Master thesis on Sound and Music Computing

Universitat Pompeu Fabra

Speech Enhancement using the Wave-U-Net with Spectral Losses

Jose David Bedoya Molina

Supervisor: Jordi Janer

Co-Supervisor: Merlijn Blaauw

August 2020

Master thesis on Sound and Music Computing

Universitat Pompeu Fabra

Speech Enhancement using the Wave-U-Net with Spectral Losses

Jose David Bedoya Molina

Supervisor: Jordi Janer

Co-Supervisor: Merlijn Blaauw

August 2020

Contents

¿Quién lo creyera? Yorda es una golden retriever de 9 años. Tuvo cuatro hijos y mi padre se quedó con dos de ellos: Ani e Igo. Este par de hermanos cometieron el dulce pecado de juntarse, y tuvieron 9 cachorros. Mi aventura empieza prácticamente con la venta de tres de los nietos de Yorda. Con el producto de esa venta, mi padre cedió el dinero para pagarme el costoso examen que me habilitaba para esta maestría. Ahí empieza todo. Y aquí estoy. Gracias Ani, gracias Igo, gracias adorable Yorda.

Abstract

Speech enhancement and source separation are related tasks that aim to extract and/or improve a signal of interest from a recording that may involve sounds from various sources, reverberation, and/or degradation of capture quality. Taking into account that the Wave-U-Net is an end-to-end deep learning architecture that has obtained relevant results for the source separation task operating in the time domain, this thesis studies the performance of this architecture for the speech enhancement task in terms of denoising, dereverberation, decoloration, and bandwidth extension.

The experiments were conducted using a combination of a noisy version of the Voice Bank Corpus (VCTK) and the Device and Produced Speech dataset (DAPS). In addition to the original framework, variations inspired by relevant deep learning networks for speech enhancement were explored here, of which losses with spectral components presented the most favorable effects for the improvement of low-quality speech signals. Also, the concatenation of the input audio with a noise vector in the network was shown to generate more coherent high-frequency content in the output signal.

Keywords: Speech Enhancement; Wave-U-Net; Spectral Loss

Chapter 1

Introduction

1.1 Motivation

Enhancement means the improvement in the value or quality of something. When applied to speech, this simply means the improvement in intelligibility and/or quality of a degraded speech signal by using signal processing tools [1]. This enhancement is of great interest in a large number of systems that use speech for tasks such as speaker identification, speech recognition, transmission through communication channels, voice conversion, hearing aids, among others.

There are a bunch of classic techniques for addressing speech enhancement, however deep learning-based approaches have become the mainstream. Considering that the Wave-U-Net is an end-to-end deep learning architecture operating in the time domain that has obtained relevant results for source separation, which is a task related to speech enhancement [2], this thesis studies the performance of the Wave-U-Net with structural variations to simultaneously solve several targets of speech enhancement, namely: denoising, dereverberation, decoloration and bandwidth extension.

1.2 Objectives

- Study the performance of the Wave-U-Net end-to-end deep learning architecture for the speech enhancement task in terms of denoising, dereverberation, decoloration and bandwidth extension.
- Explore simple variations of the Wave-U-Net model inspired by some relevant deep learning frameworks for speech enhancement.
- Train and test models using speech enhancement datasets in which noisy samples are not only clean data mixed with various types of noise, but also feature reverberation from real-world acoustic environments and coloration from different capture devices.

1.3 Structure of the Report

The thesis is structured as follows: Chapter 2 presents a review of related work, focusing on relevant deep learning frameworks that serve as inspiration for the variants implemented in this research. Chapter 3 describes the raw datasets that were used. Data processing, training setup, baseline, and implemented variations are all defined in Chapter 4. In Chapter 5 the metrics used are presented and the results obtained are discussed. In Chapter 6 conclusions and future work are pointed out.

Chapter 2

Related Work

This chapter brings together several of the most promising frameworks for speech enhancement, which are outlined in Table 1. It can be seen that most of them are designed for the time domain, this is because this thesis pursues the implementation of a system in that domain. However, some relevant frameworks in the frequency domain, or that use both domains, are also considered. It can also be seen that the selected frameworks focus mainly on denoising or bandwidth extension. However, as in [3], their non-linear mapping functions can be trained to simultaneously solve several speech enhancement targets, including dereverbaration, decoloration, or others.

Framework		**Domain**	**Target**	**Loss**	**Year**
Wave-U-Net	[4]	Time	Denosing	$L2$	2018
Att. Wave-U-Net	[5]	Time	Denosing	$L1$, $L2$	2019
SEGAN+	[6]	Time	Denosing	GAN, $L1$	2019
MfNet	[7]	Time	BWE	GAN, Mel, $L2$	2020
3WSS-FFNet	[8]	Time	BWE	$L1$, Mel	2019
SSR-GAN	[9]	Freq.	BWE	GAN, LSD	2019
TFNet	[10]	Time, Freq.	BWE	$L2$	2018

Table 1: Relevant frameworks for speech enhancement.

2.1 Denoising

A major part of speech enhancement is the task of speech denoising [5], which is also referred to as noise suppression in the literature. Classic approaches for this task include spectral subtraction [11], Wiener filtering [12], statistical-based methods [13] such as the minimum mean squared error, and subspace algorithms [14, 15].

Neural networks have become the mainstream for speech enhancement. Recent widely used architectures typically work in the spectral domain, as with classic techniques, to learn a regression to the clean spectrum, typically in the form of a denoising auto-encoder [16, 17]. Other approaches work by predicting masks with deep neural networks that palliate noisy spectral regions [18, 3, 19]. Recurrent neural networks are also used, owing to their success in modeling sequential processes. Research shows that recurrent networks can predict a better contextualized set of frames or masks [20, 21, 22, 23]. The use of dropout, post-filtering, and perceptually motivated metrics is also effective. [24] propose to use a weighted denoising auto-encoder, altering the mean squared error loss function by assigning weighting factors to each spectral component. Furthermore, [25] use a loss function that considers the perceptual quality of speech, and [26] use an intelligibility loss to obtain better scores than those of plain regression losses. [3] use a deep neural network in the spectral domain, including the phase, by working with complex masks.

