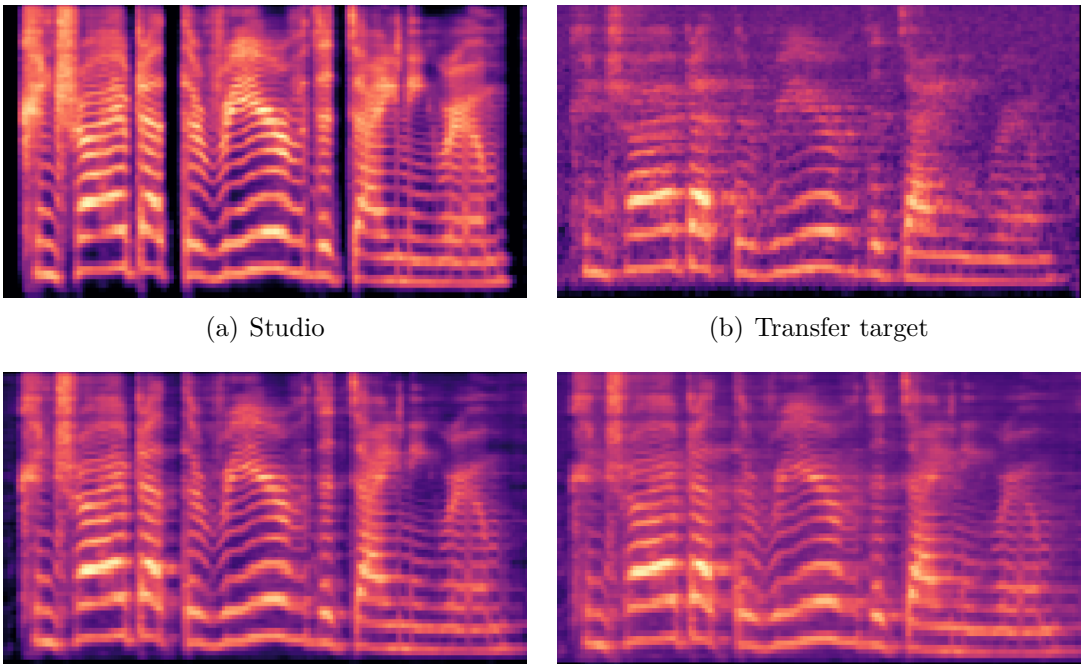
$$\hat{\boldsymbol{z}}_c = (1-\alpha) * \boldsymbol{z}_c^{pro} + \alpha * \boldsymbol{z}_c^{iph}, \tag{4.9}$$

where $\boldsymbol{z}_c^{pro}$ and $\boldsymbol{z}_c^{iph}$ denote the channel factors extracted from a professional studio recording and *iphone_bedroom* recording, respectively, and α is the scale value that ranges from 0 to 1. We successfully controlled the transferred effect from less reverberant (Figure 4.7 (c)) to more reverberant (Figure 4.7 (d)) by increasing the scale value α. We can also see that the transferred mel-spectrogram in Figure 4.7 (d) shares a similar audio effect (or channel characteristics) with the ground-truth transfer target in Figure 4.7 (b).

4.4 Summary

This chapter focuses on **issue 3: Improving noise reduction for device-degraded speech**. Compared to additive noise, device degradation, which

[7]There is a little overlap between the conditions of *ipad_livingroom* and *ipadflat_office*. This is because recording device used under both conditions was same (iPad Air), resulting in relatively similar channel characteristics.



(a) Studio (b) Transfer target

(c) Transferred at $\alpha = 0.6$ (d) Transferred at $\alpha = 1.0$

Figure 4.7: Examples of transferred mel-spectrograms: (a) Studio recording, (b) Target recording recorded in *iphone_bedroom* condition, (c) Transferred recording with $\alpha = 0.6$, and (d) Transferred recording with $\alpha = 1.0$.

includes not only additive noise but also reverberation and poor microphone response, is more likely to happen in the real-world speech application scenarios. The proposed system can transform low-quality speech recordings into high-quality ones. Specifically, by extending the noise token to channel token, we manage to disentangle the channel factor from a high-quality reference recording, which is then used to guide the system to predict the target high-quality mel-spectrogram. We also use WaveRNN vocoder to synthesize the final waveform.

Experimental results show that our system works well and outperforms several state-of-the-art baselines. Moreover, we show that it can be flexibly extended to transform the input recording into not only professional studio quality (as our primary target) but also with other acoustic (or channel) characteristics based on the reference we designate. Our future work will include improving the naturalness of the predicted mel-spectrogram. The possible approach is to incorporate a

generative adversarial network-based spectrogram discriminator [35, 36]. Also, we find DAPS dataset used in this chapter is relatively smaller with limited recording conditions. This inspires us to collect a new large-scale dataset consisting of various realistic device recordings to facilitate the future research. The details of this new dataset can be found in Appendix A.

5

Neural Intelligibility Boosting using Generative Adversarial Networks

Chapters 3 and 4 focus on the improved noise reduction models. This chapter looks into the another speech enhancement task – intelligibility boosting.

Instead of suppressing noise from noisy speech, intelligibility boosting is supposed to modify the speech signal only in such a way as to increase its intelligibility under the noisy environments. As explained in Section 2.2.4, deep learning techniques have not yet been widely applied in this task due to the lack of ground truth label, i.e., the "perfectly" intelligible speech. In this chapter, we explore and propose a novel neural approach for intelligibility boosting. To achieve this, we introduce generative adversarial networks (GANs) model [103] to simultaneously optimize multiple advanced speech metrics, including both intelligibility- and quality-related ones, which results in notable improvements in performance and robustness. Our system can not only work in non-real-time mode for offline audio playback but also support practical real-time speech applications.

Experimental results using both objective measurements and subjective listening tests indicate that the proposed system significantly outperforms state-of-the-art baseline systems under various noisy and reverberant listening conditions.

This chapter is structured as follows. Section 5.1 describes the application scenario and mathematically formulates the problem of intelligibility boosting. Section 5.2 gives details on our proposed neural system. Section 5.3 presents the experimental setup and results.

5.1 Scenario description and problem formulation

Real-life speech communication, such as mobile telephony and public-address announcement, usually occurs in noisy environments. These challenging environments severely degrade speech intelligibility, resulting in stressful listening or even non-understanding for listeners. Since noise sources are physically present in the near-end listener side, typical noise reduction methods which recover the clean speech from the noisy input, however, cannot be applied in such scenarios. As an alternative, intelligibility boosting aims to modify the speech signal only to improve its intelligibility when exposed to noise and reverberation.

Figure 5.1 depicts an application scenario of intelligibility boosting. Let $s(n)$ be the input speech signal with sampling index n. An algorithm is applied to modify $s(n)$, and then the intelligibility-enhanced signal $y(n)$ is output and played via a loudspeaker in a noisy environment. The signal $o(n)$ observed by the near-end listener can thus be represented as

$$o(n) = y(n) * h(n) + v(n), \tag{5.1}$$

where $*$ denotes the convolution operation, $h(n)$ is the room impulse response (RIR)[1], and $v(n)$ is the near-end noise disturbance. We further consider a common scenario in which the noise properties of $v(n)$ can be measured using a noise tracking algorithm via a reference microphone, such as the phone microphone for

[1]Loudspeaker response is integrated into the RIR for simplicity.

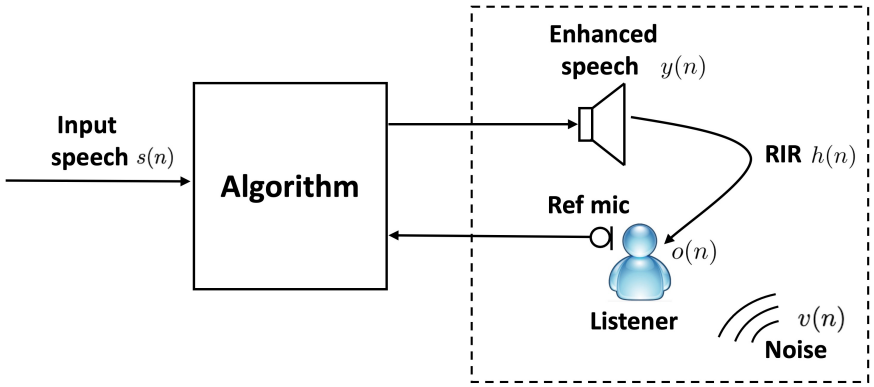


Figure 5.1: Real-life scenario of intelligibility boosting task.

mobile telephony. On the other hand, we disregard the effect of reverberation[2], i.e., $h(n)$, since in practice it is difficult to estimate reverberation parameters in the presence of additive noise. With these assumptions, our target is thus to develop a system that transforms $s(n)$ into $y(n)$ to improve the intelligibility of $o(n)$ under a known noise condition.

More specifically, speech modification is carried out to redistribute the speech energy over time and frequency. Let $S(m,k)$ be the short-time Fourier transform (STFT) spectrogram of the raw input signal $s(n)$, with the frame index m and frequency index k. We divide and group the frequency bins into the ERB-scaled bands [104] using triangular filter banks with the peak response being at the boundary between bands. Therefore, the input speech energy within one ERB band (indexed by band i at frame m) is given by

$$E_s(m,i) = \sum_k g_i(k)|S(m,k)|^2, \tag{5.2}$$

where $i \in \{1,2,\cdots,I\}$ with I the total number of ERB-scaled bands, and $g_i(k)$ is the amplitude of the i-th triangular band at the k-th frequency bin. Similarly, we denote the spectrogram and energy band of noise as $V(m,k)$ and $E_v(m,i)$, respectively. The modified speech energy within one ERB band can be represented as $\alpha^2(m,i)E_s(m,i)$, where $\alpha(m,i)$ are the amplification factors that redistribute

[2]We tried to model reverberation in the preliminary experiment but got unsatisfactory results. Details can be found in Appendix B

the speech energy across time and frequency bands. Also, we do not change the speech energy level, i.e., signal power before and after modification must be the same; thereby, we have the following equal-power constraint with respect to $\alpha(m,i)$:

$$\sum_{m,i} \alpha^2(m,i) E_s(m,i) = \sum_{m,i} E_s(m,i). \tag{5.3}$$

Next, the interpolated amplification factors applied to each frame m and frequency bin k are obtained by

$$\hat{\alpha}^2(m,k) = \sum_i g_i(k) \alpha^2(m,i). \tag{5.4}$$

They are then multiplied with the input spectrogram $S(m,k)$ to produce the enhanced spectrogram $\hat{\alpha}(m,k)S(m,k)$, which is subsequently converted to the enhanced signal $y(n)$ through the inverse STFT.

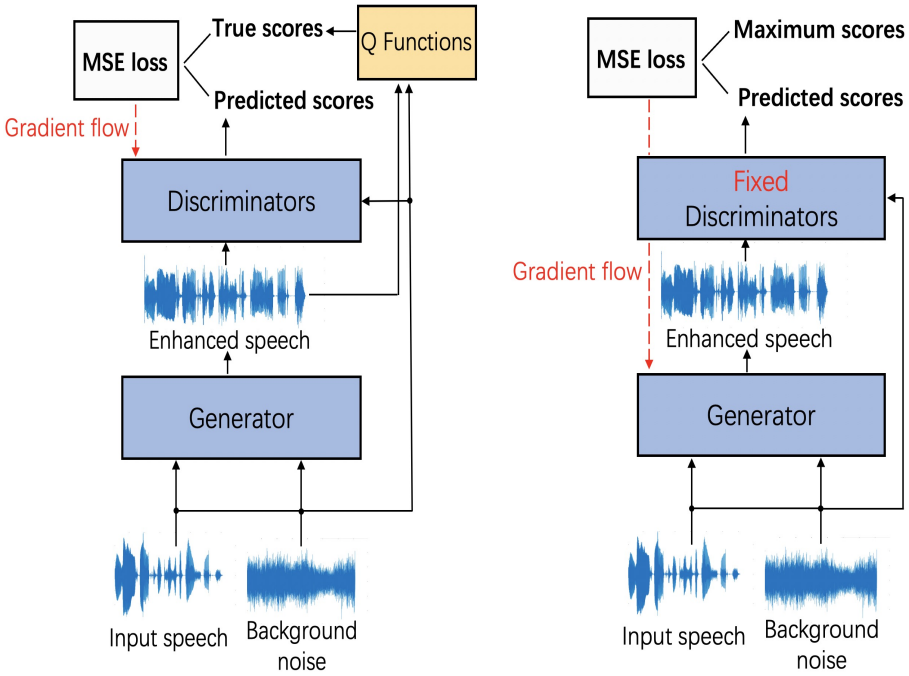
Instead of relying on expert knowledge to design an algorithm, we select several objective intelligibility and quality metrics as our optimization targets. We will further introduce the selected metrics in Section 5.2.1. On the basis of the above discussion, we now reformulate the problem as follows. Given the noise estimation (in the form of $E_s(m,i)$) and the constraint of Equation (5.3), *our target is to find the amplification factors $\alpha(m,i)$ per time frame and ERB band to optimize the objective metrics of interest.*

5.2 GAN-based intelligibility boosting

In this section, we introduce our proposed GAN-based system to jointly optimize multiple speech metrics for improved intelligibility.

5.2.1 Target speech metrics

Objective metrics are used to measure the intelligibility of speech distorted by noise and reverberation. Very recently, Van Kuyk *et al.* [64] extensively tested the accuracy of many of these metrics by comparing their correlation coefficients with listening test scores. We accordingly selected the top three reliable intelligibility metrics to build up and evaluate our proposed system. The target metrics are SIIB [67], HASPI [65], and ESTOI [73]. In addition to intelligibility metrics, we also



(a) Training process of discriminators (b) Training process of generator

Figure 5.2: Diagram of the GAN model of the proposed system for intelligibility boosting.

selected the following two state-of-the-art quality metrics: PESQ [69] and ViSQOL [70]. These two quality metrics are incorporated as the optimization targets to compensate for the quality loss caused by intelligibility-enhancing modifications. The details of above-mentioned metrics have been mentioned in Section 2.3.

Although these metrics can achieve high correlations with human subjective perception, they are too complex and mathematically intractable to handle. Particularly for a DNN model, we cannot directly use such metrics as the training criteria since most of them are non-differentiable[3]. To overcome this obstruction, we then introduce the GAN model into our system.

5.2.2 System overview

Figure 5.2 shows the diagram of the GAN model of our proposed system. It is composed of a generator (G), an intelligibility discriminator (D_{int}), and a quality discriminator (D_{qua}). The G receives the input speech s and the near-end noise v and then outputs the enhanced speech $y = G(s, v)$, where we omit sampling index n from this point forward. Next, D_{int} and D_{qua} predict the intelligibility and quality scores of the enhanced speech, respectively. The predicted scores of the discriminators are expected to be close to the true scores calculated from the target objective metrics. Compared with the original complex metrics, the gradients of DNN-based discriminators can be easily computed and back-propagated to G. Therefore, with the guidance of D_{int} and D_{qua}, G can be effectively trained to optimize the learned metrics of interest.

More specifically, we now explain the training process of D_{int} in detail. As shown in Figure 5.2(a), to predict the intelligibility scores, D_{int} takes three inputs: (1) the enhanced speech $G(s, v)$; (2) undistorted input speech s; and (3) background noise v. We introduce the so-called Q functions to represent the target metrics to be modelled, with $Q_{\text{int}}(.)$ the functions for intelligibility metrics and $Q_{\text{qua}}(.)$ for quality metrics. Moreover, the signal example $\hat{y}$, which is pre-enhanced using other reference algorithms (e.g., SSDRC [40] or OptSII [44]), is also fed into D_{int} in the training. As demonstrated in our earlier study [107], learning such additional examples can stabilize the training process and improve performance. Given all the above notations, the loss function of D_{int} is represented as follows:

$$\begin{aligned}\mathcal{L}_D^{int} = \mathbb{E}_{s,v}\{&[D_{\text{int}}(G(s,v),s,v) - Q_{\text{int}}(G(s,v),s,v)]^2 \\ &+ [D_{\text{int}}(\hat{y},s,v) - Q_{\text{int}}(\hat{y},s,v)]^2\}.\end{aligned} \tag{5.5}$$

By minimizing $\mathcal{L}_D^{int}$, D_{int} is encouraged to accurately predict the intelligibility scores. Similarly, we can represent the loss function of D_{qua} as Equation (5.6).

$$\begin{aligned}\mathcal{L}_D^{qua} = \mathbb{E}_{s,v}\{&[D_{\text{qua}}(G(s,v),s) - Q_{\text{qua}}(G(s,v),s)]^2 \\ &+ [D_{\text{qua}}(\hat{y},s) - Q_{\text{qua}}(\hat{y},s)]^2\}\end{aligned} \tag{5.6}$$

[3]ESTOI and PESQ metrics are technically differentiable under certain approximations, which have been studied in [105] and [106], respectively.

Note that different from D_{int}, D_{qua} takes only two inputs: the enhanced speech $G(s, v)$ and reference input speech s. This is because we have D_{qua} focus on measuring the quality of enhanced speech rather than the noisy observed speech.

Figure 5.2(b) illustrates the training process of G. We first fix the parameters of D_{int} and D_{qua}, and then apply the back-propagated gradients to update G to maximize the predicted intelligibility and quality scores. In order to increase the predicted scores as much as possible, we use the following loss function:

$$\mathcal{L}_G = \mathbb{E}_{s,v}\{[D_{\text{int}}(G(s, v), s, v) - t_{\text{int}}]^2 + \lambda[D_{\text{qua}}(G(s, v), s) - t_{\text{qua}}]^2\}, \tag{5.7}$$

where t_{int} and t_{qua} denote the maximum scores of the selected intelligibility and quality metrics, respectively, and λ is a hyper-parameter controlling the weight of speech quality to compensate for the quality degradation caused by intelligibility-enhancing modifications.

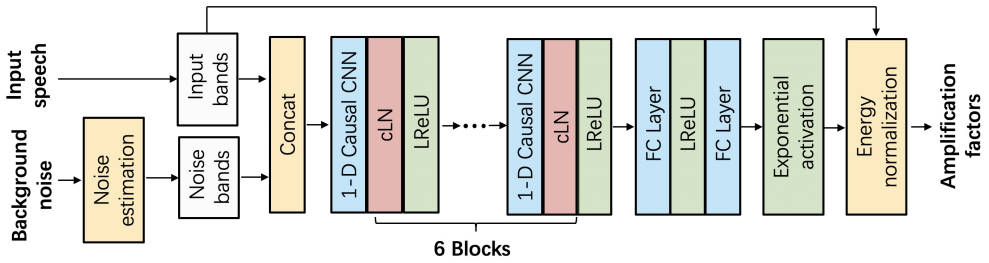
The generator (G) and discriminators (D_{int} and D_{qua}) are trained alternatively. At one training step, D_{int} and D_{qua} are trained individually with their corresponding loss functions, i.e., $\mathcal{L}_D^{int}$ and $\mathcal{L}_D^{qua}$. At the next training step, we fix the discriminators and only train G by minimizing loss $\mathcal{L}_G$. By this means, G can be effectively trained to optimize multiple advanced speech metrics, and the intelligibility of the enhanced speech (output by G) can be greatly improved and without too much quality degradation.