Convolutional neural networks are also known to perform well for locally correlated data, such as speech waveforms or spectrograms. As such, [27] have used them for one of the first speech enhancement systems working with the raw audio signal. Other contemporary studies use deep convolutional structures for this task in the form of regression architectures, such as the work by [28], who emphasize the need for reduction in model size, or the denoising WaveNet [29]. Other approaches use improvements in the adversarial setup for training stabilization [30, 31]. Moreover, adversarial losses have been used in the speech enhancement field to work without parallel corpora of aligned pairs [32].

2.1.1 Wave-U-Net [4, 33]

The Wave-U-Net is a one-dimensional adaptation of the U-Net architecture [34, 35]. Its original authors applied it to singing voice and multi-instrument separation [33]. Taking into account that speech enhancement is a task related to general audio source separation [2], [4] studied the Wave-U-Net architecture with the aim of separating noisy speech into clean speech and noise.

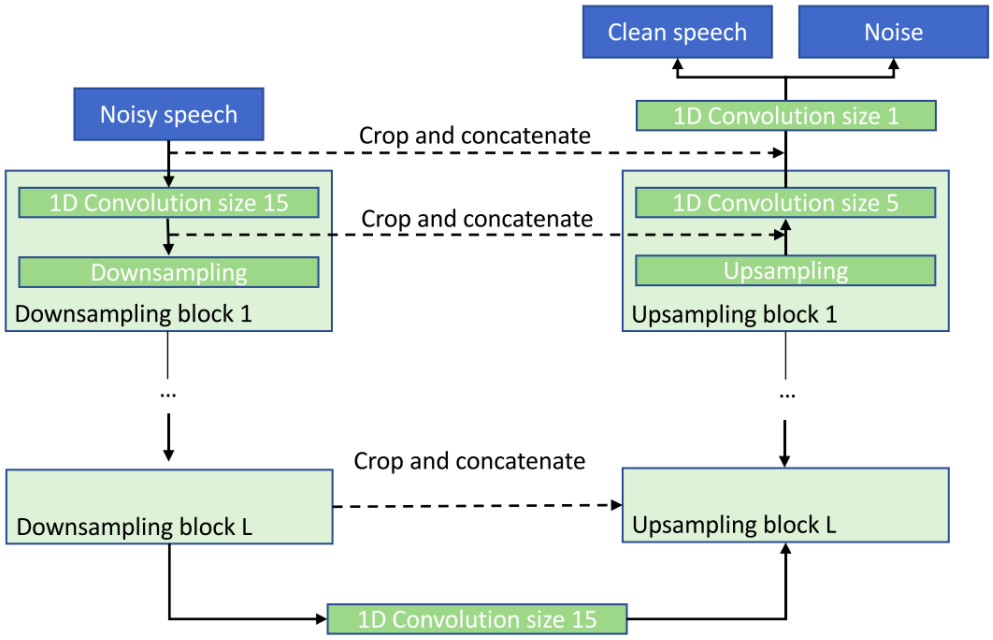

Figure 1: The Wave-U-Net architecture applied to speech enhancement.

A diagram of the Wave-U-Net architecture applied to speech enhancement is shown in Figure 1. The network has L levels in total. Each downsampling block is a 1D-convolution followed by a decimation operation that discards features for every other time step to decrease the time resolution. The first convolutional layer has F feature channels, and each such layer, up to and including the bottom layer adds another extra F number of features. Each upsampling block increases the number of samples in the time direction using a linear interpolation layer, followed by concatenation of features from the same-level downsampling block, followed by a convolution layer. Each convolutional layer in the network uses a LeakyReLU activation (except the last one, which uses tanh). For a detailed explanation of the architectural variations of the Wave-U-Net, please refer to [4, 33].

As in [27], the authors in [4] evaluate the effectiveness of the approach using the clean speech in the VCTK corpus [36] and the noises from the Demand dataset [37], together with some extra synthesized noises following the structure and scripts of Valentini-Botinhao [38]. The framework is trained using a mean squared error loss. The Wave-U-Net is compared with the classic Wiener filtering method [12], and the SEGAN framework [27]. The results presented shows that the proposed approach outperforms the baseline systems in terms of objective metrics (PESQ, CSIG, CBAK, COVL, SSNR).

2.1.2 Attention Wave-U-Net [5]

The Wave-U-Net structure has also been explored for speech enhancement by including a local self-attention mechanism that is applied to the skip connections in the architecture. Instead of directly concatenating the earlier layer features from the same scale as is usual in the U-Net architecture, the skip-connected layers are first multiplied by an attention-mask whose goal is to identify relevant features. Unlike in [4], the target output in [5] is only the foreground/clean speech, ridding the architecture of mixed responsibilities.

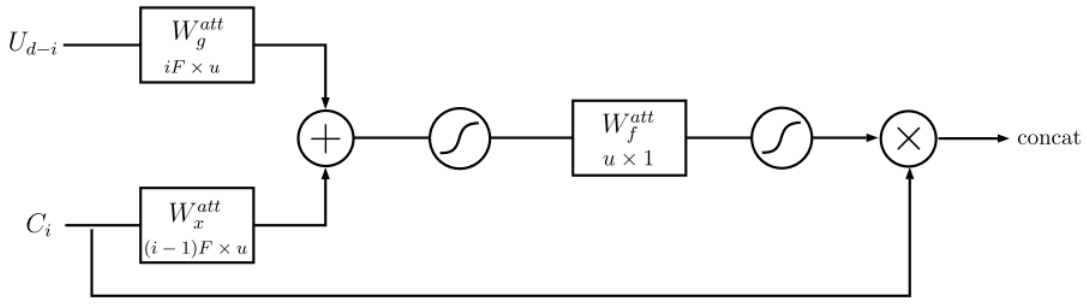

Figure 2: The attention mechanism implemented in [5] for the Wave-U-Net .

Figure 2 presents a visual description of the attention mechanism. $C_i, i = 1, ..., d$ is the output of the convolution in the ith downsampling block, and U_{d-i} is the output of the linear interpolation in the ith upsampling block. There are additional 1D convolutions, W_g^{att} and W_x^{att} (u convolutions each), of kernel size 1, which are used to compute an intermediate u-feature layer: $B_i = \sigma(W_x^{att} Ci + W_g^{att} U_{d-i} + b_{i,1})$,

which is fed to a single convolution W_f^{att} with kernel size 1 to give the attention mask: $A_i = \sigma(W_f^{att} B_i + b_{i,2})$. The term-wise product $A_i.C_i$ is then concatenated with U_{d-i}. At the final layer before the output layer, there is an attention mechanism computed the same way as done previously, but with different inputs: U_{d+1} is the output of the convolution in the last upsampling block, rather than upsampling layer, and C_0 is the noisy neural net input rather than an intermediate layer. For a more detailed explanation of the integration of the attention mechanism into the framework, please refer to [5].

Following [27], the effectiveness of the approach is evaluated using the clean speech in the VCTK corpus [38] and the noises from the Demand dataset [37]. For training, the authors consider different variants related to the loss function ($L1$, $L2$), the sampling frequency of the end signals (16 kHz, 48 kHz), and the amount of data. The 16 kHz models are compared with the classic Wiener filtering method [12], and with several deep learning models (SEGAN [27], WaveNet [29], Wave-U-Net [4], Deep Feature Loss [39]). The results presented show that the Attention Wave-U-Net structure, trained with an $L1$ loss, outperforms the baseline systems in terms of objective metrics (PESQ, CSIG, CBAK, COVL, SSNR). For a detailed appreciation of the results of the different trained variants, please refer to [5]. Interestingly, the authors show that the attention mask for the final layer is learning a soft Voice Activity Detector (VAD), which enables the network to only keep lower-level features from regions with voice activity.

2.1.3 SEGAN+ [6]

It is an improved version of the SEGAN [27] architecture. The original authors improved the framework through an extensive exploration of variations that led to an increase in performance and efficiency. It is structured as a deep convolutional auto-encoder, and its main goal is to regenerate a noisy signal into a clean version.

The auto-encoder architecture is illustrated in Figure 3. The input signal is decimated feature-wise through a number of strided convolutional layers, followed by PReLUs [40]. Decimation is implemented until a condensed representation of a few

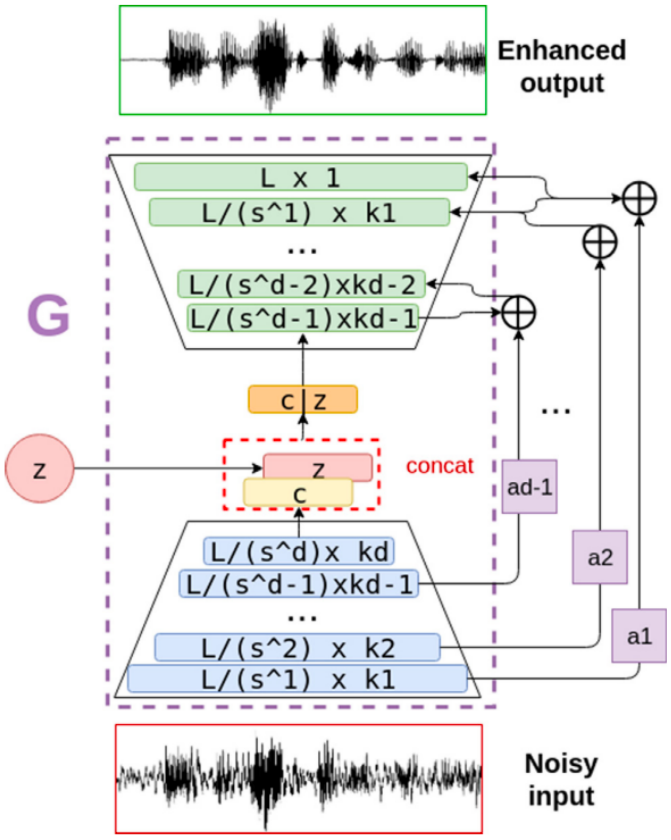

Figure 3: Auto-encoder of the SEGAN+ architecture. Feature maps are depicted in blue and green. The decimation/interpolation factor s^d depends on the stride s and layer depth index d. The input waveform length is designated L, and the number of kernels/channels at each layer is k_d. The right-side arrows denote skip connections, which have a multiplicative scalar factor a_d.

time samples is obtained, commonly called the thought vector c. This result is concatenated with the generative noise component z, which adds stochastic behavior to the generator predictions (isotropic Gaussian noise is used for z). The encoding process is reversed in the decoding stage by means of transposed convolutions, followed again by PReLUs. The only exception is the last layer, which has a tanh activation. The generator also features skip connections, which contain a multiplicative scalar factor $a_{l,k}$ per signal channel k and layer l. These $a_{l,k}$ are learned together with the whole convolutional structure, and can act as pseudo-attention mechanisms of what levels of features are more important to be shuttled in the decoding process. At the jth decoder layer input, the (scaled) lth encoder layer is concatenated with the $j-1$th decoder layer responses. The discriminator network basically follows the

same one-dimensional convolutional structure as the encoder. However, there are a few differences: (1) the discriminator network provides two input channels, (2) it uses batch normalization [41] before LeakyReLU non-linearities of $\alpha = 0.3$, and (3) in the last activation layer, there is a one-dimensional convolution layer with a single filter of width 1 and stride 1. For a detailed explanation of the model variations explored, please refer to [6].