5.2.3 Network architectures

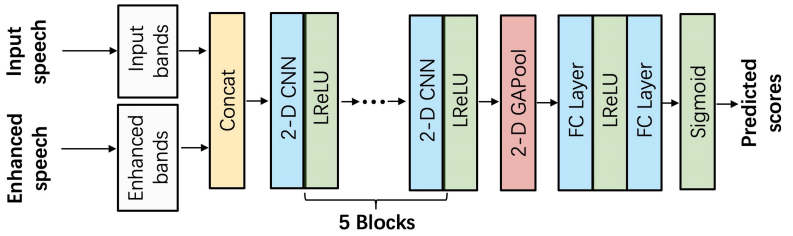
The details of the network architectures are given in Figure 5.3.

Generator

The input features for G are extracted from input speech and background noise. Specifically, the speech signal is transformed into the features containing 64 ERB-derived bands per time frame using Equation (5.2). There are two advantages for choosing features in the form of ERB-scaled bands rather than the raw frequency bins: (1) ERB-filterbank groups several perceptually-similar frequency bins into



(a) G architecture



(b) D_{qua} architecture

Figure 5.3: Network architectures of the proposed GAN model. Concat denotes concatenation operation. We set slope = 0.3 for all LReLU activations.

one band, producing a more robust feature[4]; and (2) The number of ERB bands is less than that of frequency bins, which can reduce the dimensions of the input and output features, resulting in a smaller model size. For background noise, we use the improved minima controlled recursive averaging algorithm (IMCRA) [12] to estimate noise power spectral density (PSD) $V^2(m, k)$, and then similarly extract 64 ERB bands as the noise features. These two features are then concatenated, resulting in a 128-channel feature vector and passed on to the following networks.

For network design, we choose the 1-D convolutional neural network (CNN) as the backbone for G due to the following reasons: (1) temporal convolution (1-D CNN with filter across time axis) has shown powerful modeling ability and been widely used in speech enhancement [38, 108, 109]; and (2) the 1-D CNN is suited for real-time applications due to its low computational complexity.

As shown in Figure 5.3(a), G consists of six blocks of causal 1-D CNN each with cumulative layer normalization (cLN) [38] and LeakyReLU activation (LReLU). The kernel size and output channels are set to (5, 256), (7, 256), (7, 256), (7,

[4]Filterbank-based grouping operations are also implemented as front-end processing in many intelligibility metrics such as SIIB[67] and ESTOI [73].

256), (7, 256), and (5, 64), respectively. Two 64-node fully connected (FC) layers are subsequently followed by the last CNN block. The element-wise exponential activation function is then applied as follows:

$$output = \exp\left(3 * \tanh\left(u\right)\right), \tag{5.8}$$

where u is the result of the last FC layer, and the scale range of Equation (5.8) is approximately 0.05 to 20. The 64-dimensional output vector serves as the raw (non-normalized) amplification factors $\alpha(m, i)$, which redistribute the speech energy across time and frequency bands: the speech energy $E_s(m, i)$ (at frame m within band i) is boosted when $\alpha(m, i) > 1$; otherwise, suppressed. Furthermore, we add an energy normalization layer where the raw amplification factors are multiplied by a global scale factor γ in order to satisfy the equal-power constraint of Equation (5.3). Finally, the normalized $\alpha(m, i)$ are applied to reconstruct the enhanced speech signal, as described in Section 5.1.

Except the last energy normalization operation, all layers in G are designed with causal configurations, which can run without dependencies of the future values of the signal. Moreover, G is a light-weight model containing only around 2.1M parameters. It performs intelligibility boosting very fast at the frame level, allowing for practical real-time speech applications. We will further discuss the extensions to real-time execution in Section 5.3.8.

Discriminators

Figure 5.3(b) gives the detailed architecture of D_{qua}. It takes two types of ERB bands as input features: the unmodified input speech bands and enhanced bands. The D_{qua} is composed of five layers of 2-D CNN with the following kernel size and number of channels: [(1, 1), 8], [(3, 3), 16], [(5, 5), 32], [(7, 7), 48], and [(9, 9), 64], each with LReLU activation. A 2-D global average pooling (GAPool) [110] is added to the last CNN block to produce a fixed 64-dimensional output vector, which is then followed by an FC layer with 64 LReLU nodes. The last FC layer with sigmoid activation predicts the scores of modelled quality metrics, i.e., PESQ and ViSQOL. Thus, the number of nodes are accordingly set to 2. Similar to our previous study [107], we apply spectral normalization with 1-Lipschitz continuity

[111] to all the layers used in D_{qua} to stabilize the training process.

For D_{int}, it shares the same network architecture with D_{qua}, except the inputs are changed to 3-channel features, i.e., (input, enhanced, noise), which requires an additional input of the estimated noise bands. Besides, the output nodes of D_{int} are set to 3, corresponding to the three intelligibility metrics to be modelled: SIIB, HASPI, and ESTOI.

5.3 Experiments

5.3.1 Data preparation

Speech materials consisted of Harvard sentences [76] spoken by two (one male [112] and one female [113]) native English speakers. The Harvard sentences are organized as 72 sets of 10 sentences each, and each set is designed to be phonetically balanced. Sentences were selected from sets 1–60, 61–66, and 67–72 for training, validation, and test data, respectively.

Six types of background noise were used: babble, restaurant, station, cafeteria, airport announcement, and speech-shaped noise (SSN), with the first five from the MS-SNSD dataset [86] and SSN artificially generated. For training and validation data, we selected four types of noise (babble, station, restaurant, and SSN) to generate noisy speech at three SNR levels, i.e., –11, –7, and –3 dB. The remaining two types of noise were used for test data. For cafeteria noise, the SNRs were set to –9, –5, and –1 dB; for airport announcements noise, they were set to –13, –9, and –5 dB.

Although reverberation was disregarded in the training, we extensively examined if the proposed system can work well in reverberant environments. Besides the original room condition (recorded in professional studios with reverberation time $T_{60} \approx 0.30$ s), another two RIRs were selected from a large room ($T_{60} = 0.61$ s) in the MIRD database [114] and stairway ($T_{60} = 0.92$ s) in the AIR database [115]. Thus, there were a total of three (1 original + 2 selected RIRs) reverberant environments considered in the test set. When generating noisy-reverberant speech, we first convolved the raw speech with the RIR, and then added the masker noise to the obtained reverberant speech at a desired SNR level.

To summarize, there were 14,400 (600 sentences × 2 genders × 3 SNRs × 4 noises) utterances in the training set; 1,440 (60 sentences × 2 genders × 3 SNRs × 4 noises) utterances in the validation set; and 2,160 (60 sentences × 2 genders × 3 SNRs × 2 noises × 3 reverberations) utterances in the test set. For the test set, a total of 18 listening conditions (comprising of 3 SNRs, 2 noises, and 3 reverberations) were extensively evaluated. It is worth noting that all the sentences, noises, reverberations (except the original condition), and SNR levels of the test set were unseen during model training.

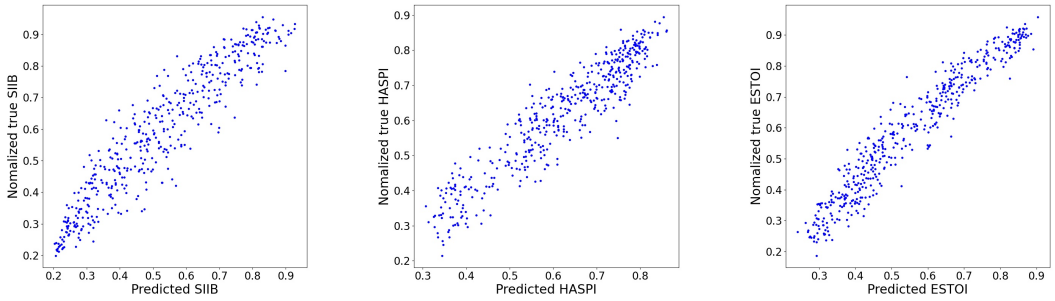
5.3.2 Implementation details

All signals were down-sampled to 16 kHz in our experiments. For feature extraction, we first used a Hanning window with a window size of 32 ms and hop size of 16 ms to compute the spectrogram. Next, 64 ERB-scaled triangular bands were applied to the spectrogram to produce the 64-dimensional input features for neural networks. All the input features were power-law compressed with a power of 1/6. We chose SSDRC [40] as the reference algorithm to generate the signal example $\hat{y}$ that was used in Equations (5.5) and (5.6). During training, we normalized all metric scores to the range of $[0, 1]$, i.e., the same range with sigmoid activation, and set the target maximum scores (t_{int} and t_{qua} in Equation (5.7)) to 1. Specifically, we used the following parametric logistic function for score normalization:

$$f(v) = \frac{1}{1 + \exp(a * (v - b))}, \tag{5.9}$$

where v denotes the raw metric score. Parameters (a, b) were accordingly set as $(-0.06, 32)$ for SIIB; $(-0.95, 2.8)$ for HASPI; $(-8.0, 0.25)$ for ESTOI; $(-1.5, 2.5)$ for PESQ; and $(-2.5, 2.2)$ for ViSQOL. These parameters were empirically chosen to make the normalized scores uniformly distributed between 0 and 1, which helps reduce bias and stabilize GAN training.

For GAN model configurations, the Adam optimizer was used in the training, with initial learning rates of 0.0004 and 0.0002 for the generator (G) and the discriminators (D_{int} and D_{qua}), respectively. The batch size was 1, and the hyper-parameter λ in Equation (5.7) was set to 0.5. The training process was



(a) SIIB, $r = 0.94$ (b) HASPI , $r = 0.93$ (c) ESTOI, $r = 0.97$

Figure 5.4: Correlations between the predicted intelligibility metrics and corresponding normalized ground-truth metrics. r denotes Pearson's correlation coefficient. Samples were selected from the test set with weak reverberation ($T_{60} \approx 0.30$ s).

terminated when all three intelligibility scores (SIIB, HASPI, and ESTOI) on the validation set stopped improving for five consecutive epochs[5].

5.3.3 Preliminary correlation test

First, we conducted a simple preliminary test to verify if the discriminator can well approximate the ground-truth metric. The proposed system can work only if this assumption holds.

Figure 5.4 plots the correlations between the predicted intelligibility metric scores, i.e., outputs of a trained D_{int}, and the corresponding normalized ground-truth metric scores. As can be seen, intelligibility metrics were highly correlated with the D_{int} predictions with all correlation coefficients $r > 0.9$, which demonstrates that the D_{int} managed to mimic the behavior of target modelled metrics.

5.3.4 Objective evaluations

In this section, we evaluated our neural intelligibility boosting system through objective measurements. We first re-implemented several baseline systems, and

[5]Source codes and the pre-trained model are available at https://github.com/nii-yamagishilab/NELE-GAN

then conducted an ablation test, yielding a total of eight systems evaluated in the experiments. We explain and notate each system as follows:

- **Unmodified:** Plain speech without any modification.
- **SSDRC:** A baseline system using the state-of-the-art SSDRC [40] algorithm, which achieved the highest and second highest intelligibility gains in the 1st [47] and 2nd [48] Hurricane challenges, respectively. It consists of two cascading non-parametric modifications: spectral shaping (SS) in frequency and dynamic range compression (DRC) in time.
- **iMetricGAN:** Our previously proposed system [107], in which we used BLSTM networks to optimize SIIB and ESTOI. Its model size was 7.8M parameters, which is much larger than the proposed system (2.1M parameters for G).
- **S-GAN:** A system optimizing only SIIB, in which D_{int} was simplified to predict only a single SIIB score, and no D_{qua} was used for optimizing quality metrics.
- **H-GAN:** A system optimizing only HASPI.
- **E-GAN:** A system optimizing only ESTOI.
- **Proposed (S+H+E):** A partial version of our proposed system jointly optimizing three intelligibility metrics, i.e., SIIB, HASPI, and ESTOI. No D_{qua} was used for optimizing quality metrics.
- **Proposed (All):** Our full proposed system jointly optimizing three intelligibility metrics (SIIB, HASPI, and ESTOI) and two quality metrics (PESQ and ViSQOL).

We used the same target metrics (SIIB, HASPI, and ESTOI) as the evaluation measurements due to their high correlations with human perception [64]. Moreover, we incorporated an additional advanced metric sEPSM [74]. Note that sEPSM was completely unseen to the model, it was thus regarded as a third-party evaluation measurement in the experiments. As discussed in Section 5.3.1, the objective

Table 5.1: Average objective scores of the compared systems across different reverberant conditions under **cafeteria** noise.

System	Intelligibility in $T_{60} \approx 0.30$ s				Intelligibility in $T_{60} = 0.61$ s				Intelligibility in $T_{60} = 0.92$ s				Quality	
	SIIB	HASPI	ESTOI	sEPSM	SIIB	HASPI	ESTOI	sEPSM	SIIB	HASPI	ESTOI	sEPSM	PESQ	ViSQOL
Unmodified	15.90	1.92	0.228	6.70	15.76	1.77	0.220	6.61	9.26	1.42	0.134	5.89	4.50	5.00
SSDRC	30.98	2.74	0.314	7.03	24.72	2.27	0.273	6.77	15.24	1.83	0.199	6.04	3.52	**2.71**
iMetricGAN	35.61	2.85	0.302	7.16	26.90	2.34	0.256	6.88	16.44	1.89	0.193	6.14	3.20	2.56
S-GAN	37.89	2.77	0.239	7.31	30.57	2.35	0.208	7.04	17.91	1.79	0.154	6.20	2.08	2.02
H-GAN	35.12	**3.12**	0.242	**7.55**	27.58	2.61	0.205	7.13	16.57	1.99	0.149	6.28	2.07	2.08
E-GAN	34.20	2.71	**0.331**	7.21	28.17	2.36	**0.285**	6.94	16.03	1.81	0.207	6.15	3.07	2.38
Proposed (S+H+E)	**41.33**	3.11	0.313	7.53	**32.99**	**2.62**	0.268	**7.17**	**18.90**	**2.00**	0.194	**6.28**	2.63	2.17
Proposed (All)	37.97	2.95	0.324	7.44	31.05	2.52	0.277	7.11	18.48	1.96	**0.209**	6.26	**3.54**	2.69

Table 5.2: Average objective scores of the compared systems across different reverberant conditions under **airport announcement** noise.

System	Intelligibility in $T_{60} \approx 0.30$ s				Intelligibility in $T_{60} = 0.61$ s				Intelligibility in $T_{60} = 0.92$ s				Quality	
	SIIB	HASPI	ESTOI	sEPSM	SIIB	HASPI	ESTOI	sEPSM	SIIB	HASPI	ESTOI	sEPSM	PESQ	ViSQOL
Unmodified	16.25	2.20	0.191	6.63	16.12	2.07	0.190	6.61	9.43	1.58	0.115	5.79	4.50	5.00
SSDRC	32.49	3.38	0.286	7.24	25.80	2.71	0.261	6.85	16.37	2.17	0.203	6.06	3.52	**2.71**
iMetricGAN	35.68	3.44	0.280	7.37	27.72	2.73	0.250	6.95	17.98	2.23	0.204	6.18	3.22	2.58
S-GAN	42.34	3.54	0.214	7.82	34.21	2.85	0.195	7.26	21.75	2.25	0.160	6.30	2.12	2.04
H-GAN	39.19	**3.80**	0.226	7.89	31.50	3.03	0.201	7.34	20.25	**2.41**	0.165	6.37	2.08	2.10
E-GAN	35.04	3.36	0.283	7.39	28.88	2.82	**0.263**	7.03	18.09	2.23	0.205	6.17	3.07	2.40
Proposed (S+H+E)	**43.45**	3.75	0.279	**7.94**	**35.31**	**3.04**	0.250	**7.36**	**22.36**	2.40	0.206	**6.37**	2.71	2.19
Proposed (All)	42.54	3.72	**0.288**	7.87	34.30	3.00	0.257	7.30	22.03	2.38	**0.209**	6.36	**3.56**	2.67

intelligibility scores were extensively tested under two types of unseen noise under three room conditions: weak, medium, and severe reverberations[6]. The quality scores (PESQ and ViSQOL) were computed by comparing the enhanced speech (without noise and reverberation) with input unmodified speech. For the above-mentioned six measurements, higher scores indicate better performance.

Tables 5.1 and 5.2 list the average objective scores of each system under cafeteria and airport announcement noise, respectively. In both tables, **Proposed (All)** clearly outperformed the state-of-the-art baseline **SSDRC** in all room

[6]When computing the intelligibility scores under these reverberant conditions, the clean and distorted signals were time-aligned in advance.

conditions with much higher intelligibility scores and comparable quality scores. It also consistently improved upon **iMetricGAN** for all six measurements with a far smaller model size. Compared with **Proposed (S+H+E)**, **Proposed (All)** achieved much higher scores for speech quality with only a slight decrease in objective intelligibility scores[7]. **S-GAN**, **H-GAN**, and **E-GAN** performed well on their corresponding optimization targets. For example, we can see that **H-GAN** achieved the best HASPI scores in some cases. However, there still remains quite a bit of room for improvement in terms of other non-target metrics. This indicates that optimizing only a single metric might cause sub-optimality in those unconsidered metrics. By jointly optimizing multiple metrics, both **Proposed (S+H+E)** and **Proposed (All)** showed much more robust performance on all intelligibility measurements. Specifically, **Proposed (S+H+E)** produced the best results in terms of unseen sEPSM scores, and this further demonstrates that the multi-metric optimization strategy can lead to effective and generalized intelligibility improvement. More interestingly, we found that **Proposed (S+H+E)** and **Proposed (All)** can achieve extra SIIB gains even compared with the pure SIIB-oriented **S-GAN** system.

5.3.5 Subjective evaluations

In this section, we evaluated the proposed system through subjective evaluations.

Intelligibility listening test

We conducted an intelligibility listening test to further evaluate the following five systems: **Unmodified**, **SSDRC**, **iMetricGAN**, **Proposed (S+H+E)**, and **Proposed (All)**.