For training and testing, the authors employ the clean speech in the VCTK corpus [36] and the noises from the Demand dataset [37], together with some extra synthesized noises following the structure and scripts of Valentini-Botinhao [38]. The framework is trained using the adversarial loss in addition to an L1 regularizer. SEGAN+ is compared with classic methods that do not require training parameters, and deep learning methods that work in the spectral domain. The authors consider an additional baseline (SEAE+) consisting of the SEGAN+ auto-enconder structure trained as a plain $L1$ regression. For a detailed explanation of these baseline methods, please refer to [6]. The results presented show that SEGAN+ outperforms the baselines in terms of the subjective metric BCK. In terms of objective metrics, it is superior to all other systems in terms of SSNR, although it has worse PESQ and MOS-like metrics than the deep learning baselines. It is notorious that the speech enhancement auto-encoder (SEAE+) is objectively comparable to SEGAN+ across perceptual metrics.

2.2 Bandwidth Extension

The problem of speech enhancement is also address by artificial bandwidth extension [42], which is also referred to as speech super-resolution in the literature. The concern is usually to extend a narrow-band signal with scarse high-frequency content, to wide-band signal. In general, the approaches to achieve this goal can be classified either as rule-based or statistical. The rule-based approaches generate the high-frequency spectrum based on the acoustic knowledge of the signal signal [43], while the statistical approaches assume that there is a non-linear relationship between the low and high-frequency components.

The statistical methods can be implemented in the frequency domain or in the time domain. One of the frequency domain methods is to predict the spectral envelope of the high-frequency component. Gaussian mixture model [44, 45, 46], hidden Markov model [47, 48, 49, 50], and neural networks [51, 52, 53, 54, 55, 56] have been used to estimate the spectral envelope. However, as indicated in [7], these methods face a common problem that the excitation, which defines the spectral fine structure of the signal, has to be estimated.

With the spread of deep learning, many approaches have been studied to directly estimate the high-frequency spectrum [57, 58, 59, 60, 9, 61]. These approaches require the phases of the high-frequency component, which is generally unknown, for signal reconstruction. However, recent deep learning techniques allow addressing the problem of bandwidth extension [62, 63, 64, 8, 65] in the time domain, thus avoiding the problem of phase estimation. [10] presents an interesting framework that uses both the time and frequency domain.

2.2.1 MfNet [7]

MfNET is a multi-scale fusion network that performs speech bandwidth extension in time domain by aggregating the speech information across different scale representations. Its original aim is to reconstruct a 16 kHz wide-band signal from a 8 kHz narrow-band signal.

The architecture is illustrated in Figure 4. C_l^r is used to represent the feature maps, where l the layer, and r indicates the time resolution or scale. In this framework, the convolution does not change the feature size, yet the downscaling is achieved by a 1D convolution of stride 2 that halves the time resolution. In this way, the neural network consists of feature maps of different scales. The multi-scale fusion block, which is used to aggregate information among the different scale representations, is composed of convolution, downscaling and upscaling operation. For upscaling, a convolution is first used to smooth the input feature, and then a bilinear interpolation is performed in the time direction by a factor of two. If the feature maps of the l-th layer are $\{C_l^1, ..., C_l^i, ..., C_l^s\}$, through the multi-scale fusion block, the feature

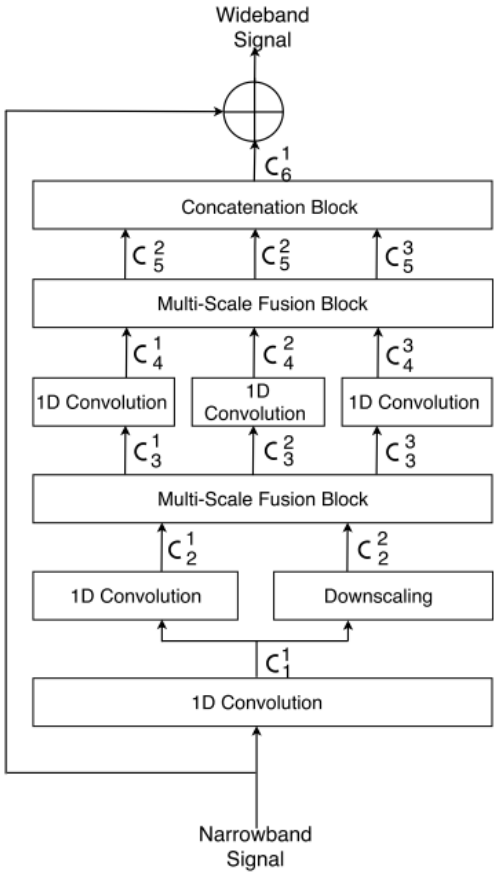

Figure 4: Schematic diagram of the MfNet model for speech bandwidth extension. $\oplus$ represents an element-wise add operation.

maps of the $(l+1)$-th layer are $\{C_{l+1}^1, ..., C_{l+1}^i, ..., C_{l+1}^s\}$, where $C_{l+1}^r = \frac{1}{S}\sum_{i=1}^{s} S(C_l^i)$. The S is one of convolution, downscaling, upscaling, and is used to resize feature maps. For a detailed explanation of the multi-scale fusion block, please refer to [7].

With MfNet, the authors explore several loss functions, namely time-domain loss, perceptually-motivated loss, adversarial loss, and a composite loss which combines perceptually-motivated loss and adversarial loss. Adopting the Valentini-Botinhao [38] corpus, the MfNet methods are compared with a simple interpolation method, a frequency domain method [57], and a time domain method [62]. The results presented show that the MfNet approaches outperform the baseline systems in terms of objective metrics (SNR, SDR, PESQ, LSD). The most notable results being those obtained with perceptually-motivated loss and/or adversarial loss. A notable advantage of MfNet is that it only requires 10% of the parameters required by the time domain baseline [62].