60 Harvard sentences (sets 67, 69, and 71 of the female speaker; and sets 68, 70, and 72 of the male speaker) were extracted and presented in each of the 18 listening conditions (3 SNRs $\times$ 2 noises $\times$ 3 reverberations), producing a total of 5,400 tested utterances (60 sentences $\times$ 18 conditions $\times$ 5 systems). We then divided these tested utterances into 90 blocks: each block consisted of 60

[7]We also found that the quality scores can be further improved using a larger weight λ in Equation (5.7) at the cost of lower intelligibility scores.

individual Harvard sentences, and each Harvard sentence was processed by a random system and under a random listening condition. A total of 90 native English speakers with no reported hearing impairments were recruited for the online test, and all were paid. Each participant was assigned to one block. They were instructed to listen to each tested utterance only once then type in as many words they heard as possible. We also implemented a cheater-detection mechanism by assigning five additional validation utterances (with very slight noise) to each block of the main listening test. Participants who did not reach 60% average word accuracy on these utterances were considered unqualified listeners, which led to three participants being excluded from the analysis. Following the evaluation rules of the 1st Hurricane challenge [47], we only accounted for the correct content words in each transcription by excluding the short common words: 'a', 'the', 'in', 'to', 'on', 'is', 'and', 'of', and 'for'. The keyword accuracy rate (KAR) was then computed as the performance measure of intelligibility.

The results are plotted in Figures 5.5 and 5.6. Fisher's least significant difference (LSD) was also separately computed for each listening condition using ANOVAs to enable statistical comparisons of different systems. As shown, modification algorithms can generally lead to substantial intelligibility gains to the unmodified speech, except for four extremely challenging conditions where all systems failed to reach 10% KAR. The best system in all but two of the 18 conditions was **Proposed (All)**. For all conditions, it consistently outperformed not only **iMetricGAN**, but also the state-of-the-art **SSDRC**. Interestingly, although its objective intelligibility scores were lower than those of **Proposed (S+H+E)** (see in Tables 5.1 and 5.2), it showed much higher increases in KAR. This reveals that incorporating quality metrics into training can largely contribute to subjective intelligibility, which is likely due to the effective suppression of audible artefacts[8].

[8]Audio samples of the tested systems are available at https://nii-yamagishilab.github.io/hyli666-demos/intelligibility/index.html

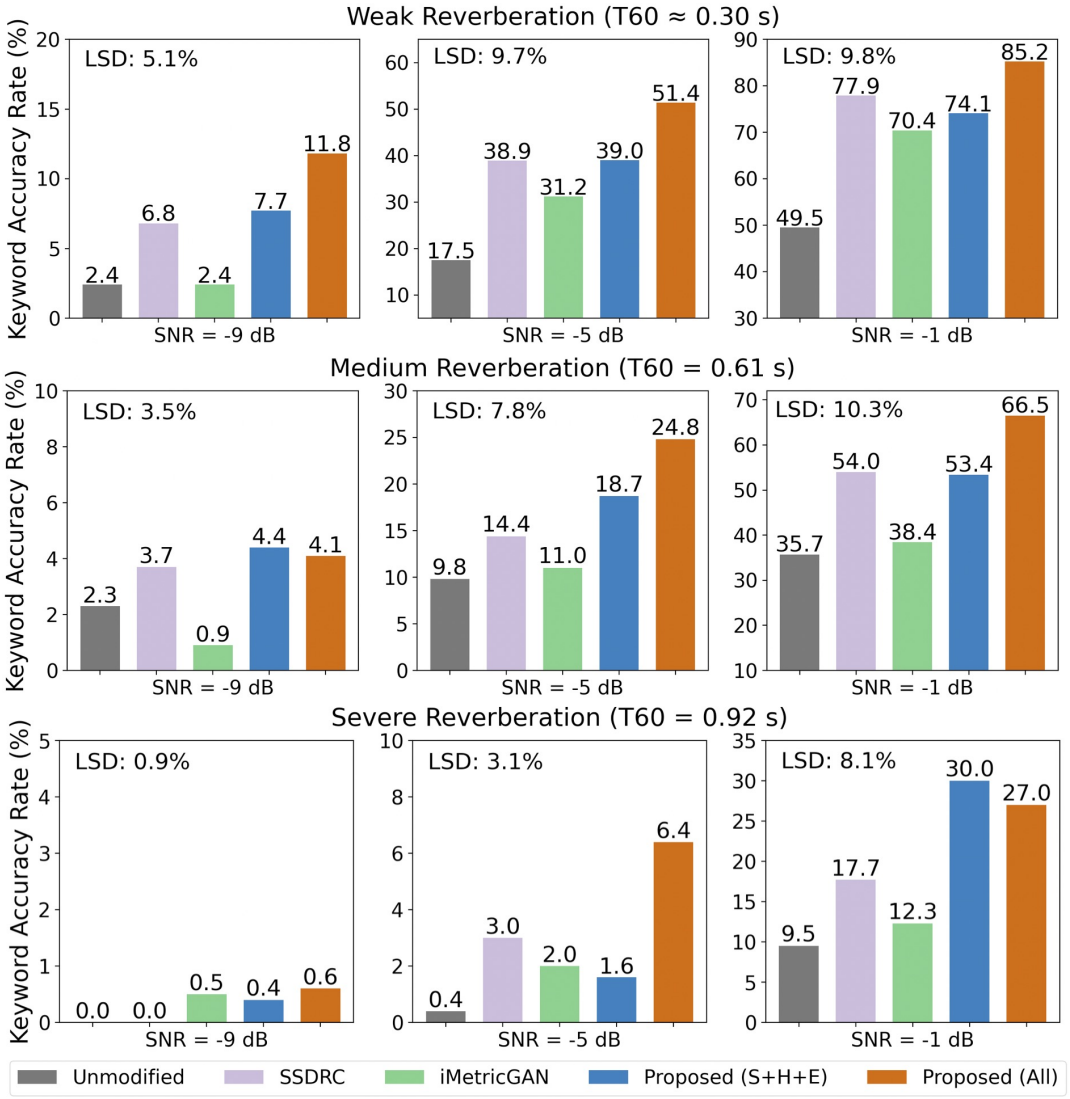


Figure 5.5: Mean keyword accuracy rates (KARs) in percentage points for each compared system across different listening conditions under **cafeteria** noise.

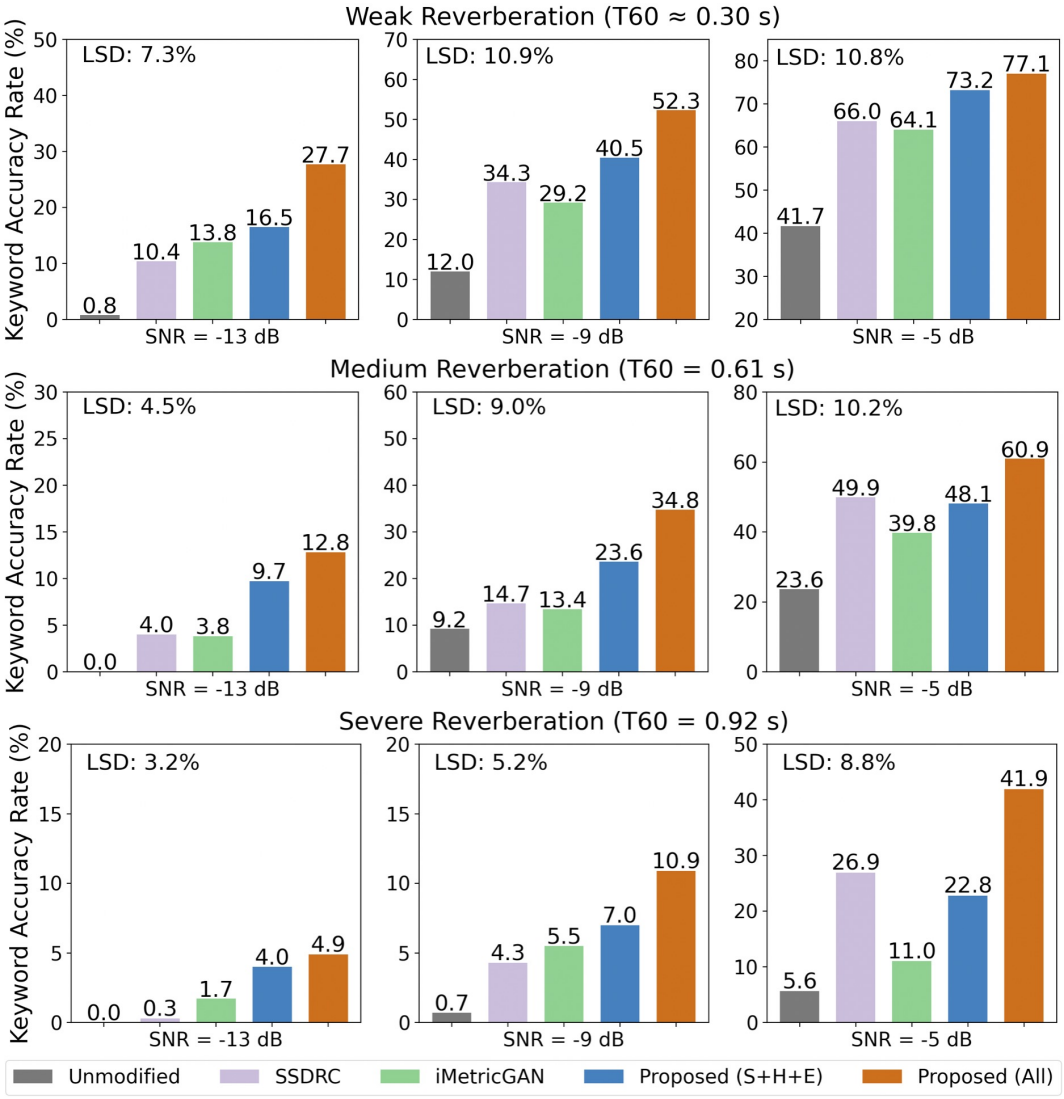


Figure 5.6: Mean keyword accuracy rates (KARs) in percentage points for each compared system across different listening conditions under **airport announcement** noise.

Quality preference test

We also conducted AB preference tests to evaluate the perceptual quality of the enhanced speech. We conducted pairwise comparisons between **Proposed (All)** and the following three systems: (1) **SSDRC**; (2) **iMetricGAN**; and (3) **Proposed (S+H+E)**. 90 enhanced samples were randomly selected from the

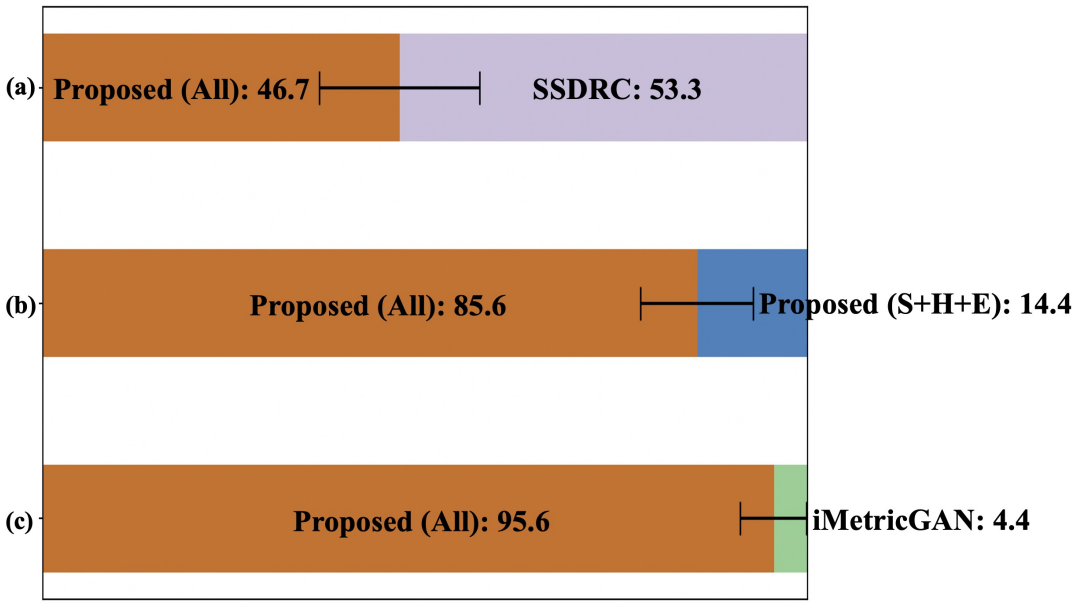
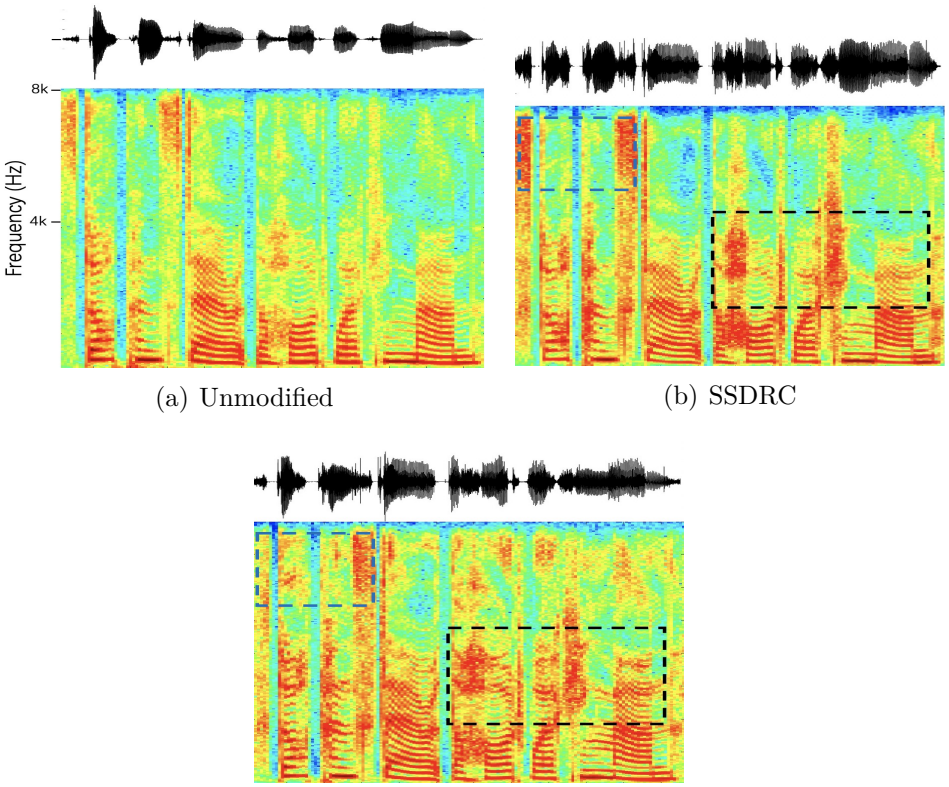


Figure 5.7: Preference scores (%) with 95% confidence intervals on speech quality compared between **Proposed (All)** and three reference systems.

test set for each system, and a total of 15 listeners participated. Each participant was instructed to listen to 18 randomized sample pairs, and for each pair they had to select the one that sounded better in terms of speech quality. As we can see from Figure 5.7, **Proposed (All)** achieved significantly higher preference scores than **iMetricGAN** and **Proposed (S+H+E)** and performed comparably with **SSDRC**. Such results clearly indicate that speech quality can be effectively improved through incorporating objective quality metrics into model training.

5.3.6 Acoustic analysis on enhanced speech

We analyzed the acoustic properties of the intelligibility-enhanced speech. For deeper insight, we used **SSDRC** as the reference system to conduct a comparative study. Figure 5.8 gives examples of waveforms and spectrograms for different signals. From the spectrograms, we found that both **SSDRC** and **Proposed (All)** modified the speech signal through redistributing its energy from low frequencies to the middle and high frequencies. By comparing Figure 5.8(c) with (b), **Proposed (All)** tended to allocate more energy on the middle-frequency



(a) Unmodified (b) SSDRC

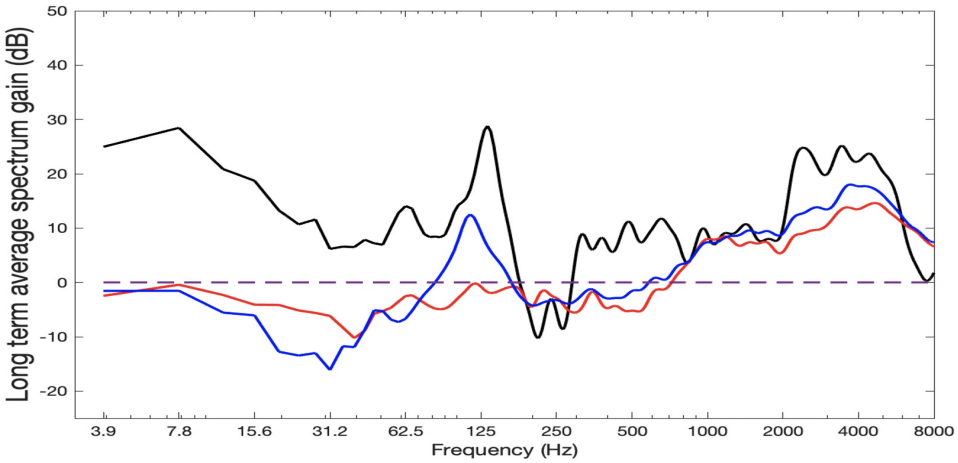
(c) Proposed (All)

Figure 5.8: Waveforms and their spectrograms on one utterance under cafeteria noise at SNR=–5 dB for different signals: (a) Unmodified input speech, (b) enhanced speech from **SSDRC**, and (c) enhanced speech from **Proposed (All)**. Utterance used is notated as "f_70_8", i.e., the 8-th utterance in 70-th list of female speaker.

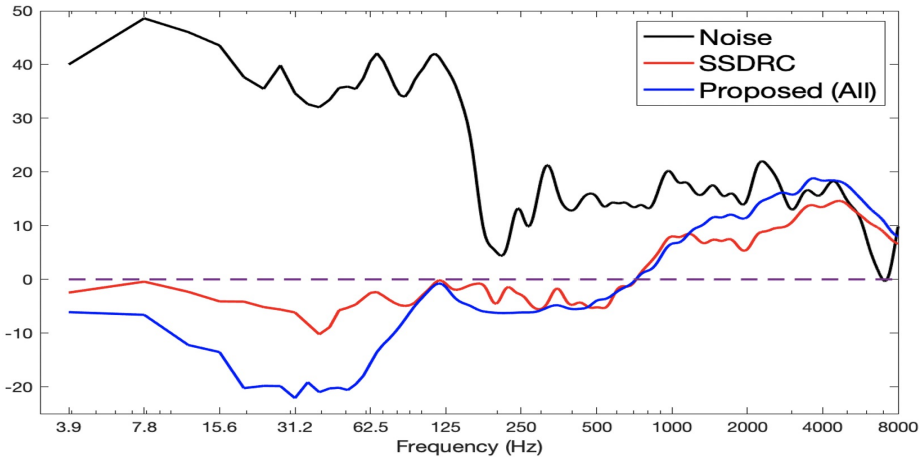
regions (2∼4 kHz) of the voiced segments (see black dashed box), while **SSDRC** emphasized the high-frequency regions (4∼8 kHz) of the unvoiced segments (see blue dashed box). We can also see that the waveform envelope of the enhanced speech from **Proposed (All)** is similar to that of the original unmodified speech. In contrast, the modified waveform of **SSDRC** drastically changed, resulting in more acoustic artefacts.

We also investigated the gain (in dB) of the long-term average spectrum (LTAS) calculated over one unmodified utterance. The gain values indicate the energy level of a signal in a certain frequency region: the signal energy is higher than the

unmodified utterance with gain > 0 dB; otherwise, lower. As shown in Figure 5.9, frequency regions from 1 kHz to 8 kHz were effectively boosted in both **SSDRC** and **Proposed (All)**, which accords with our observations in Figure 5.8. Different from noise-independent **SSDRC**, **Proposed (All)** can adapt well to the changing environments. For example, the noise in Figure 5.9(b) was extremely strong (up to 40 dB gain) in the low-frequency regions ($\sim$125 Hz). Thus, the system automatically gave up much more speech components in these regions, compared with how it performed under weaker noise in Figure 5.9(a). We also found that the speech components between 65 Hz to 150 Hz were particularly boosted under cafeteria noise, as shown in Figure 5.9(a). Interestingly, this coincides with the properties of the cafeteria noise where a peak gain was also exhibited near the same regions (see blue and black lines). We hypothesize that by increasing the speech components in such narrow but noise-dominant regions, the target speech can be differentiated from the surrounding noise in an easier manner through achieving a certain perception threshold. On the other hand, **SSDRC** performed merely the same processing of the speech with two different noises (see red lines); therefore, it cannot make full use of additional noise information. This is one of the points explaining why our proposed system performed better in both objective and subjective evaluations.



(a) LTAS gain under cafeteria noise at SNR=–5 dB



(b) LTAS gain under airport announcement noise at SNR=–13 dB

Figure 5.9: Long-term average spectrum (LTAS) gain (dB) over LTAS of unmodified utterance (f_70_8) for: (1) masker noise, (2) enhanced speech from **SSDRC**, and (3) enhanced speech from **Proposed (All)**.

5.3.7 Analysis of system robustness

We further analyzed the system's robustness in two particular situations, where (1) speaker and language are unseen to the model; and (2) background noise estimation is not accurate.

Table 5.3: Average objective scores on new German speaker test set.

System	Intelligibility				Quality	
	SIIB	HASPI	ESTOI	sEPSM	PESQ	ViSQOL
Unmodified	12.64	1.63	0.167	6.65	4.50	5.00
SSDRC	25.27	2.40	0.252	7.00	3.40	2.58
Proposed (All)	**28.94**	**2.66**	**0.254**	**7.44**	**3.46**	**2.81**

Speaker and language generalization

We tested the proposed system on a separately-created German speaker test set to examine if it can work under the mismatched speaker and language conditions. Specifically, we extracted 100 clean utterances from an unseen male German speaker [48] and set the same 18 listening conditions (i.e., 2 noise types, 3 SNRs and 3 room conditions) as used in the original test set (see Section 5.3.1), resulting in a total of 1,800 tested utterances. Table 5.3 lists the objective evaluation results on this new German speaker test set, where the scores were averaged over all listening conditions. We can see that even though **Proposed (All)** was built only upon English training data, it still achieved significant intelligibility gains and outperformed **SSDRC** by a large margin. This further demonstrates that the proposed system is robust, which can generalize well to mismatched speaker and language.

Tolerance to noise estimation error

Next, we measured the tolerance of the proposed system to inaccuracy of background noise estimation. As described in Section 5.1, in order to exploit noise information, the proposed system requires a reference microphone and runs IMCRA [12] algorithm to estimate noise PSD, i.e., $V^2(m,k)$. However, such noise estimation might be inaccurate, for example, when the noise is highly non-stationary or the reference microphone is distant from the listener's position. To simulate estimation error in this process, we randomly marked certain noise PSD bins as error bins with an error rate of $\epsilon\%$; thus, the corrupted noise PSD $V_e^2(m,k)$ is given as

follows:

$$V_e^2(m,k) = \begin{cases} exp(N_G), & if\ error \\ V^2(m,k), & else \end{cases} \tag{5.10}$$

where N_G is the random noise generated from Gaussian distribution with the same mean and variance as those of $logV^2(m,k)$, and error rate controls the corruption level: a higher $\epsilon\%$ indicates that each estimated bin is more likely filled with random noise, making noise estimation more inaccurate. Figure 5.10 shows the objective metric scores under different error rates. For the intelligibility metrics (i.e., SIIB, HASPI, ESTOI, and sEPSM), the corrupted noise PSD did not affect performance much when the error rate ϵ was less than 40% but decreased intelligibility scores incrementally when $\epsilon > 40\%$. However, even when noise estimation completely failed (i.e., $\epsilon = 100\%$), **Proposed (All)** could still surpass the performance of **SSDRC** in intelligibility metrics (except ESTOI), which demonstrates that the proposed system is very robust against noise estimation error. From another point of view, by simply substituting random values for noise PSD, **Proposed (All)** degenerates into a noise-independent system. This also indicates that our system is flexible and can adapt to scenarios in which the implementation of a reference microphone is not available. More interestingly, we found that the quality metrics (PESQ and ViSQOL) increased with increasing noise estimation error. We hypothesize that the system tends to modify the speech in a relatively aggressive manner to fully make use of noise information, e.g., giving up much more speech components in low-frequency regions when low-frequency noise is strong (see blue line in Figure 5.9(b)). For larger $\epsilon\%$, the system cannot exploit useful information as the given noise PSD becomes random; therefore, it tends to perform moderate modification, resulting in higher quality scores.

5.3.8 Extensions to real-time execution

Real-time execution is crucial for many speech applications such as mobile telephony. In this section, we discuss the causality of the proposed system in detail. As discussed in Section 5.2.3, the G used in **Proposed (All)** can inherently perform intelligibility boosting at the frame level in a causal manner. However, due to the equal-power constraint of Equation (5.3), we still need to collect the

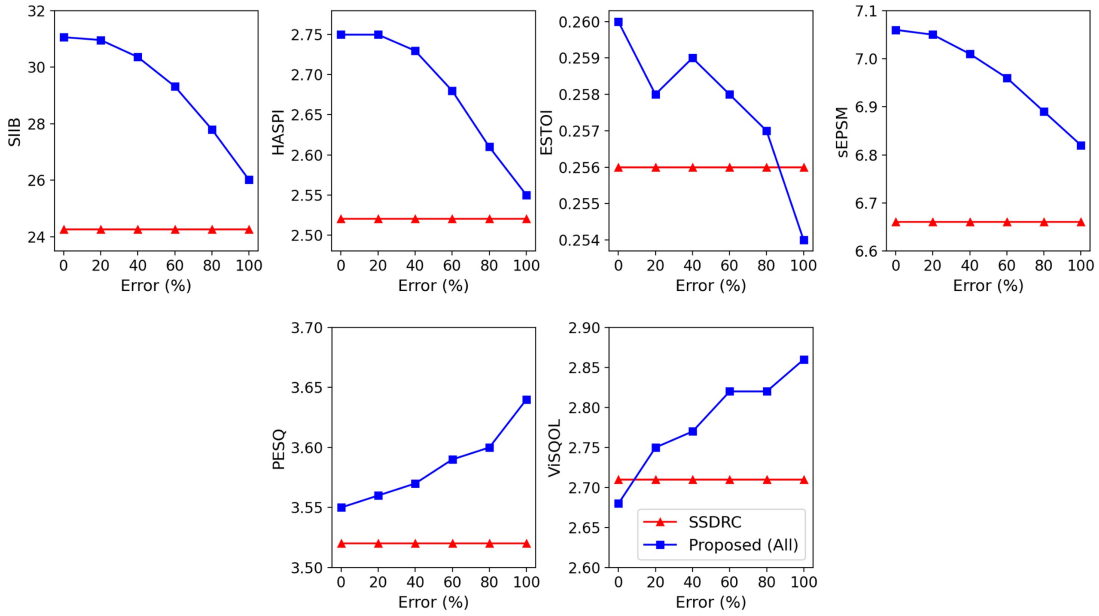


Figure 5.10: Average objective scores as noise estimation error is artificially added to noise PSD.

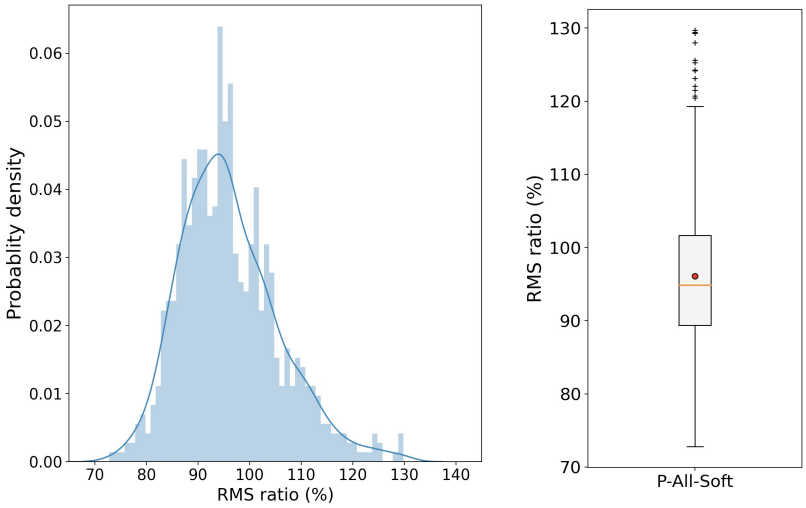
entire signal to calculate the global energy of an utterance. Thus, we consider two extended methods to overcome this limitation.

First, we revised the original utterance-level normalization (Equation (5.3)) to the following frame-level normalization:

$$\sum_i \alpha^2(m,i)E_s(m,i) = \sum_i E_s(m,i), \quad \forall m. \tag{5.11}$$

As shown in Equation (5.11), the energy is normalized at each frame m instead of the whole utterance, which enables the system to perform real-time execution under the equal-power constraint. We denote this modified frame-level normalization method for our proposed system as **P-All-FL**. Compared with the original utterance-level normalization method (denoted as **P-All-UL**), **P-All-FL** can only redistribute the speech energy across the frequency bands within one frame but not perform inter-frame redistribution.

Second, we consider another normalization method for application scenarios in which the equal-power constraint is not rigorous. As mentioned in Section 5.2.3, the global scale factor γ originally used in **P-All-UL** is calculated dynamically for



(a) Histogram. (b) Box plot.

Figure 5.11: Statistical results (γ used in **P-All-Soft** was set to 5.62) of RMS ratios between enhanced and unmodified raw speech: (a) frequency density histogram of RMS ratios, and (b) box plot on RMS ratios, with red dot representing mean score.

each utterance to achieve perfect energy normalization. With this new method, however, we prepare such a γ in advance by statically calculating the average energy ratio between the unmodified and enhanced speech over the whole training set. The γ can be then applied to the raw amplification factors to compensate for the energy loss, achieving a soft energy normalization where the enhanced speech has approximately the same energy with the unmodified one. We denote this method as **P-All-Soft**, and γ was determined as 5.62 by calculating over the training set.

Figure 5.11 presents the statistical results of the root-mean-square (RMS) ratios between the enhanced and unmodified speech on the test set. As shown in Figure 5.11, the distribution of RMS ratios was concentrated close to one with a very small deviation. This indicates that the energy of enhanced speech can be well maintained within the approximately same level as the unmodified one by using **P-All-Soft** method.

Table 5.4 lists the objective evaluation results on the three normalization methods. The scores were averaged over the whole test set across three SNR levels, three room conditions, and two unseen noises. We can see that **P-All-FL**

Table 5.4: Average objective scores for systems with different normalization methods on test set.

Normalization method	Causal	Equal-power constrained	Intelligibility				Quality	
			SIIB	HASPI	ESTOI	sEPSM	PESQ	ViSQOL
Unmodified	–	–	13.79	1.82	0.180	6.37	4.50	5.00
P-All-UL	×	✓	31.06	2.75	0.260	7.06	3.55	2.68
P-All-FL	✓	✓	20.15	2.26	0.193	6.73	3.29	2.53
P-All-Soft	✓	×	29.79	2.68	0.249	7.06	3.55	2.67

did provide intelligibility gains to the unmodified speech. However, it performed much worse than the other two methods due to the lack of inter-frame energy distribution, which further reveals that energy reallocation in time is crucial for intelligibility boosting. Although **P-All-Soft** cannot perfectly fulfill the equal-power constraint, it satisfies the causality requirement and showed a comparable performance to **P-All-UL**. Note that all three methods differed only in the energy normalization strategy, while the core model of G used in the experiments was identical. By choosing a suitable normalization method in accordance with actual needs, the proposed system can satisfy different requirements of causality and energy constraint.

Finally, we give a brief analysis on the system complexity. The intelligibility boosting module, i.e., G, is composed of 2.1M weight parameters. Since each weight is used once for one multiply-add operation per frame (16 ms), G thus takes 262.5 million floating-point operations per second (MFLOPS) for real-time execution[9]. For other main modules, including two FFTs (for input speech and background noise analyses, respectively), one inverse FFT (for enhanced speech reconstruction), and IMCRA noise estimation, they take around 4.0 MFLOPS. The total complexity of the proposed system is around 270 MFLOPS. Considering both model size and the computational complexity, our proposed system is light-weight and can be easily implemented in practice.

[9]One multiply-add operation is counted as two operations.

5.4 Summary

This chapter focuses on **issue 4: Incorporating deep learning into intelligibility boosting task**. To generate the intelligible and high-quality speech, we introduce a GAN model into our system to jointly optimize multiple intelligibility and quality metrics.

Three modules are used in the GAN model to carry out such multi-metric optimization: an intelligibility discriminator that learns to predict the objective intelligibility scores of speech as accurately as possible, quality discriminator that similarly learns to predict the objective quality scores, and a generator that enhances the input speech signal to maximize both intelligibility and quality scores, which are computed with the above two discriminators, respectively.

Experimental results from both objective measurements and large-scale listening tests indicated that the proposed system can lead to significant intelligibility gains and perform much better than compared baselines. It also generalizes well to various listening environments including unseen noises and reverberations. Moreover, the system is light-weight with only 2.1M parameters and can be easily extended to enable real-time execution.

For modified enhanced speech, there is a trade-off between the intelligibility gain and quality loss. In the future, we plan to investigate on this point and further propose a flexible system in which the intelligibility and/or quality can be adjusted by user demand.

6

Joint Framework for Full-End Speech Enhancement

In previous chapters, we have presented the improved methods for noise reduction (NR) and intelligibility boosting (IB), respectively. In real-world speech communication, however, noises often exist in both speaker and listener environments. For this complicated but common scenario, speech processing should be accordingly carried out as two sub-tasks: (1) NR: to suppress noise and recover clean speech in the far-end speaker side; and (2) IB: to pre-process speech signals (the output of the NR module) before playback to improve its intelligibility in the near-end listener side. In this paper, we refer to this two-stage task as *full-end speech enhancement* and propose to address it by a joint framework integrating NR with IB. Experimental results show that our proposed framework achieves promising results and significantly outperforms the disjoint processing methods in terms of various speech evaluation metrics.

This chapter is structured as follows. Section 6.1 formulates the problem

of full-end speech enhancement task. Section 6.2 introduces the proposed joint framework. Experimental setup and results are given in Section 6.3.

6.1 Introduction to full-end speech enhancement

In real-world application scenarios, as depicted in Figure 6.1, noises may exist in not only speaker but also listener environments, resulting in severe degradation of speech quality and intelligibility. To improve the listener's listening experience, we have to carry out NR and IB for simultaneously suppressing noise in far-end speaker side and boosting intelligibility in near-end listener side. The signal model follows[1]:

$$x = s + u, \quad \tilde{s} = NR(x), \quad y = IB(\tilde{s}|v), \quad o = y + v, \tag{6.1}$$

where s is the clean speech, u is the far-end environmental noise[2], v is the near-end environmental noise, and x is the signal received by the far-end microphone. The NR module receives x and outputs the estimated clean speech $\tilde{s}$, i.e., the denoised speech. By conditioning on the near-end noise estimation, the IB module further modifies $\tilde{s}$ before it is played by loudspeaker. The output enhanced speech is denoted as y. Finally, the signal o is observed by the near-end listener. Our goal is to improve the listening experience for listeners, i.e., the quality of y (without the near-end noise v) and intelligibility of o under v, by designing effective NR and IB modules. Also, to limit loudspeaker overload and unpleasant playback volume, we follow the equal-power constraint that requires that signal power before and after intelligibility boosting (i.e., $\tilde{s}$ and y) to be the same. Note that the input signal for IB module in Chapter 5 is assumed to be **perfectly** clean speech s, while in this chapter it is **estimated** clean speech $\tilde{s}$ which contains residual noise.

To address the full-end speech enhancement, researchers have explored joint processing of noise reduction and intelligibility boosting [116, 117, 118, 119]. Most existing works jointly control the NR filter along with near-end IB filter gain to optimize a certain target intelligibility metric, e.g., SII in [116] or mutual information in [117]. However, to make the optimization problem mathematically tractable,

[1] We disregard all related room transfer functions for simplicity.

[2] We only take into account the additive noise, but it can be generalized to other degradation such as reverberation and audio clipping.

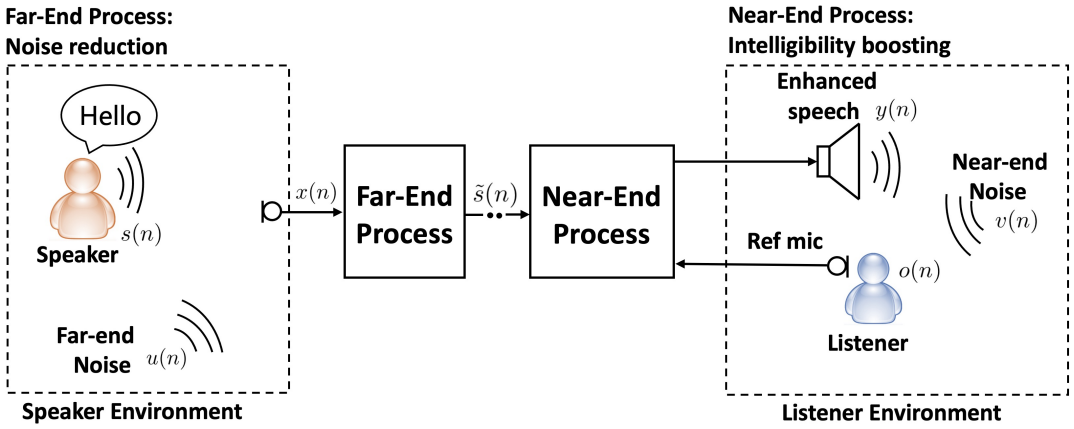


Figure 6.1: A scenario of real-world speech communication where noises exist in both speaker and listener environments. A reference microphone is used to measure the near-end noise properties.

the NR filter is considered to be relatively simple (e.g., Niermann *et al.* [116] used Wiener filter). Besides, most of them introduce additional assumptions and approximations to some extent, including target metric approximation [116, 118] and Gaussian signal model assumption [117], therefore limiting performance.