2.2.2 3WSS-FFTNet [8]

This architecture is a three-way split variant of the FFTNet neural vocoder structure introduced in [66]. It operates in the time domain, and its original purpose is to super-resolve speech to high-definition audio, extending the bandwidth of a signal from 8 kHz to 44.1 kHz. Although these numbers are just parameters of the framework, the authors note that by extending beyond 16 kHz, it is not simply intended to emphasize intelligibility as in traditional bandwidth extension, but rather perceptual quality and sense of presence in the recording, since the extreme upper bands offer information beyond just speech content, including the finer details of the speaker's voice.

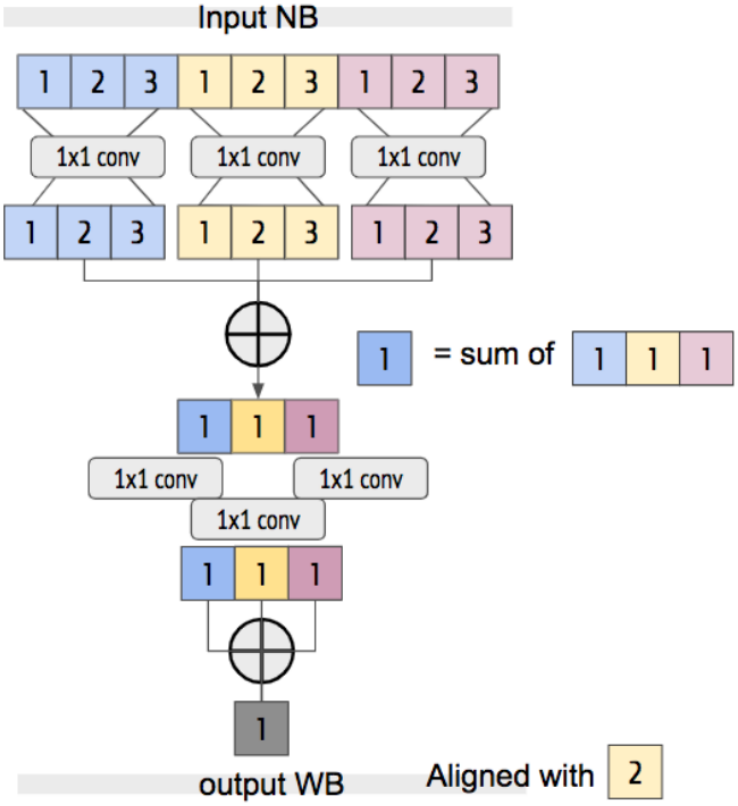

Figure 5: 3WSS-FFTNet. Starting from an input narrow-band waveform, it iteratively performs a 1×1 convolutional transformation on each split and then sum the results, producing an wide-band output waveform.

Figure 5 shows a diagram of the architecture. For an input time series with size n, it splits the input sequence into thirds of size $n/3$ each. Each third is transformed using a different 1×1 convolution and then added together: $z = W_L * x_L + W_C * x_C + W_R * x_R$,

where W_L, W_R, and W_R are the weights of the kernel applied to the left, center, and right splits, respectively. The output z is of size $n/3$ and can be further transformed with 1×1 convolution and activation before being fed into the next layer. Continuing in the same way, the next layer reduce the output size to $n/3/3$ and so on for the following layers until we have a one-sample prediction. To increase non-linearity, 1×1 convolution are replaced with Gated Linear Units [67], and one additional 1×1 convolution after summation followed by ReLU activation. The middle split x_C is added to the input of the next three-way split summation FFTNet layer to form skip connections. The final model is composed of two stacks, each of six consecutive FFTNet structures. For a more detailed explanation of the deep FFTNet architecture, please refer to [8].

Using the Device and Produced Speech (DAPS) [68] dataset, the network is trained with a loss function consisting of two parts: an $L1$ loss between the predicted waveform and the original wide-band waveform, and the $L1$ distance between the log mel-spectrograms. The 3WSS-FFTNet is compared with a frequency domain method [57], and a time domain method [62], which are the same deep learning baseline methods used in the MfNet experiments [7]. The results presented show that the proposed approach outperforms the baseline systems in terms of the subjective metric MOS. However, in terms of objective metrics, the waveform-based baseline performs best by SNR, while the spectrogram-based baseline performs best by LSD. The authors claim that their approach achieves a balance between waveform-level optimization and spectrogram-level optimization, as it consistently ranks in between both baselines by both objective metrics.

2.2.3 SSR-GAN [9]

SSR-GAN is a speech super-resolution neural network that leverages adversarial training and a regularization method for stabilizing the adversarial training. Its goal is to obtain the wide-band log-power spectra from the low-frequency log-power spectra. The authors present two variants, which use $2\times$ and $4\times$ super-resolution scales to obtain a 16 kHz signal.

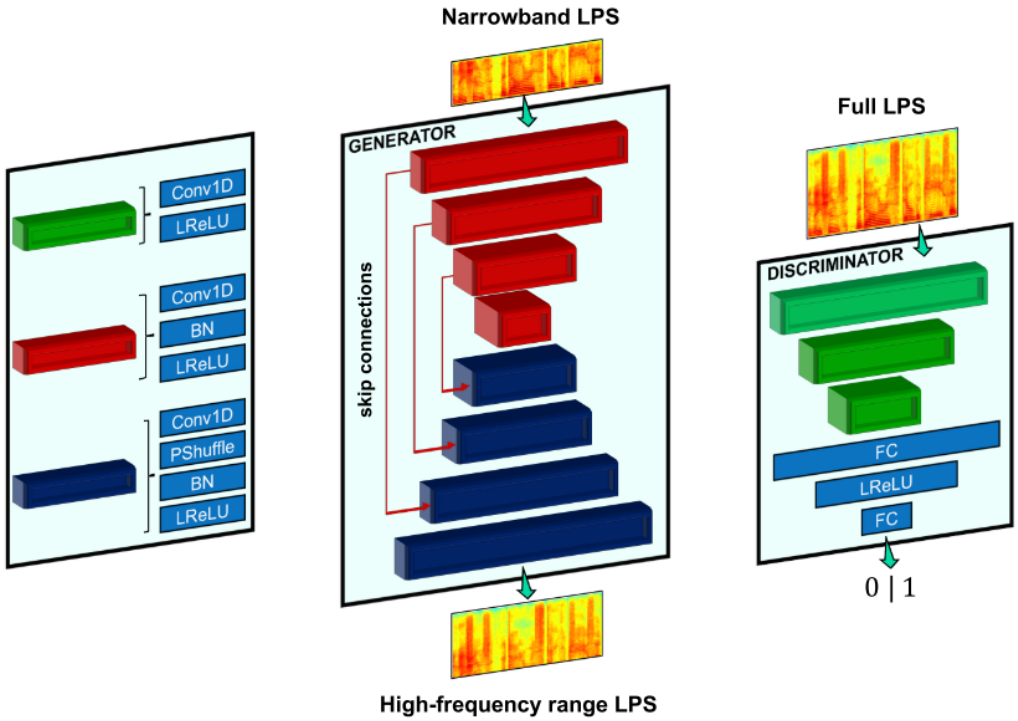

Figure 6: SSR-GAN architectures for the generator (middle) and the discriminator (right). Each rectangular block is a convolutional layer with structures color coded and detailed on the left subfigure. Notations: *BN* - batch normalization layer, *FC* - fully connected layer, *LReLU* - LeakyReLU activation, and *PShuffle* - pixel shuffle or sub-pixel layer, *LPS* - log-power spectrogram.

The architecture is illustrated in Figure 6. The generator is a sequence-to-sequence convolutional auto-encoder network that accepts log power spectrogram as input and generates the corresponding high-frequency log power spectrogram. The convolutional kernels are 1D, which operates on the time axis of the spectrogram. Batch normalization layers are used after the convolutional layers followed by LeakyReLU activations with a slope of 0.2, except for the output layer, where linear activation is used and a batch normalization layer is not used. Pixel shuffle layers, introduced in [69] for upsampling, are employed. The discriminator includes three convolutional layers that are followed by two fully connected layers. LeakyReLU activations with a slope of 0.2 are used in all layers, except for the output layer, where a linear activation function is used. It receives the concatenated narrow-band and high-frequency log power spectrograms as input. Following [57], the phase of the high-frequency range is artificially produced by flipping and repeating the narrow-band phase and

reverting the sign. For a detailed explanation of the architectural changes in the two proposed variants, please refer to [9].

The generator network is initialized by training it with only a reconstruction loss for a few epochs. Then, the framework is trained using the adversarial loss in addition to weighted reconstruction loss. The Speech Technology Research (CSTR) Voice Cloning Toolkit (VCTK) [36], and the Wall Street Journal (WSJ0) [70] corpuses are used for training and testing, respectively. The SSR-GAN variants are compared with a frequency domain method [57], and a time domain method [62], which are the same deep learning baseline methods used in the experiments of MfNet [7], and 3WSS-FFNet [8]. The results presented show that the SSR-GAN approaches outperform the baseline systems in terms of objective (SegSNR, PESQ, LSD) and subjective metrics. In addition, a notably advantage is that SSR-GAN is light-weight in terms of computational complexity and capable of running in real-time on edge devices.

2.2.4 TFNet [10]

TFNet is a deep convolutional neural network that uses both the time and frequency domain for the task of super audio resolution. Its original aim is to reconstruct a 16 kHz high-resolution signal from a 4 kHz low-resolution signal. The authors come up with this framework arguing that regression from low-resolution to high-resolution in time domain or frequency domain solves a different problem. In the time domain, it is analogous to the image super-resolution task, mapping "audio patches" from low-resolution to high-resolution, and in the frequency domain it is analogous to the semantic image inpainting task, outputting the high-frequency components from the low-frequency components of a spectrogram.

Figure 7 illustrates the architecture. It consists of a fully convolutional encoder-decoder based network $H(x;\Theta)$. For a given low-resolution input x, H predicts the high-resolution audio reconstruction, $\hat{z}$, and the high-resolution spectral magnitude $\hat{m}$. The spectral fusion layer combines the $\hat{z}$ and $\hat{m}$: $M = w \odot \mid \mathscr{F}(\hat{z}) \mid + (1-w) \odot \hat{m}$, and finally the output is reconstructed: $\hat{y} = \mathscr{F}^{-}(Me^{j\angle \mathscr{F}(\hat{z})})$, where $\mathscr{F}$ denotes the