In this chapter, we propose a novel joint model for full-end speech enhancement. On the basis of previous explorations, we intuitively extend our proposed IB method (in Chapter 5) by integrating it with a mainstream neural NR method, leading to a fully DNN-based solution. This model can fully benefit from the powerful modeling capabilities of neural networks. Moreover, it can be jointly optimized using a unified loss function and without being dependent on unnecessary assumptions and approximations. Our experiments in Section 6.3 indicate that the proposed model significantly improves speech quality and intelligibility and clearly outperforms the disjoint pipeline methods.

6.2 Integrating noise reduction with intelligibility boosting

In this section, we introduce the proposed joint model. Figure 6.2 shows its overall diagram. It consists of three main modules: (1) a far-end NR module that

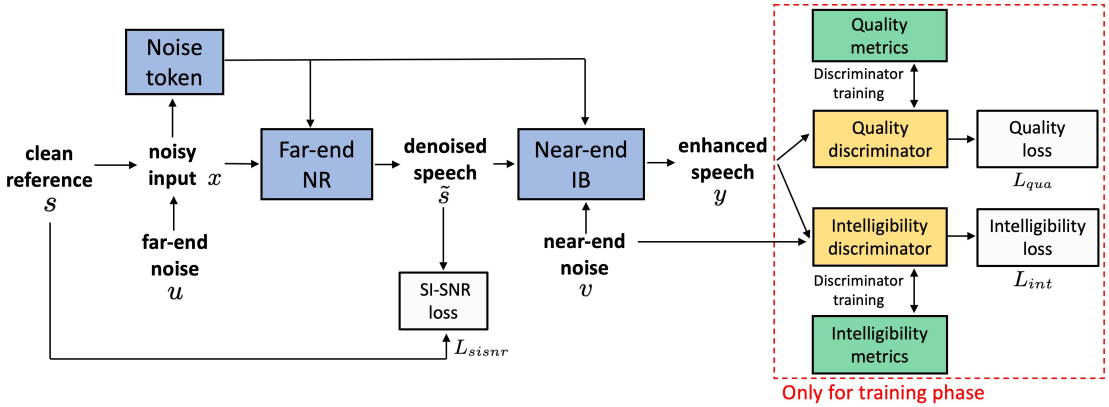
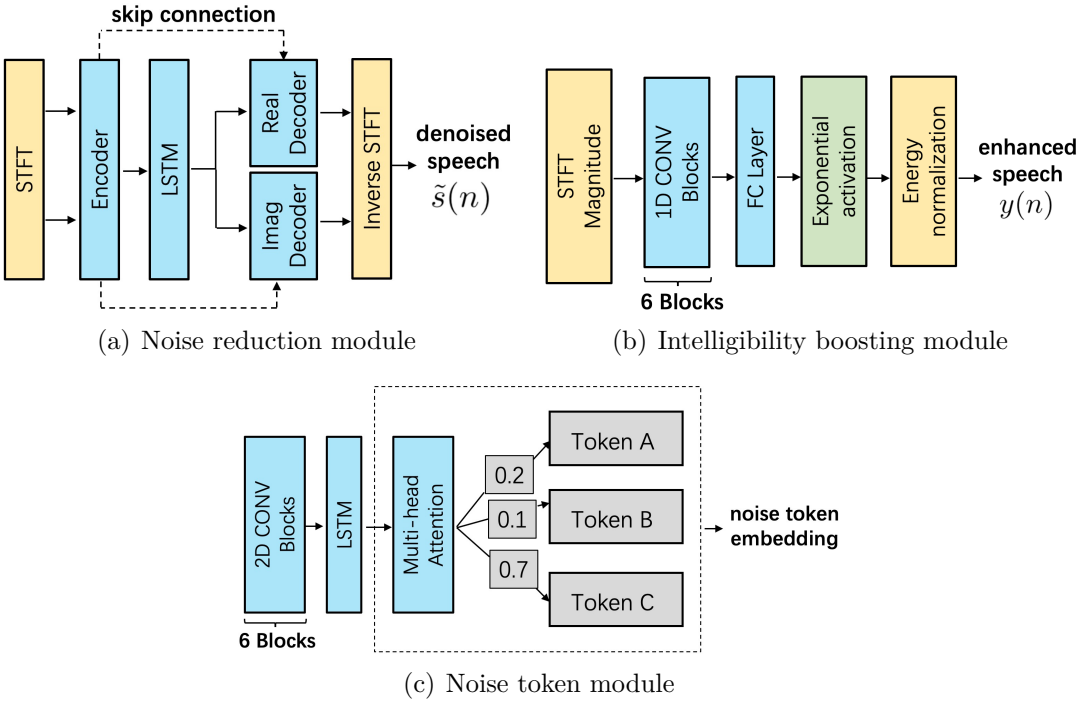


Figure 6.2: Overall diagram of the proposed method for full-end speech enhancement.

suppresses noise; (2) a near-end IB module that increases the intelligibility of denoised speech by redistributing its energy over time and frequency; and (3) the noise token module that extracts noise embedding and informs other modules of far-end environmental information. We use causal configurations for these three modules, which enables the model to perform real-time speech processing. Next, we will describe details of each module.

6.2.1 Far-end noise reduction

Far-end NR aims to suppress noise. The input is the noisy speech recorded by the far-end microphone, and the ideal output is a clean speech signal without noise disturbance. To achieve this, we use a convolutional recurrent network (CRN) [27] as the main neural architecture. As shown in Figure 6.3(a), the noisy speech is first converted into real and imaginary spectrograms by using STFT. We use a Hanning window with window size of 32 ms and hop size of 8 ms. The encoder consists of five 2D convolutional layers each with 1 (along the time axis)×3 (along the frequency axis) kernel, 1×2 stride, layer normalization [120], and parametric ReLU (PReLU). The output channels are set to 16, 32, 48, 64, 96, and 128, respectively. Between the encoder and the decoder, we insert a two-layer unidirectional LSTM with 512 nodes to model the temporal dependencies. The decoder comprises five transposed 2D convolutional layers with the same kernel and stride size as the



(a) Noise reduction module

(b) Intelligibility boosting module

(c) Noise token module

Figure 6.3: Illustration of the three major modules. CONV and FC denote convolutional and fully-connected layers, respectively.

encoder. Since skip connections are used to feed the output of each encoder layer as the additional input of the decoder layer, the output channels of the decoder are accordingly set to 256, 192, 128, 96, 64, and 32, respectively. The following two decoders respectively predict the real and imaginary parts of a complex ratio mask [39, 121], which are then multiplied with the original complex spectrogram to obtain the denoised one. The denoised speech is then generated with inverse STFT and passed on to the following IB module.

6.2.2 Near-end intelligibility boosting

The IB module modifies the denoised speech to make it sound more intelligible under the near-end environmental noise. We use the same IB system proposed in Chapter 5. The architecture of IB module is given in Figure 6.3(b). The detailed parameters are same as those described in Section 5.2.3, except the original input filterbank is replaced to the spectrogram magnitude. This is because we need to

make the data format of IB input consistent with that of NR output. Besides, we feed the far-end noise information into the IB module. As shown in Figure 6.2, the input features for the IB module includes: (1) spectrogram magnitude of denoised speech, (2) the near-end noise estimation (i.e., noise power spectral density estimated by reference microphone), and (3) neural noise embedding extracted from noisy input speech (we will explain this in Section 6.2.3). The optimization targets we selected are: SIIB [67], HASPI [65], ESTOI [73], PESQ [69], ViSQOL [70], and HASQI [65]. The former five metrics have been used in Chapter 5, and the last HASQI is a newly added quality metric.

6.2.3 Noise knowledge encoding

We also insert the noise token module into the joint model. As introduced in Chapter 3, noise tokens are a set of neural noise templates used to encode the far-end environment information and generate the corresponding noise embedding. Such embedding is regarded as additional noise knowledge and fed into both NR and IB modules. Figure 6.3(c) shows the detailed structure of the noise token module. It is exactly same with that used in Chapter 3.

We previously demonstrated in Section 3.3 that noise token embedding can improve the performance of the NR module. We expect that they can also benefit the IB module. For example, by exploiting far-end noise knowledge, the IB module may learn to avoid amplifying speech regions (in T-F bins) containing much residual noise.

6.2.4 Training objective

The training objective is composed of three terms:

$$L = L_{int} + \alpha * L_{qua} + \beta * L_{sisnr}, \tag{6.2}$$

where L_{int} is intelligibility loss calculated by the intelligibility discriminator (similar to that in Section 5.2.3), L_{qua} is quality loss calculated by the quality discriminator, and L_{sisnr} is speech denoising loss. α and β denote the weight parameters, respectively. To be more specific, L_{sisnr} is the scale-invariant signal-to-noise ratio

(SI-SNR) [38] calculated by comparing the denoised speech with the clean reference speech:

$$\begin{cases} s_{\text{target}} & := (<\tilde{s}, s>\cdot s)/||s||_2^2 \\ e_{\text{noise}} & := \tilde{s} - s_{\text{target}} \\ \text{SI-SNR} & := 10\log 10(\frac{||s_{\text{target}}||_2^2}{||e_{\text{noise}}||_2^2}) \end{cases} \tag{6.3}$$

where $<\cdot,\cdot>$ denotes the dot product between two vectors and $||\cdot||_2^2$ is Euclidean norm (L2 norm). Intelligibility loss L_{int} is defined as the mean square error between the predicted intelligibility scores and the maximum scores of target metrics:

$$L_{int} = ||D_{int}(y|v) - t_{int}||^2 \tag{6.4}$$

where y is the final enhanced speech output by the IB module, $D_{int}(y|v)$ is the predicted scores (under noise v) output by the intelligibility discriminator, and t_{int} is the maximum scores of the selected intelligibility metrics, respectively. By means of this loss, the IB module has to reach intelligibility scores as high as possible. Similarly we can define the quality loss L_{qua}. We jointly optimize the whole model (including noise token, NR, and IB modules) by using the loss function of Equation (6.2).

6.3 Experiments

6.3.1 Data preparation

We used two public corpora of Harvard sentences [76] (one spoken by male [112] and one by female [113]) in the experiments. We split the whole 720 Harvard sentences into 600, 60, and 60 for training, validation, and test data, respectively.

For training and validation, eight noise types were used in both far-end (speaker) and near-end (listener) environments. Far-end SNR levels were set to 4, 8, and 12 dB; near-end SNR levels were set to -11, -7, and -3 dB. By randomly combining these settings, we generated 28,800 and 2,880 utterances for training and validation, respectively.

For test set, the far-end noise type is cafeteria at three SNRs, i.e., 6, 10, and 14

dB; near-end noise type is airport announcement at three SNRs, i.e., -9, -5, and -1 dB. To summarize, the test set contained 1,080 utterances (60 sentences $\times$ 2 genders $\times$ 3 far-end SNRs $\times$ 3 near-end SNRs). Note that all the sentences, SNR levels, and noise types of the test set were unseen during model training.

6.3.2 Implementation details

All signals used in the experiments were resampled at 16 kHz. Improved minima controlled recursive averaging algorithm (IMCRA) [12] was used to estimate power spectral density of the near-end noise. During training, we applied parametric logistic function to normalize all metric scores into the range of $[0, 1]$, i.e., the same range with sigmoid activation, and set the corresponding target maximum scores (e.g., t_{int} in Equation (6.4)) to 1. We used Adam optimizer for training, with initial learning rates of 0.0002 for the three neural module components (noise token, NR, and IB) and 0.0001 for the discriminators (D_{int} and D_{qua}). In the training phase, we first trained NR and IB modules separately. We then combined and jointly trained them together with the noise token module. The batch size was 1, and the hyper-parameters α and β in Equation (6.2) were set to 0.6 and 0.005, respectively.

6.3.3 Objective evaluations

We evaluated the proposed joint model using six objective metrics. As mentioned in Section 6.2.2, the intelligibility metrics are SIIB, HASPI, and ESTOI; the quality metrics are PESQ, ViSQOL, and HASQI. For all these metrics, higher scores indicate better performance. The far-end noisy speech is processed by a certain system and then played under the near-end noise. We evaluated seven systems[3] and notate them as follows.

- **Noisy**: The far-end input noisy speech is played under the near-end noise without any modification.
- **Noisy+NR**: The far-end input noisy speech is processed only by the NR module.
- **Noisy+IB**: Processed only by the IB module.

[3]Audio samples: https://nii-yamagishilab.github.io/hyli666-demos/full-end-se

Table 6.1: Objective intelligibility scores averaged over three near-end SNRs for each far-end SNR condition.

System	Far-end SNR = 6 dB			Far-end SNR = 10 dB			Far-end SNR = 14 dB		
	SIIB	HASPI	ESTOI	SIIB	HASPI	ESTOI	SIIB	HASPI	ESTOI
Noisy	17.98	2.20	0.221	19.72	2.31	0.237	21.07	2.41	0.249
Noisy+NR	19.52	2.24	0.250	20.73	2.32	0.259	21.65	2.39	0.266
Noisy+IB	15.79	2.09	0.180	18.76	2.28	0.206	21.91	2.47	0.232
DSPPipeline	15.58	1.96	0.208	18.22	2.10	0.229	21.06	2.24	0.251
NeuralPipeline	24.47	2.67	0.302	27.34	2.85	0.319	30.09	**3.00**	0.333
Joint	26.16	2.70	0.305	28.65	2.84	0.319	30.77	2.96	0.330
Joint+NT	**28.48**	**2.73**	**0.320**	**31.45**	**2.87**	**0.334**	**33.79**	2.99	**0.344**

- **DSPPipeline**: Processed by the signal processing-based disjoint pipeline, which consists of Wiener filter (for NR) and SSDRC algorithm [40] (for IB).
- **NeuralPipeline**: Processed by neural network-based disjoint pipeline, which consists of the pretrained CRN-based NR [27] and GAN-based IB modules.
- **Joint**: Processed by the partial joint model (without the noise token module), in which the NR and IB models are jointly optimized.
- **Joint+NT**: Processed by the full proposed joint model (with the noise token module).

Intelligibility evaluation results are listed in Table 6.1, where the scores were averaged over the three near-end SNR levels. As we can see, applying only NR (**Noisy+NR**) or IB (**Noisy+IB**) does not increase the intelligibility. **Noisy+IB** has even lower scores than **Noisy**. This is mostly because the IB module wrongly amplifies the noise contained in the noisy input. To address the full-end speech enhancement problem, **NeuralPipeline**, **Joint**, and **Joint+NT** integrate both NR and IB modules, resulting in significant intelligibility gains compared with **Noisy**. In contrast, **DSPPipeline** has extremely low scores. This is because the SSDRC processor amplifies the residual noise that is produced by the former Wiener filter. Besides, we can clearly see that joint trained models improve upon the disjoint processing methods (**DSPPipeline** and **NeuralPipeline**). Moreover, benefiting from the noise token module that exploits the far-end environment

Table 6.2: Objective quality scores averaged over three near-end SNRs for each far-end SNR condition.

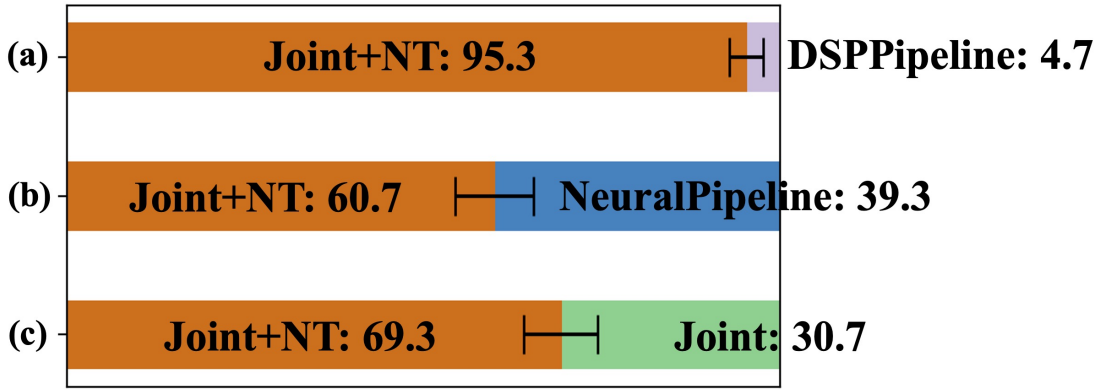
System	Far-end SNR = 6 dB			Far-end SNR = 10 dB			Far-end SNR = 14 dB		
	PESQ	HASQI	ViSQOL	PESQ	HASQI	ViSQOL	PESQ	HASQI	ViSQOL
Noisy	1.41	0.15	1.83	1.55	0.18	1.94	1.69	0.21	2.09
Noisy+NR	**2.33**	0.28	**2.48**	**2.52**	0.32	**2.69**	**2.70**	**0.36**	**2.91**
Noisy+IB	1.24	0.10	1.66	1.32	0.12	1.71	1.41	0.14	1.78
DSPPipeline	1.32	0.10	1.68	1.43	0.12	1.74	1.54	0.14	1.81
NeuralPipeline	2.01	0.23	2.14	2.19	0.26	2.25	2.35	0.28	2.35
Joint	2.14	0.28	2.20	2.30	0.30	2.32	2.43	0.33	2.43
Joint+NT	2.26	**0.30**	2.32	2.45	**0.32**	2.43	2.58	0.35	2.52

information, **Joint+NT** consistently outperforms **Joint** and achieves the overall best performance.