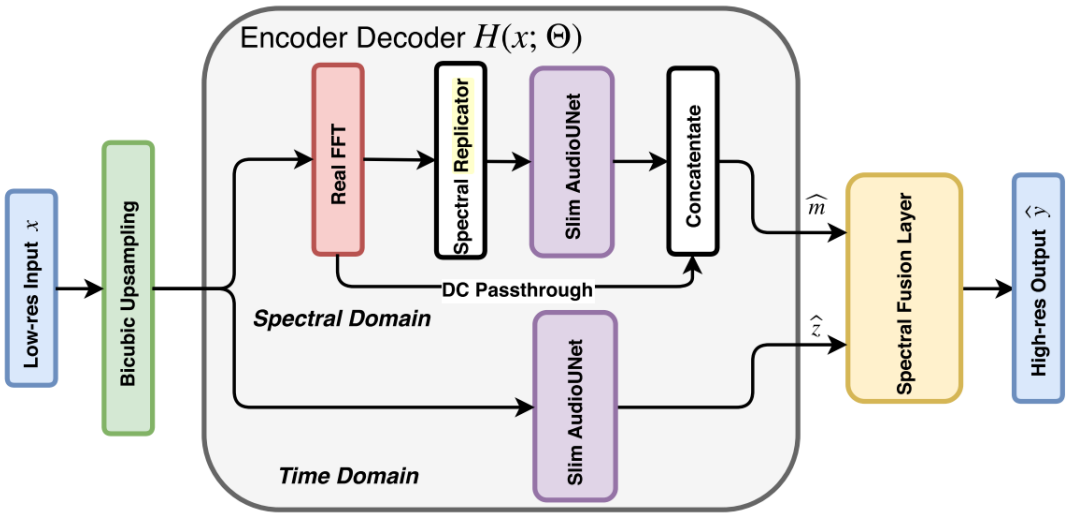

Figure 7: Overall pipeline of Time-Frequency Network. TFNet contains a branch which explicitly models the reconstruction's spectral magnitude, while the other branch models the reconstruction in time domain. The output of the two branches are finally combined with our Spectral fusion layer to synthesize the high resolution output.

Fourier transform, $\odot$ is a element-wise multiplication and w is a trainable parameter. The high frequency components of the input are zeros before the spectral replicator layer, which replaces the zero values with copies of the low frequency components. For more detailed explanations of the replication layer and the spectral fusion layer, please refer to [10].

Adopting the VCTK dataset [36], and the Piano dataset[1], the framework is trained using the $L2$ reconstruction loss with weight decay. The TFNet is compared with a frequency domain method [57], and a time domain method [62], which are the same deep learning baseline methods used in the experiments of MfNet [7], 3WSS-FFNet [8], and SSR-GAN [9]. The results presented show that the FFTNet outperforms the baseline systems in terms of objective metrics (SNR and LSD). Also, the authors conduct an ablation study whose results support the worthwhile contribution of both branches of the framework.

[1]Publicy available at `https://archive.org/`

Chapter 3

Datasets

3.1 VCTK [38]

This popular dataset has been used in almost all approaches considered in Chapter 2. It is publicly available on the DataShare repository of the University of Edinburgh[1]. The clean data are recordings of sentences uttered by 30 English speakers (15 male and 15 female), 28 for training and 2 reserved for testing. The noisy data has been generated by mixing clean data with various types of noise.

For the training set, 40 different noise conditions are considered: 10 types of noise (2 artificial and 8 from the DEMAND database) with 4 signal-to-noise ratios (SNRs) each (15, 10, 5 and 0 dB). In total, this brings about 11572 samples (around 9 hours and 20 minutes) with approximately 10 different sentences in each condition per training speaker.

As for the test set, the speakers, the noise types and the SNRs are all different from those of the training. This set consists of 20 different noise conditions are considered: 5 types of noise (all from the DEMAND database) with 4 SNRs each (17.5, 12.54, 7.5 and 2.5 dB). In total, this brings about 824 samples (around 34 minutes) with approximately 20 different sentences in each condition per test speaker.

[1] `https://datashare.is.ed.ac.uk/handle/10283/1942`

3.2 DAPS [68]

This dataset was originally designed for the purpose of investigating the transformation of low-quality speech recordings into recordings that sound as if they were produced in a professional recording studio. Nonetheless, it has also been used in the research of more conventional speech enhancement targets such as denoising, dereverberation, decoloration, and bandwidth extension. It is publicly available on the Internet Archive[2].

The dataset consists of 20 speakers (10 female and 10 male) reading 5 excerpts each from public domain books (which provides about 14 minutes of data per speaker). The clean recordings were done in a professional recording studio. For the noisy versions, the clean recordings were played through a high quality coaxial loudspeaker in real-world acoustic environments (two offices, two conference rooms, a bedroom, a living room, and a balcony) and recorded on devices consumer (an iPad Air and an iPhone 5S), yielding a total of 12 versions. For a more detailed description of the creation of the DAPS dataset, please refer to [68].

[2] `https://archive.org/details/daps_dataset`

Chapter 4

Experiments

4.1 VCTK + DAPS

A combination of the VCTK and DAPS datasets has been used for the experiments. From the DataShare repository, the VCTK dataset is separated into training set (11572 samples) and test set (824 samples). All of their samples are used in the experiments. 100 samples from the training set were randomly selected to be part of the validation set. The samples in this dataset are intended for noise suppression from sources other than those of interest.

Regarding the DAPS dataset, the recordings of 18 speakers with two scripts each were used for training. These recordings were segmented into 5 second excerpts to balance the number of samples with those in the VCTK dataset. In this way, 11320 training samples were generated. The recordings of the script number 5 uttered by the remaining 2 speakers were reserved for testing (20 samples). Again, 100 training samples were randomly selected to be part of the validation set. In all subsets, the samples recorded on a balcony near a road with heavy traffic were omitted because they are too harsh compared to the other settings. In addition to suppressing noise from sources other than that of interest, this dataset also targets the dereverberation of acoustic environments and the decoloration of capture devices.

To account for the bandwidth extension target, all input samples are first down-

sampled to 8 kHz and then resampled to 44.1 kHz. In total, 22692 samples for training (11472 VCTK + 11220 DAPS), 200 samples for validation (100 VCTK + 100 DAPS) and 844 samples for evaluation (824 VCTK + 20 DAPS) are used in the experiments.

4.2 Baseline

The structure of this baseline is shown in Figure 8. It was inspired by the frameworks reviewed in Chapter 2, especially those in sections 2.1.1 and 2.1.2 which are the ones that explore the Wave-U-Net for speech enhancement. As in [5], the output target is only the foreground/clean speech, ridding the architecture of mixed responsibilities. Following [33], the desired number of audio samples in the output waveform is 16384, the network has 12 layers, the first convolution uses 24 filters, the filters of the downsampling blocks are of size 15 and those of the upsampling blocks are of size 5. This baseline is trained using an L1 loss on the raw waveform.