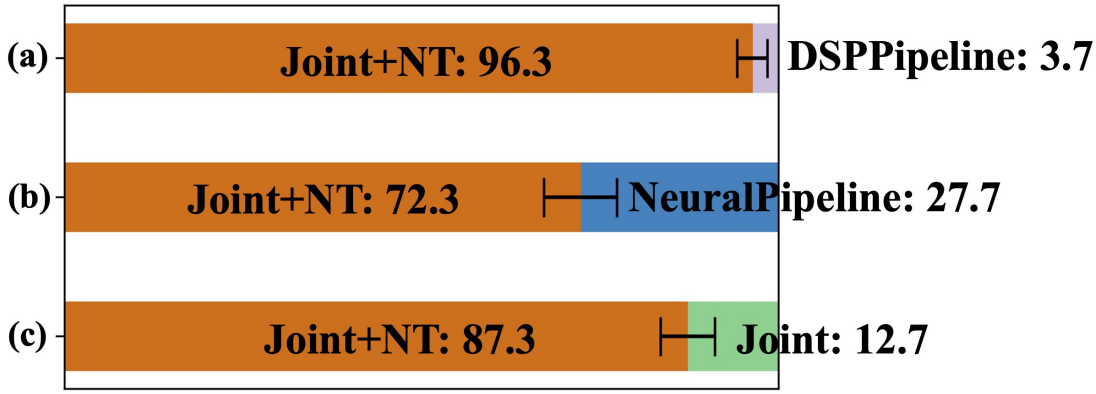
Table 6.2 lists the objective quality scores of enhanced speech y (without the near-end noise v). Since intelligibility-boosting modification inevitably degrades the speech quality at the cost of increasing intelligibility, **Noisy+NR** performs slightly better than the proposed joint models. However, we can see that joint models preserve speech quality much better than **DSPPipeline** and **NeuralPipeline**, which indicates the effectiveness of our proposed method.

6.3.4 Subjective listening tests

We conducted subjective preference tests to further evaluate the speech intelligibility and perceptual quality. Specifically, we conducted pairwise comparisons between **Joint+NT** and the three systems: (1) **DSPPipeline**, (2) **NeuralPipeline**, and (3) **Joint**. 300 enhanced samples were randomly selected from the test set for each system, resulting in 900 tested sample pairs (300 samples $\times$ 3 system pairs). A total of 20 native English speakers were recruited to participate in intelligibility and quality preference tests, respectively, and all were paid. For intelligibility test, each participant was instructed to listen to 45 randomized sample pairs played back under the near-end noise (i.e., the signal o in Equation (6.1)), and for each pair, they had to select the one that sounded clearer or that they could hear with



(a) Intelligibility preference test



(b) Quality preference test

Figure 6.4: Preference scores (%) with 95% confidence intervals.

less listening efforts. For quality test, each participant had to listen to 45 sample pairs (without the near-end noise, i.e., the signal y in Equation (6.1)) and select the one that sounded better in terms of listening quality.

As we can see from Figure 6.4, the proposed **Joint+NT** achieved significantly higher preference scores than all three compared systems in terms of both speech intelligibility and quality. We can see that **Joint+NT** outperformed **Joint** by a large margin in listening test results, which further indicates that exploiting far-end noise knowledge is useful for not only far-end noise reduction but also near-end intelligibility boosting.

Last, spectrogram examples (without near-end noise) of the evaluated systems are shown in Figure 6.5. We can see that **Noisy+IB** amplifies noise, making the

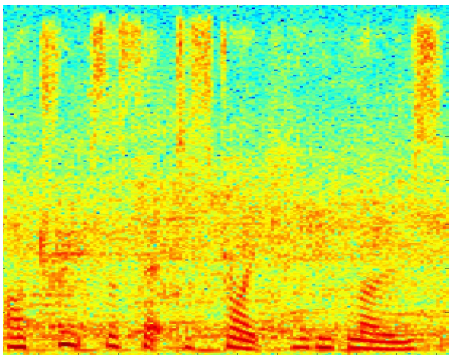
processed speech even noisy. The residual noise is wrongly amplified in **DSP-Pipeline**, resulting in severe distorted spectrogram. Compared with **Noisy+NR**, the middle-frequency regions of speech are emphasized in **NeuralPipeline**, **Joint**, and **Joint+NT**. Benefiting from joint optimization and far-end noise knowledge, the spectrogram of **Joint+NT** shows less residual noise and more clear fine structure (see black dashed boxes).

6.4 Summary

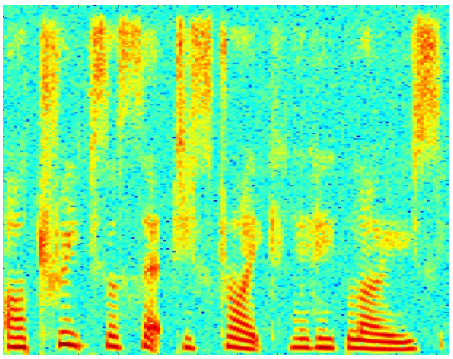
This chapter looks into **issue 5: Can we integrate noise reduction with intelligibility boosting for the scenario where noises exist in both speaker and listener environments?** The answer is yes, and we demonstrate such a joint framework is more efficient than disjoint pipeline methods.

Under the proposed framework, noise reduction and intelligibility boosting modules can be jointly optimized, where the NR module suppresses the noise of the input noisy speech, and the IB module further improves its intelligibility. Experimental results using both objective evaluations and subjective listening tests indicate that the joint framework can achieve significant intelligibility gain while preserving speech quality well. It also consistently outperforms the disjoint processing pipelines by a large margin.

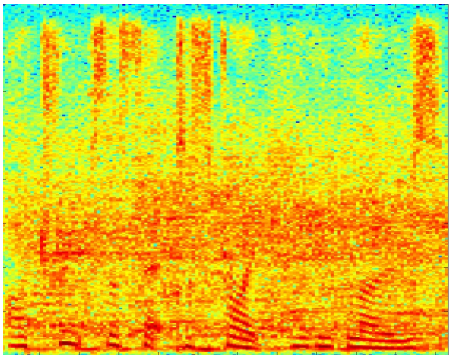
The far-end SNR in this study were set to a moderate level, i.e., 6, 10, and 14 dB. This is because we assume the close-talk microphone is placed close to the speaker. The full-end speech enhancement task becomes much challenging for the far-talk situation where speaker-side SNR is lower (e.g., –5 or 0 dB). We will further investigate on this corner case and try to improve our framework.



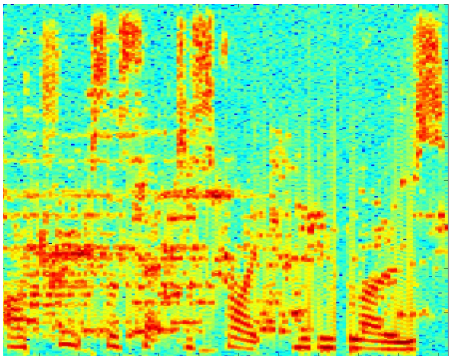
(a) Noisy



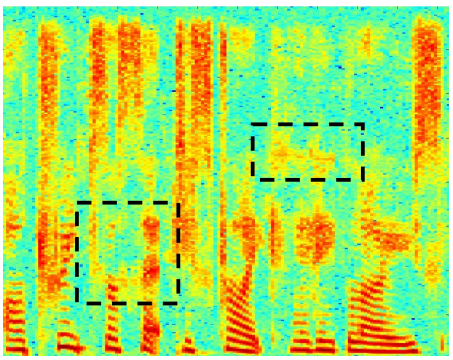
(b) Noisy+NR



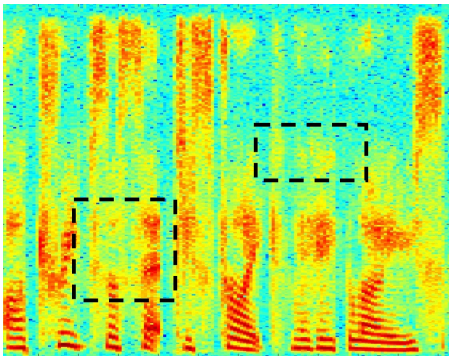
(c) Noisy+IB



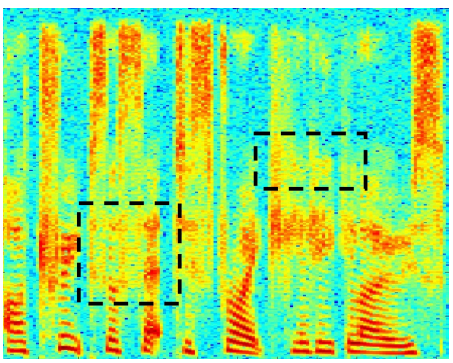
(d) DSPPipeline



(e) NeuralPipeline



(f) Joint



(g) Joint+NT

Figure 6.5: Examples of processed spectrograms (without near-end noise) for different systems: (a) Noisy, (b) Noisy+NR, (c) Noisy+IB, (d) DSPPipeline, (e) NeuralPipeline, (f) Joint, and (g) Joint+NT.

7 Conclusion

In the previous chapters, we presented improved neural network-based speech enhancement techniques for noise reduction and intelligibility boosting. As a result of exploring the five issues listed in Chapter 1, we summarize the main conclusions of this thesis as follows.

Issue 1: For neural noise reduction models, how to improve their generalization ability to unseen noise?

We proposed a noise token module that is composed of a set of trainable neural noise templates to dynamically encode the noise information and thus enrich a DNN's generalization. Experimental results show that the noise token module was effective across various neural architectures and contributed to higher performance growth with increasing noise diversity.

Issue 2: When reconstructing speech signals, how to alleviate the phase distortion introduced by inverse STFT?

Instead of using inverse STFT, we proposed a neural vocoder-based waveform generation module to directly generate speech signals from a mel-spectrogram, which avoids the use of noisy phases. Experiments found that the neural vocoder-based model improved the listening quality of the generated speech.

Issue 3: How to improve the noise reduction performance for device-degraded speech?

We directly modeled the joint degradation effect of device-degraded speech, which included not only additive noise but also reverberation and the bad frequency response of a microphone. We proposed an encoder-decoder neural network to automatically transform device-degraded speech into high-quality speech. Specifically, we first filtered out the channel characteristics of input speech and then predicted a target high-quality mel-spectrogram by assigning a high-quality recording as a reference. We used neural vocoder to synthesize the final waveform. Experimental results show that the proposed method worked well and outperformed several state-of-the-art baselines in terms of listening quality.

Issue 4: How to improve the performance of intelligibility boosting by leveraging deep learning?

We proposed a novel neural intelligibility boosting method by using generative adversarial networks. To overcome the lack of ground-truth labels, the network was trained to approximate and mimic the behavior of speech intelligibility metrics. The intelligibility-boosting module then modified speech signals in such a way as to maximize speech metric scores under the guidance of a learned surrogate. Experimental results from both objective measurements and large-scale listening tests indicate that the proposed method achieved significant intelligibility gains and performed much better than the compared baselines with a small model size.

Issue 5: How to integrate noise reduction with intelligibility boosting for a full-end speech communication scenario where noise exists in both speaker and listener environments?

We proposed a DNN-based joint framework. Under this framework, noise reduction (NR) and intelligibility boosting (IB) modules can be jointly optimized, where the NR module suppresses the noise of the input noisy speech, and the IB module further improves its intelligibility. Experiments found that the enhanced speech could be less noisy and more intelligible. They also showed that the joint framework achieved a significant intelligibility gain while preserving speech quality well, and it consistently outperformed the disjoint processing pipelines by a large margin.

Future directions

This thesis investigated improving neural network-based speech enhancement. In addition to the methods discussed in each chapter, there are a number of future directions that could continue to improve the performance of speech enhancement.

- In Chapter 3, we used the WaveRNN vocoder to generate speech waveforms. However, some generated samples were seriously distorted, resulting in muffled voice. This was probably caused by the autoregressive mechanism of the WaveRNN vocoder, where the previous bad samples have negative effects on future predictions. The inference speed was also slow since the speech waveforms are generated sample by sample. One future direction is to try a non-autoregressive neural vocoder, such as HiFi-GAN [122]. Since a conditioned mel-spectrogram may contain noise, it is also worth studying how to adapt the vocoder to make it more robust to mel-spectrogram distortion.

- In Chapter 4, we filtered out the channel characteristics of input signals by using an adversarial network classifier. However, there may still exist residual channel information on the encoder representations. Recently, mutual information (MI) minimization [123, 124] has been shown to have a powerful ability for extracting disentangled features. It would be interesting to investigate the use of MI minimization technique.

- In Chapter 5, we proposed a noise-aware neural intelligibility boosting system while disregarding reverberation. We attempted to incorporate a reverberation effect into the system in Appendix B but failed. Therefore, reverberation modeling is still a challenging problem. Besides, we modified the spectrogram magnitude and simply used the phase of the input clean signal to reconstruct enhanced speech. To incorporate phase estimation, in the future, we will consider estimating real and imaginary spectrograms of enhanced speech simultaneously. Speaking style (e.g., prosody and/or duration) modification via voice conversion approaches [125, 126] is also a possible direction for intelligibility boosting.

- In Chapter 6, we proposed a joint framework integrating noise reduction with intelligibility boosting to address full-end speech enhancement. However, as the far-end SNR levels (on the speaker side) become lower, the task becomes much more challenging. Very recently, Shifas *et al.* [119] proposed directly mapping noisy speech to an intelligibility-boosted target with only a single network. In the future, we will look into this end-to-end method and further update our model.

- All speech enhancement models proposed in this thesis require using only a single-channel microphone. They can be further extended to the use of multi-channel microphone arrays to benefit from spatial information.

A Device-degraded Speech Dataset

A.1 Motivation

Training a data-driven speech enhancement model requires large amount of data (i.e., pairs of clean speech and noisy counterpart), but the existing datasets are relatively smaller compared to those used in other domains such as image classification [127]. Also, most datasets consist of only synthetic noisy speech rather than real noisy recordings. Although noisy speech can be obtained easily enough by adding clean speech with random noise segments [86, 128] or convolving with room impulse responses [129], Reddy *et al.* [130] pointed out that models trained on synthetic datasets often degrade significantly on real recordings. This is mostly because the realistic device degradation cannot be perfectly simulated by synthetic datasets. For example, the measured transfer functions cannot capture the nonlinear reverberation and nonlinear distortion of microphone occurred real-world recording.

Given this background, to better facilitate the research on speech enhancement,

especially the noise reduction for device-degraded speech (discussed in Chapter 4), we collected and released a new dataset consisting of realistic device-degraded speech, named *DDS*.

DDS contains real recordings that are collected in diverse realistic environments using various microphone devices. Specifically, DDS is built on top of two existing datasets: DAPS [91] and VCTK [88]. We play clean speech recordings (four hours from DAPS and eight hours from VCTK) and re-record waveforms in nine environments (two offices, two conference rooms, three working studios[1], one living room, and one waiting room) on three different devices (one MEMS and two condenser microphones), producing 27 different recording conditions. For each condition, recordings are conducted with six microphone positions to simulate different noise and reverberation levels. In total, DDS contains 1,944 hours (3 devices $\times$ 9 environments $\times$ 6 positions $\times$ 12 hours) of real recordings.

As far as we are concerned, this is the largest public dataset comprehensively covering various recording factors (i.e., environment, device, and position). In addition to the study of speech enhancement, it can be used in research domains such as domain adaptation in automatic speech recognition (ASR) [131], text-to-speech (TTS) from found voice data [132], and replay spoof detection in automatic speaker verification (ASV) [133]. The dataset is publicly available online: https://doi.org/10.5281/zenodo.5464104.

A.2 Dataset overview

In this section, we explain how we collected the DDS dataset and conduct an initial analysis. Table A.1 gives an overview of the dataset settings.

Speech materials

Clean speech materials are selected from the DAPS [91] and VCTK [88] datasets, which both contain professional voice recordings. Specifically, the DAPS portion has four hours of speech data consisting of 20 speakers (ten female and ten male),

[1]Specifically: a photo studio, a capture studio, and a voice studio.

Table A.1: Overview of dataset settings. *MEMS* and *condenser* are microphone types. For device position, parameter *(distance, angle)* denotes the distance and angle between device and sound source, respectively.

Setting	Count	Description
Speech materials	2	DAPS, VCTK clean sets
Environments	9	conference rooms (2), offices (2), studios (3), living room (1), waiting room (1)
Devices	3	iPad Air (MEMS), Uber Mic (condenser), MPM-1000 (condenser)
Device positions	6	A(50 cm, 0°), B(100 cm, 15°) C(125 cm, 30°), D(150 cm, 45°) E(175 cm, 60°), F(200 cm, 75°)

and the VCTK portion has eight hours[2] of speech data consisting of 28 speakers (14 female and 14 male). As shown in Fig. A.1, we played and recorded speech using devices at a sampling rate of 48 kHz. To avoid the probable bias caused by the loudspeaker characteristics, we used a high-quality coaxial monitor speaker (Presonus Sceptre S6[3]) with very nice flat frequency response. For the DAPS portion, we re-sampled speech files into 44.1 kHz to match the original sampling rate of the DAPS clean set. Finally, we applied a cross-correlation algorithm to align the recorded speech with the original clean speech.

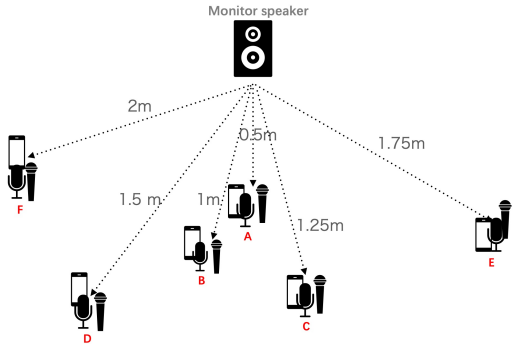
Environments

All recordings were conducted in realistic rooms[4]. We selected a total of nine rooms with different layouts and sizes: two conference rooms, two offices, three studios, one living room, and one waiting room. Each room had a certain level of environmental noise and reverberation. It is worth noting that there is no

[2]We only selected part of VCTK speech instead of using the entire set.

[3]https://www.presonus.com/products/sceptre-s6

[4]Details of room information (e.g., room size) and text scripts are included in the released DDS dataset.



(a) Schematic diagram of recording setup

(b) Example of device recording in the *livingroom1*

Figure A.1: Recording setup. Under each environment, studio-quality speech is played through a monitor loudspeaker and re-recorded on three devices (iPad Air, Uber Mic, and MPM-1000) at six (A–E) positions.

constraint on the room noise. For example, the noise collected during recording may contain the sound of air conditioner, computer fans, or outdoor noise. We expect such background noise is close to that occurred in real-world recording, e.g., in home and office.

Devices and recording positions

Table A.1 lists the three microphone devices used during recording. These were a micro-electromechanical system (MEMS)-processed microphone, which is of small size and commonly embedded in smart devices, and two condenser microphones, which can offer a better sound quality than the MEMS microphones.

In addition to recording device, we conducted multiple recordings at six different positions for each device in each environment. The closest position was set to 50 cm directly in front of the speaker, while the farthest was set to 200 cm and at 75°angle from the speaker. In this manner, we collected replayed speech with various noise and reverberation levels for each recording condition.

Summary of DDS dataset

In total, the DDS dataset consists of 9 environment settings and 3 device settings, resulting in a total of 27 recording conditions. Each condition consists of 83,058

Table A.2: Average PESQ and ESTOI scores in different environments.

Environment	DAPS portion		VCTK portion	
	PESQ	ESTOI	PESQ	ESTOI
confroom1	2.34	0.715	2.58	0.630
confroom2	1.98	0.617	2.27	0.527
office1	2.60	0.758	2.80	0.660
office2	2.31	0.724	2.54	0.627
studio1	2.37	0.725	2.59	0.602
studio2	3.01	0.815	3.10	0.735
studio3	3.10	0.811	3.16	0.735
waitingroom1	3.02	0.796	3.13	0.722
livingroom1	2.34	0.723	2.61	0.647

speech files (13,843 files × 6 positions) at sampling rates of 44.1 kHz (for the DAPS portion) and 48 kHz (for the VCTK portion).

A.3 Initial analysis of DDS

Last, we conducted an initial analysis to investigate the effects of the various environments and devices on recording quality. We used PESQ [69] and ESTOI [73] measures to evaluate objective speech quality and intelligibility, respectively. Tables A.2, A.3, and A.4 list the average scores under different conditions of environment, device, and position, respectively.

We can clearly see that all recording factors dramatically affect speech quality and intelligibility. For example, as shown in Table A.2, recording quality is directly related to room environment. Table A.3 shows that the condenser microphones (Uber Mic and MPM-1000) can offer a better sound quality than the MEMS one (iPad). Table A.4 shows that speech recorded at a closer position has a better quality. In summary, these results indicate that DDS provides a sufficiently large variation of speech data to comprehensively cover common recording factors.

Table A.3: Average PESQ and ESTOI scores for different devices.

Device	DAPS portion		VCTK portion	
	PESQ	ESTOI	PESQ	ESTOI
iPad	2.35	0.688	2.56	0.585
Uber Mic	2.66	0.767	2.85	0.684
MPM-1000	2.68	0.773	2.86	0.693

Table A.4: Average PESQ and ESTOI scores with different device positions.

Device position	DAPS portion		VCTK portion	
	PESQ	ESTOI	PESQ	ESTOI
A (50cm, 0°)	3.22	0.901	3.32	0.840
B (100cm, 15°)	2.77	0.810	2.94	0.728
C (125cm, 30°)	2.57	0.770	2.78	0.680
D (150cm, 45°)	2.44	0.720	2.65	0.624
E (175cm, 60°)	2.27	0.656	2.50	0.557
F (200cm, 75°)	2.11	0.597	2.35	0.495

B

Reverberation Modeling for Intelligibility Boosting

B.1 Reverberation modeling

We attempted to incorporate the reverberation effect into the intelligibility boosting task (the topic of Chapter 5).

For intelligibility boosting, our goal is to modify a clean speech signal to make it sound clearer under noise and reverberation. However, unlike additive noise that is independent of clean speech, the modification to speech affects the reverberation effect. Following the derivation in [43], early reverberation is disregarded, and late reverberation can be modeled as:

$$\sigma_L^2(m,i) = \rho^2(1-a^{2N}) \times \sum_{b=0}^{B-1} a^{2bR}\alpha^2(m-n_0-bR,i)\sigma_S^2(m-n_0-bR,i), \quad \text{(B.1)}$$

where $\sigma_L^2(m,i)$ is late reverberation variance at the m-th frame and i-th band, σ_S^2 is input speech variance, R is hop size, N is window size, n_0 denotes the sample index from which the late reverberation of the impulse response starts (typically 50 ms after the impulse response peak), B acts as a frame index to which the late reverberation effect ends, and $\alpha(m,i)$ denotes the amplification factor (see Section 5.1) used to redistribute the speech energy across time and frequency bands. The remaining a and ρ^2 are two reverberation-related parameters. a is a damping factor defined by the Polack model [134] in Equation (B.2), and ρ^2 is the diffuse response energy given in Equation (B.3).

$$a = 10^{-\frac{3}{T_{60} f_s}}, \tag{B.2}$$

$$\rho^2 = \sum_{l=n_0}^{+\infty} E[h^2(l)] \frac{1}{f_s}, \tag{B.3}$$

where T_{60} denotes reverberation time, f_s denotes sampling rate, and $h(l)$ denotes the room impulse response. By using Equation (B.1), late reverberation can be represented by a convolution between room-related parameters, i.e., $\rho^2(1-a^{2N})a^{2bR}$, and the modified speech variance, i.e., $\alpha^2(m-n_0-bR,i)\sigma_S^2(m-n_0-bR,i)$. The same as [43], we assume T_{60} and $\sum_{l=n_0}^{+\infty} E[h^2(l)]$ to be known, such that a and ρ^2 can be computed.

To enable the neural intelligibility boosting system to be reverberation-aware, we alter both the generator (G) and intelligibility discriminator (D_{int}). Specifically, we concatenate a and ρ^2 and repeat this 32 times to produce a 64-dimensional reverberation-related vector, which is then fed into G as an additional input feature. G outputs the amplification factor α, and then the modified speech variance $\alpha^2\sigma_S^2$ can be obtained. Next, we compute the late reverberation variance σ_L^2 using Equation (B.1). Finally, σ_L^2 is fed into D_{int} as additional input, such that D_{int} can predict the intelligibility score under reverberant conditions.

B.2 Experiments

We conducted experiments to verify if reverberation modeling works for the intelligibility boosting task. We used **Proposed (S+H+E)** (see Section 5.3.4)

Table B.1: Average intelligibility scores for different systems.

System	Intelligibility		
	SIIB	HASPI	ESTOI
Base w/o reverb	**31.89**	**2.69**	**0.265**
Base with reverb	25.29	2.34	0.238

as the **Base** system, in which quality metrics were neglected for simplicity. We extended the original training set (see in Section 5.3.1) by adding reverberation. Specifically, we selected four room impulse responses from an external dataset [115], with reverberation time T_{60} ranging from 500 to 950 ms. The test set remained the same, which contained three unseen reverberation types from weak to severe reverberation. We followed the same implementation recipe (as described in Section 5.3.2) to train the system.

We compared the systems with and without reverberation modeling. Table B.1 reports the objective intelligibility scores, where the scores were averaged over the whole test set including the three tested reverberation levels. As can be seen, the reverberation modeling used in this Appendix did not help improve the intelligibility boosting performance for reverberant conditions. Therefore, how to model reverberation for the intelligibility boosting task is still an unsolved problem.

C

Online Resources

This appendix lists the attached resources of the thesis.

Chapter 3

- Audio samples: https://nii-yamagishilab.github.io/samples-NTs

Chapter 4

- Audio samples: https://nii-yamagishilab.github.io/hyli666-demos/evr-slt2021

Chapter 5

- Source codes: https://github.com/nii-yamagishilab/NELE-GAN
- Audio samples: https://nii-yamagishilab.github.io/hyli666-demos/intelligibility

Chapter 6

- Audio samples: https://nii-yamagishilab.github.io/hyli666-demos/full-end-se

Appendix A

- DDS dataset: https://zenodo.org/record/5464104

D

List of Publications

Journal paper

1. Haoyu Li and Junichi Yamagishi, "Multi-Metric Optimization Using Generative Adversarial Networks for Near-End Speech Intelligibility Enhancement," in IEEE/ACM Transactions on Audio, Speech, and Language Processing, vol. 29, pp. 3000-3011, 2021, doi: 10.1109/TASLP.2021.3111566.

International conference paper

1. Haoyu Li, Yun Liu and Junichi Yamagishi, "Joint Noise Reduction and Listening Enhancement for Full-End Speech Enhancement," Under review, submitted to SLT 2022.

2. Haoyu Li and Junichi Yamagishi, "DDS: A New Device-Degraded Speech Dataset for Speech Enhancement," Accepted by Interspeech 2022.

3. Haoyu Li, Yang Ai and Junichi Yamagishi, "Enhancing Low-Quality Voice

Recordings Using Disentangled Channel Factor and Neural Waveform Model," 2021 IEEE Spoken Language Technology Workshop (SLT), Shenzhen, China, 2021, pp. 734-741.

4. Haoyu Li, Szu-Wei Fu, Yu Tsao and Junichi Yamagishi, "iMetricGAN: Intelligibility Enhancement for Speech-in-Noise Using Generative Adversarial Network-Based Metric Learning," Proc. Interspeech 2020, 1336-1340.

5. Haoyu Li and Junichi Yamagishi, "Noise Tokens: Learning Neural Noise Templates for Environment-Aware Speech Enhancement," Proc. Interspeech 2020, 2452-2456.

6. Yang Ai, Haoyu Li, Xing Wang, Junichi Yamagishi and Zhen-hua Ling, "Denoising-and-Dereverberation Hierarchical Neural Vocoder for Robust Waveform Generation," 2021 IEEE Spoken Language Technology Workshop (SLT), Shenzhen, China, 2021, pp. 477-484.

7. Yi Zhao, Haoyu Li, Cheng-I Lai, Jennifer Williams, Erica Cooper and Junichi Yamagishi, "Improved Prosody from Learned F0 Codebook Representations for VQ-VAE Speech Waveform Reconstruction," Proc. Interspeech 2020, 4417-4421.

Bibliography

[1] Karl S Pearsons, Ricarda L Bennett, and Sanford A Fidell. *Speech levels in various noise environments*. Office of Health and Ecological Effects, Office of Research and Development, US EPA, 1977.

[2] SN Graetzer, Jon Barker, Trevor J Cox, Michael Akeroyd, John F Culling, Graham Naylor, Eszter Porter, R Viveros Munoz, et al. Clarity-2021 challenges: Machine learning challenges for advancing hearing aid processing. In *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, volume 2, pages 686–690. International Speech Communication Association (ISCA), 2021.

[3] Jean-Marc Valin. A hybrid DSP/deep learning approach to real-time full-band speech enhancement. In *2018 20th International Workshop on Multimedia Signal Processing (MMSP)*, pages 1–5. IEEE, 2018.

[4] Felix Weninger, Hakan Erdogan, Shinji Watanabe, Emmanuel Vincent, Jonathan Le Roux, John R Hershey, and Björn Schuller. Speech enhancement with LSTM recurrent neural networks and its application to noise-robust ASR. In *International Conference on Latent Variable Analysis and Signal Separation*, pages 91–99. Springer, 2015.

[5] Pinaki Shankar Chanda and Sungjin Park. Speech intelligibility enhancement using tunable equalization filter. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, volume 4, pages 613–616. IEEE, 2007.

[6] Sander J van Wijngaarden and Jan A Verhave. Prediction of speech intelligibility for public address systems in traffic tunnels. *applied acoustics*, 67(4):306–323, 2006.

[7] Kuldip Paliwal, Kamil Wójcicki, and Benjamin Shannon. The importance of phase in speech enhancement. *Speech Communication*, 53(4):465–494, 2011.

[8] Yong Xu, Jun Du, Li-Rong Dai, and Chin-Hui Lee. Dynamic noise aware training for speech enhancement based on deep neural networks. In *Fifteenth Annual Conference of the International Speech Communication Association*, 2014.

[9] Philipos C Loizou. *Speech enhancement: theory and practice*. CRC press, 2007.

[10] Steven Boll. Suppression of acoustic noise in speech using spectral subtraction. *IEEE Transactions on acoustics, speech, and signal processing*, 27(2):113–120, 1979.

[11] Israel Cohen and Baruch Berdugo. Noise estimation by minima controlled recursive averaging for robust speech enhancement. *IEEE signal processing letters*, 9(1):12–15, 2002.

[12] Israel Cohen. Noise spectrum estimation in adverse environments: Improved minima controlled recursive averaging. *IEEE Transactions on speech and audio processing*, 11(5):466–475, 2003.

[13] Sundarrajan Rangachari and Philipos C Loizou. A noise-estimation algorithm for highly non-stationary environments. *Speech communication*, 48(2):220–231, 2006.

[14] Meng Sun, Yinan Li, Jort F Gemmeke, and Xiongwei Zhang. Speech enhancement under low SNR conditions via noise estimation using sparse and low-rank NMF with Kullback–Leibler divergence. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 23(7):1233–1242, 2015.

[15] Jingdong Chen, Jacob Benesty, Yiteng Huang, and Simon Doclo. New insights into the noise reduction Wiener filter. *IEEE Transactions on audio, speech, and language processing*, 14(4):1218–1234, 2006.

[16] Yariv Ephraim and David Malah. Speech enhancement using a minimum-mean square error short-time spectral amplitude estimator. *IEEE Transactions on acoustics, speech, and signal processing*, 32(6):1109–1121, 1984.

[17] Yariv Ephraim and David Malah. Speech enhancement using a minimum mean-square error log-spectral amplitude estimator. *IEEE transactions on acoustics, speech, and signal processing*, 33(2):443–445, 1985.

[18] Robert Gray, Andrés Buzo, Augustine Gray, and Yasuo Matsuyama. Distortion measures for speech processing. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 28(4):367–376, 1980.

[19] Yong Xu, Jun Du, Li-Rong Dai, and Chin-Hui Lee. A regression approach to speech enhancement based on deep neural networks. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 23(1):7–19, 2014.

[20] Xugang Lu, Yu Tsao, Shigeki Matsuda, and Chiori Hori. Speech enhancement based on deep denoising autoencoder. In *Interspeech*, pages 436–440, 2013.

[21] Anurag Kumar and Dinei Florencio. Speech enhancement in multiple-noise conditions using deep neural networks. *arXiv preprint arXiv:1605.02427*, 2016.

[22] Tian Gao, Jun Du, Li-Rong Dai, and Chin-Hui Lee. Densely connected progressive learning for lstm-based speech enhancement. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5054–5058. IEEE, 2018.

[23] Szu-Wei Fu, Yu Tsao, Xugang Lu, and Hisashi Kawai. Raw waveform-based speech enhancement by fully convolutional networks. In *2017 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, pages 006–012. IEEE, 2017.

[24] Se Rim Park and Jinwon Lee. A fully convolutional neural network for speech enhancement. *arXiv preprint arXiv:1609.07132*, 2016.

[25] Jaeyoung Kim, Mostafa El-Khamy, and Jungwon Lee. T-GSA: Transformer with Gaussian-weighted self-attention for speech enhancement. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6649–6653. IEEE, 2020.

[26] Quan Wang, Hannah Muckenhirn, Kevin Wilson, Prashant Sridhar, Zelin Wu, John Hershey, Rif A Saurous, Ron J Weiss, Ye Jia, and Ignacio Lopez Moreno. Voicefilter: Targeted voice separation by speaker-conditioned spectrogram masking. *arXiv preprint arXiv:1810.04826*, 2018.

[27] Ke Tan and DeLiang Wang. Complex spectral mapping with a convolutional recurrent network for monaural speech enhancement. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6865–6869. IEEE, 2019.

[28] Anmol Gulati, James Qin, Chung-Cheng Chiu, Niki Parmar, Yu Zhang, Jiahui Yu, Wei Han, Shibo Wang, Zhengdong Zhang, Yonghui Wu, et al. Conformer: Convolution-augmented transformer for speech recognition. *arXiv preprint arXiv:2005.08100*, 2020.

[29] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep learning*. MIT press, 2016.

[30] DeLiang Wang. On ideal binary mask as the computational goal of auditory scene analysis. In *Speech separation by humans and machines*, pages 181–197. Springer, 2005.

[31] Yuxuan Wang, Arun Narayanan, and DeLiang Wang. On training targets for supervised speech separation. *IEEE/ACM transactions on audio, speech, and language processing*, 22(12):1849–1858, 2014.

[32] DeLiang Wang and Jitong Chen. Supervised speech separation based on deep learning: An overview. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 26(10):1702–1726, 2018.

[33] Aaron van den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, Andrew Senior, and Koray Kavukcuoglu. Wavenet: A generative model for raw audio. *arXiv preprint arXiv:1609.03499*, 2016.

[34] Dario Rethage, Jordi Pons, and Xavier Serra. A wavenet for speech denoising. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5069–5073. IEEE, 2018.

[35] Jiaqi Su, Zeyu Jin, and Adam Finkelstein. HiFi-GAN: High-fidelity denoising and dereverberation based on speech deep features in adversarial networks. *arXiv preprint arXiv:2006.05694*, 2020.

[36] Jiaqi Su, Adam Finkelstein, and Zeyu Jin. Perceptually-motivated environment-specific speech enhancement. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7015–7019. IEEE, 2019.

[37] Yi Luo and Nima Mesgarani. Tasnet: time-domain audio separation network for real-time, single-channel speech separation. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 696–700. IEEE, 2018.

[38] Yi Luo and Nima Mesgarani. Conv-tasnet: Surpassing ideal time–frequency magnitude masking for speech separation. *IEEE/ACM transactions on audio, speech, and language processing*, 27(8):1256–1266, 2019.

[39] Donald S Williamson, Yuxuan Wang, and DeLiang Wang. Complex ratio masking for monaural speech separation. *IEEE/ACM transactions on audio, speech, and language processing*, 24(3):483–492, 2015.

[40] Tudor-Catalin Zorila, Varvara Kandia, and Yannis Stylianou. Speech-in-noise intelligibility improvement based on spectral shaping and dynamic range compression. In *Proc. Interspeech*, pages 635–638, 2012.

[41] Carol Chermaz and Simon King. A Sound Engineering Approach to Near End Listening Enhancement. In *Proc. Interspeech*, pages 1356–1360, 2020.

[42] Yan Tang and Martin Cooke. Optimised spectral weightings for noise-dependent speech intelligibility enhancement. In *Proc. Interspeech*, pages 955–958, 2012.

[43] Richard C Hendriks, João B Crespo, Jesper Jensen, and Cees H Taal. Optimal near-end speech intelligibility improvement incorporating additive noise and late reverberation under an approximation of the short-time SII. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 23(5):851–862, 2015.

[44] Cees H Taal, Jesper Jensen, and Arne Leijon. On optimal linear filtering of speech for near-end listening enhancement. *IEEE Signal Processing Letters*, 20(3):225–228, 2013.

[45] Yan Tang and Martin Cooke. Learning static spectral weightings for speech intelligibility enhancement in noise. *Computer Speech & Language*, 49:1–16, 2018.

[46] Cassia Valentini-Botinhao, Junichi Yamagishi, Simon King, and Ranniery Maia. Intelligibility enhancement of HMM-generated speech in additive noise by modifying Mel cepstral coefficients to increase the glimpse proportion. *Computer Speech & Language*, 28(2):665–686, 2014.

[47] Martin Cooke, Catherine Mayo, and Cassia Valentini-Botinhao. Intelligibility-Enhancing Speech Modifications: the Hurricane Challenge. In *Proc. Interspeech*, pages 3552–3556, 2013.

[48] Jan Rennies, Henning Schepker, Cassia Valentini-Botinhao, and Martin Cooke. Intelligibility-Enhancing Speech Modifications — The Hurricane Challenge 2.0. In *Proc. Interspeech*, pages 1341–1345, 2020.

[49] Jean-Claude Junqua. The Lombard reflex and its role on human listeners and automatic speech recognizers. *The Journal of the Acoustical Society of America*, 93(1):510–524, 1993.

[50] Youyi Lu and Martin Cooke. Speech production modifications produced by competing talkers, babble, and stationary noise. *The Journal of the Acoustical Society of America*, 124(5):3261–3275, 2008.

[51] Sungyub D Yoo, J Robert Boston, Amro El-Jaroudi, Ching-Chung Li, John D Durrant, Kristie Kovacyk, and Susan Shaiman. Speech signal modification to increase intelligibility in noisy environments. *The Journal of the Acoustical Society of America*, 122(2):1138–1149, 2007.

[52] Karan Nathwani, Gaël Richard, Bertrand David, Pierre Prablanc, and Vincent Roussarie. Speech intelligibility improvement in car noise environment by voice transformation. *Speech Communication*, 91:17–27, 2017.

[53] Ana Ramírez López, Shreyas Seshadri, Lauri Juvela, Okko Räsänen, and Paavo Alku. Speaking Style Conversion from Normal to Lombard Speech Using a Glottal Vocoder and Bayesian GMMs. In *Proc. Interspeech*, pages 1363–1367, 2017.

[54] Shreyas Seshadri, Lauri Juvela, Paavo Alku, Okko Räsänen, et al. Augmented CycleGANs for Continuous Scale Normal-to-Lombard Speaking Style Conversion. In *Interspeech*, pages 2838–2842, 2019.

[55] Shreyas Seshadri, Lauri Juvela, Okko Räsänen, and Paavo Alku. Vocal effort based speaking style conversion using vocoder features and parallel learning. *IEEE Access*, 7:17230–17246, 2019.

[56] Manu Airaksinen, Bajibabu Bollepalli, Lauri Juvela, Zhizheng Wu, Simon King, and Paavo Alku. GlottDNN-a full-band glottal vocoder for statistical parametric speech synthesis. In *Interspeech*, volume 9, pages 2473–2477, 2016.

[57] Hideki Kawahara. STRAIGHT, exploitation of the other aspect of VOCODER: Perceptually isomorphic decomposition of speech sounds. *Acoustical science and technology*, 27(6):349–353, 2006.

[58] Gilles Degottex, Pierre Lanchantin, and Mark Gales. A log domain pulse model for parametric speech synthesis. *IEEE/ACM Transactions on audio, speech, and language processing*, 26(1):57–70, 2017.

[59] Shreyas Seshadri, Lauri Juvela, Junichi Yamagishi, Okko Räsänen, and Paavo Alku. Cycle-consistent adversarial networks for non-parallel vocal effort based speaking style conversion. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6835–6839. IEEE, 2019.

[60] American National Standards Institute. *American National Standard: Methods for Calculation of the Speech Intelligibility Index*. Acoustical Society of America, 1997.

[61] W Bastiaan Kleijn and Richard C Hendriks. A simple model of speech communication and its application to intelligibility enhancement. *IEEE Signal Processing Letters*, 22(3):303–307, 2014.

[62] Martin Cooke. A glimpsing model of speech perception in noise. *The Journal of the Acoustical Society of America*, 119(3):1562–1573, 2006.

[63] John Henry Holland. *Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence*. MIT press, 1992.

[64] Steven Van Kuyk, W Bastiaan Kleijn, and Richard Christian Hendriks. An evaluation of intrusive instrumental intelligibility metrics. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 26(11):2153–2166, 2018.

[65] James M Kates and Kathryn H Arehart. The hearing-aid speech perception index (haspi) version 2. *Speech Communication*, 131:35–46, 2021.

[66] James M Kates and Kathryn H Arehart. The Hearing-Aid Speech Perception Index (HASPI). *Speech Communication*, 65:75–93, 2014.

[67] Steven Van Kuyk, W Bastiaan Kleijn, and Richard C Hendriks. An instrumental intelligibility metric based on information theory. *IEEE Signal Processing Letters*, 25(1):115–119, 2017.

[68] Yi Hu and Philipos C Loizou. Evaluation of objective quality measures for speech enhancement. *IEEE Transactions on audio, speech, and language processing*, 16(1):229–238, 2007.

[69] Antony W Rix, John G Beerends, Michael P Hollier, and Andries P Hekstra. Perceptual evaluation of speech quality (PESQ)-a new method for speech quality assessment of telephone networks and codecs. In *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, volume 2, pages 749–752. IEEE, 2001.

[70] Michael Chinen, Felicia SC Lim, Jan Skoglund, Nikita Gureev, Feargus O'Gorman, and Andrew Hines. ViSQOL v3: An open source production ready objective speech and audio metric. In *2020 Twelfth International Conference on Quality of Multimedia Experience (QoMEX)*, pages 1–6. IEEE, 2020.

[71] Abigail A Kressner, David V Anderson, and Christopher J Rozell. Evaluating the generalization of the hearing aid speech quality index (HASQI). *IEEE transactions on audio, speech, and language processing*, 21(2):407–415, 2012.

[72] Cees H Taal, Richard C Hendriks, Richard Heusdens, and Jesper Jensen. A short-time objective intelligibility measure for time-frequency weighted noisy speech. In *2010 IEEE international conference on acoustics, speech and signal processing*, pages 4214–4217. IEEE, 2010.

[73] Jesper Jensen and Cees H Taal. An algorithm for predicting the intelligibility of speech masked by modulated noise maskers. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 24(11):2009–2022, 2016.

[74] Kurt Steinmetzger, Johannes Zaar, Helia Relaño-Iborra, Stuart Rosen, and Torsten Dau. Predicting the effects of periodicity on the intelligibility of masked speech: An evaluation of different modelling approaches and their

limitations. *The Journal of the Acoustical Society of America*, 146(4):2562–2576, 2019.

[75] Søren Jørgensen, Stephan D Ewert, and Torsten Dau. A multi-resolution envelope-power based model for speech intelligibility. *The Journal of the Acoustical Society of America*, 134(1):436–446, 2013.

[76] EH Rothauser. IEEE recommended practice for speech quality measurements. *IEEE Transactions on Audio and Electroacoustics*, 17:225–246, 1969.

[77] Yuxuan Wang, Daisy Stanton, Yu Zhang, RJ Skerry-Ryan, Eric Battenberg, Joel Shor, Ying Xiao, Fei Ren, Ye Jia, and Rif A Saurous. Style tokens: Unsupervised style modeling, control and transfer in end-to-end speech synthesis. *arXiv preprint arXiv:1803.09017*, 2018.

[78] Kevin Wilson, Michael Chinen, Jeremy Thorpe, Brian Patton, John Hershey, Rif A Saurous, Jan Skoglund, and Richard F Lyon. Exploring tradeoffs in models for low-latency speech enhancement. In *2018 16th International Workshop on Acoustic Signal Enhancement (IWAENC)*, pages 366–370. IEEE, 2018.

[79] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017.

[80] Nal Kalchbrenner, Erich Elsen, Karen Simonyan, Seb Noury, Norman Casagrande, Edward Lockhart, Florian Stimberg, Aaron van den Oord, Sander Dieleman, and Koray Kavukcuoglu. Efficient neural audio synthesis. *arXiv preprint arXiv:1802.08435*, 2018.

[81] Hyeong-Seok Choi, Jang-Hyun Kim, Jaesung Huh, Adrian Kim, Jung-Woo Ha, and Kyogu Lee. Phase-aware speech enhancement with deep complex U-Net. In *International Conference on Learning Representations*, 2018.

[82] Jaime Lorenzo-Trueba, Fuming Fang, Xin Wang, Isao Echizen, Junichi Yamagishi, and Tomi Kinnunen. Can we steal your vocal identity from

the internet?: Initial investigation of cloning Obama's voice using GAN, WaveNet and low-quality found data. *arXiv preprint arXiv:1803.00860*, 2018.

[83] Jesús Villalba, Nanxin Chen, David Snyder, Daniel Garcia-Romero, Alan McCree, Gregory Sell, Jonas Borgstrom, Leibny Paola García-Perera, Fred Richardson, Réda Dehak, et al. State-of-the-art speaker recognition with neural network embeddings in NIST SRE and speakers in the wild evaluations. *Computer Speech & Language*, page 101026, 2019.

[84] Erica Cooper, Cheng-I Lai, Yusuke Yasuda, Fuming Fang, Xin Wang, Nanxin Chen, and Junichi Yamagishi. Zero-shot multi-speaker text-to-speech with state-of-the-art neural speaker embeddings. *arXiv preprint arXiv:1910.10838*, 2019.

[85] Alex Graves and Navdeep Jaitly. Towards end-to-end speech recognition with recurrent neural networks. In *International conference on machine learning*, pages 1764–1772, 2014.

[86] Chandan KA Reddy, Ebrahim Beyrami, Jamie Pool, Ross Cutler, Sriram Srinivasan, and Johannes Gehrke. A scalable noisy speech dataset and online subjective test framework. *arXiv preprint arXiv:1909.08050*, 2019.

[87] Guoning Hu. 100 nonspeech environmental sounds. *The Ohio State University, Department of Computer Science and Engineering*, 2004.

[88] Junichi Yamagishi, Christophe Veaux, Kirsten MacDonald, et al. CSTR VCTK Corpus: English multi-speaker corpus for CSTR Voice Cloning Toolkit (version 0.92). 2019.

[89] John S Garofolo. TIMIT acoustic phonetic continuous speech corpus. *Linguistic Data Consortium, 1993*, 1993.

[90] Nadim Nachar et al. The Mann-Whitney U: A test for assessing whether two independent samples come from the same distribution. *Tutorials in quantitative Methods for Psychology*, 4(1):13–20, 2008.

[91] Gautham J Mysore. Can we automatically transform speech recorded on common consumer devices in real-world environments into professional production quality speech?—a dataset, insights, and challenges. *IEEE Signal Processing Letters*, 22(8):1006–1010, 2014.

[92] Tomohiro Nakatani, Takuya Yoshioka, Keisuke Kinoshita, Masato Miyoshi, and Biing-Hwang Juang. Speech dereverberation based on variance-normalized delayed linear prediction. *IEEE Transactions on Audio, Speech, and Language Processing*, 18(7):1717–1731, 2010.

[93] Ori Ernst, Shlomo E Chazan, Sharon Gannot, and Jacob Goldberger. Speech dereverberation using fully convolutional networks. In *26th European Signal Processing Conference (EUSIPCO)*, pages 390–394. IEEE, 2018.

[94] Vincent Verfaille, Udo Zolzer, and Daniel Arfib. Adaptive digital audio effects (A-DAFx): A new class of sound transformations. *IEEE Transactions on Audio, Speech, and Language Processing*, 14(5):1817–1831, 2006.

[95] Giovanni Pepe, Leonardo Gabrielli, Stefano Squartini, and Luca Cattani. Designing audio equalization filters by deep neural networks. *Applied Sciences*, 10(7):2483, 2020.

[96] Chien-Feng Liao, Yu Tsao, Hung-Yi Lee, and Hsin-Min Wang. Noise adaptive speech enhancement using domain adversarial training. *arXiv preprint arXiv:1807.07501*, 2018.

[97] Wei-Ning Hsu, Yu Zhang, Ron J Weiss, Yu-An Chung, Yuxuan Wang, Yonghui Wu, and James Glass. Disentangling correlated speaker and noise for speech synthesis via data augmentation and adversarial factorization. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5901–5905. IEEE, 2019.

[98] Jonathan Shen, Ruoming Pang, Ron J Weiss, Mike Schuster, Navdeep Jaitly, Zongheng Yang, Zhifeng Chen, Yu Zhang, Yuxuan Wang, Rj Skerrv-Ryan, et al. Natural TTS synthesis by conditioning wavenet on mel spectrogram predictions. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4779–4783. IEEE, 2018.

[99] Heiga Zen, Viet Dang, Rob Clark, Yu Zhang, Ron J Weiss, Ye Jia, Zhifeng Chen, and Yonghui Wu. LibriTTS: A corpus derived from LibriSpeech for text-to-speech. *arXiv preprint arXiv:1904.02882*, 2019.

[100] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[101] Jiaqi Su, Zeyu Jin, and Adam Finkelstein. Acoustic matching by embedding impulse responses. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 426–430. IEEE, 2020.

[102] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-SNE. *Journal of Machine Learning Research*, 9(Nov):2579–2605, 2008.

[103] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *Advances in neural information processing systems*, 27, 2014.

[104] Brian R Glasberg and Brian CJ Moore. Derivation of auditory filter shapes from notched-noise data. *Hearing research*, 47(1-2):103–138, 1990.

[105] Szu-Wei Fu, Tao-Wei Wang, Yu Tsao, Xugang Lu, and Hisashi Kawai. End-to-end waveform utterance enhancement for direct evaluation metrics optimization by fully convolutional neural networks. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 26(9):1570–1584, 2018.

[106] Jaeyoung Kim, Mostafa El-Kharmy, and Jungwon Lee. End-to-end multi-task denoising for joint SDR and PESQ optimization. *arXiv preprint arXiv:1901.09146*, 2019.

[107] Haoyu Li, Szu-Wei Fu, Yu Tsao, and Junichi Yamagishi. iMetricGAN: Intelligibility Enhancement for Speech-in-Noise Using Generative Adversarial Network-Based Metric Learning. In *Proc. Interspeech*, pages 1336–1340, 2020.

[108] Ashutosh Pandey and DeLiang Wang. TCNN: Temporal convolutional neural network for real-time speech enhancement in the time domain. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6875–6879. IEEE, 2019.

[109] Yuichiro Koyama, Tyler Vuong, Stefan Uhlich, and Bhiksha Raj. Exploring the best loss function for DNN-based low-latency speech enhancement with temporal convolutional networks. *arXiv preprint arXiv:2005.11611*, 2020.

[110] Min Lin, Qiang Chen, and Shuicheng Yan. Network in network. *arXiv preprint arXiv:1312.4400*, 2013.

[111] Takeru Miyato, Toshiki Kataoka, Masanori Koyama, and Yuichi Yoshida. Spectral normalization for generative adversarial networks. In *International Conference on Learning Representations*, 2018.

[112] Cassia Valentini-Botinhao, Cassie Mayo, and Martin Cooke. Hurricane natural speech corpus - higher quality version, 2019.

[113] Philippa Demonte. HARVARD speech corpus - audio recording 2019, 2019.

[114] Elior Hadad, Florian Heese, Peter Vary, and Sharon Gannot. Multichannel audio database in various acoustic environments. In *Proc. IWAENC*, pages 313–317, 2014.

[115] Marco Jeub, Magnus Schafer, and Peter Vary. A binaural room impulse response database for the evaluation of dereverberation algorithms. In *Proc. ICDSP*, pages 1–5, 2009.

[116] Markus Niermann, Peter Jax, and Peter Vary. Joint near-end listening enhancement and far-end noise reduction. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4970–4974. IEEE, 2017.

[117] Seyran Khademi, Richard C Hendriks, and W Bastiaan Kleijn. Intelligibility enhancement based on mutual information. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 25(8):1694–1708, 2017.

[118] Andreas Jonas Fuglsig, Jan Østergaard, Jesper Jensen, Lars Søndergaard Bertelsen, Peter Mariager, and Zheng-Hua Tan. Joint far-and near-end speech intelligibility enhancement based on the approximated speech intelligibility index. *arXiv preprint arXiv:2111.07759*, 2021.

[119] Muhammed P.V. Shifas, Tudor-Catalin Zorila, and Yannis Stylianou. End-to-end neural based modification of noisy speech for speech-in-noise intelligibility improvement. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 30:162–173, 2022.

[120] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016.

[121] Yanxin Hu, Yun Liu, Shubo Lv, Mengtao Xing, Shimin Zhang, Yihui Fu, Jian Wu, Bihong Zhang, and Lei Xie. DCCRN: Deep complex convolution recurrent network for phase-aware speech enhancement. *arXiv preprint arXiv:2008.00264*, 2020.

[122] Jungil Kong, Jaehyeon Kim, and Jaekyoung Bae. HiFi-GAN: Generative adversarial networks for efficient and high fidelity speech synthesis. *Advances in Neural Information Processing Systems*, 33:17022–17033, 2020.

[123] Eduardo Hugo Sanchez, Mathieu Serrurier, and Mathias Ortner. Learning disentangled representations via mutual information estimation. In *European Conference on Computer Vision*, pages 205–221. Springer, 2020.

[124] Detai Xin, Tatsuya Komatsu, Shinnosuke Takamichi, and Hiroshi Saruwatari. Disentangled speaker and language representations using mutual information minimization and domain adaptation for cross-lingual TTS. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6608–6612. IEEE, 2021.

[125] Carl Robinson, Nicolas Obin, and Axel Roebel. Sequence-to-sequence modelling of F0 for speech emotion conversion. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6830–6834. IEEE, 2019.

[126] Kun Zhou, Berrak Sisman, Rui Liu, and Haizhou Li. Seen and unseen emotional style transfer for voice conversion with a new emotional speech dataset. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 920–924. IEEE, 2021.

[127] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. IEEE, 2009.

[128] Yi Hu and Philipos C. Loizou. Subjective evaluation and comparison of speech enhancement algorithms. *Speech Communication*, 49:588–601, 2007.

[129] Cassia Valentini-Botinhao, Xin Wang, Shinji Takaki, and Junichi Yamagishi. Investigating RNN-based speech enhancement methods for noise-robust Text-to-Speech. In *SSW*, pages 146–152, 2016.

[130] Chandan KA Reddy, Vishak Gopal, Ross Cutler, Ebrahim Beyrami, Roger Cheng, Harishchandra Dubey, Sergiy Matusevych, Robert Aichner, Ashkan Aazami, Sebastian Braun, et al. The Interspeech 2020 deep noise suppression challenge: Datasets, subjective testing framework, and challenge results. *arXiv preprint arXiv:2005.13981*, 2020.

[131] Sining Sun, Binbin Zhang, Lei Xie, and Yanning Zhang. An unsupervised deep domain adaptation approach for robust speech recognition. *Neurocomputing*, 257:79–87, 2017.

[132] Shan Yang, Yuxuan Wang, and Lei Xie. Adversarial feature learning and unsupervised clustering based speech synthesis for found data with acoustic and textual noise. *IEEE Signal Processing Letters*, 27:1730–1734, 2020.

[133] Massimiliano Todisco, Xin Wang, Ville Vestman, Md Sahidullah, Héctor Delgado, Andreas Nautsch, Junichi Yamagishi, Nicholas Evans, Tomi Kinnunen, and Kong Aik Lee. ASVspoof 2019: Future horizons in spoofed and fake audio detection. *arXiv preprint arXiv:1904.05441*, 2019.

[134] Jean-Dominique Polack. *La transmission de l'énergie sonore dans les salles.* PhD thesis, Le Mans, 1988.