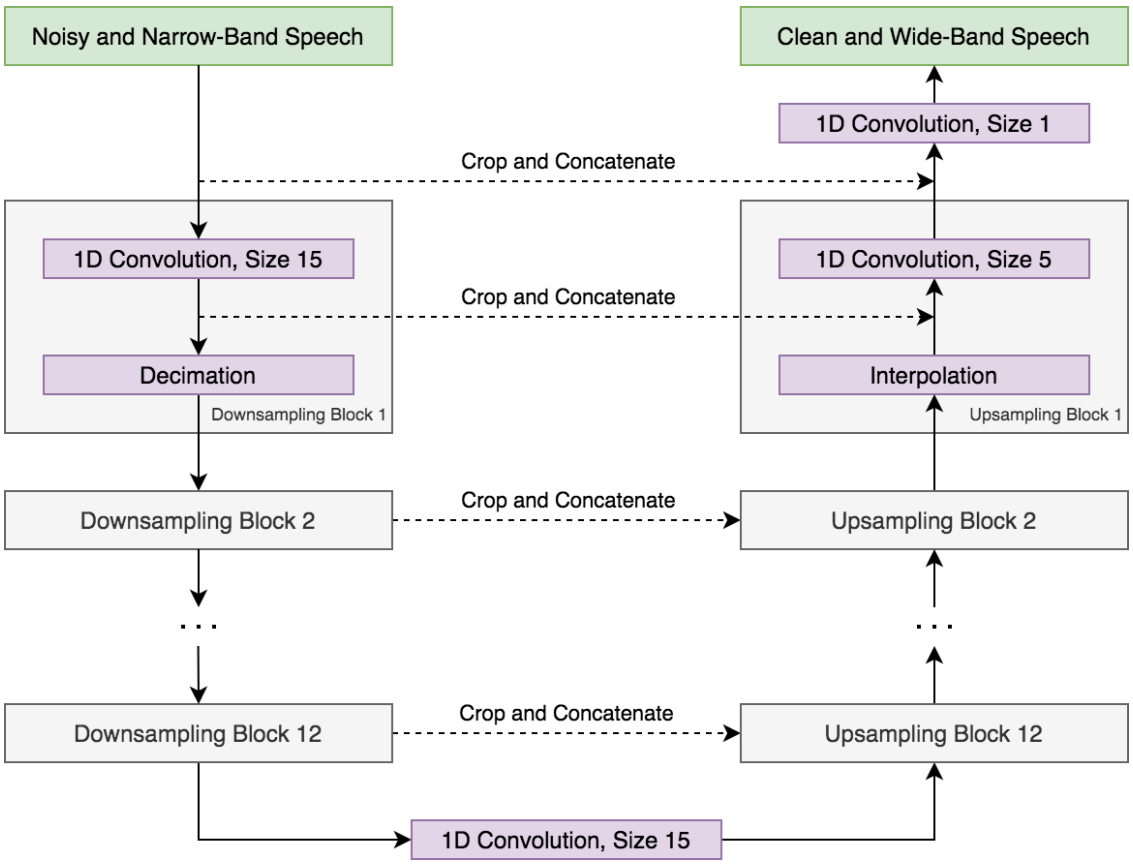

Figure 8: Baseline

4.3 Training Setup

During training, audio excerpts are sampled randomly and inputs padded with input context. The Adam optimization algorithm is used with learning rate 10^{-4}, decay rates $\beta_1 = 0.9$ and $\beta_2 = 0.999$, and a batch size of 32. Early stopping is applied if there is no improvement on the validation set for 20 epochs, with one epoch defined as 2000 iterations. After that, the last model is fine-tuned with the batch size doubled and the learning rate lowered to 10^{-5}, again until 20 epochs without improvement in validation loss. Finally, the model with the best validation loss is selected.

4.4 Variations

4.4.1 Spectral Losses

In [71], the introduction of a spectral training objective has shown a significant improvement in the performance of the Wave-U-Net architecture in a task other than speech enhancement. The above inspired here the training of the network using an $STFT$ loss, a Mel loss, and simple combinations of these with an $L1$ loss on the raw waveform.

The STFT spectrograms have been computed using a window size of 4096 samples and a hop size of 1024 samples. Such a window size provides an appropriate resolution for the low frequencies of male subjects. The $STFT$ loss is defined as the $L1$ distance between the magnitudes of the $STFT$ spectrograms of the predicted signal and the clean reference.

The insight behind the use of Mel spectrograms is that their scale is linked to human perception. Here, the Mel spectrograms are derived from the $STFT$ spectrograms using triangular filters on the full spectrum of human hearing (20 Hz to 22050 kHz). The Mel loss is defined as the $L1$ distancia between the Mel spectrogram of the predicted signal and the clean reference.

4.4.2 Noise Input Vector

It consists of concatenating the input data with a noise vector just before the first convolution in the network. This is done with the intention of stimulating the generation of the high-frequency content that the input samples completely lack. The noise is sampled from a normal distribution. In addition to an $L1$ loss on the raw waveform, this variation is also trained using all the losses considered in Section 4.4.1.

4.4.3 Attention Mechanism

In [5], the authors claim that the inclusion of an attention mechanism significantly improves the performance of the Wave-U-Net in terms of objective speech quality metrics. This has motivated experimentation here with the same attention mechanism, which is described in detail in Section 2.1.2. In addition to an $L1$ loss on the raw waveform, this variation is also trained using a composite objective (i.e. $L1 + STFT + Mel$).

4.4.4 SEGAN+ Variations

Three of the most important variations that led to an increase in the performance of the SEGAN architecture in [6] are simultaneously implemented here. These are: (1) the introduction of lernable skip connections using a scalar factor per hidden feature that allows for importance filtering of feature maps from encoder to decoder; (2) the selection of feature maps prior to the non-linear activation in the encoder to shuttle them up; and (3) the concatenation of a noise vector with the thought vector in the network bottleneck. In addition to an $L1$ loss on the raw waveform, these variations are also trained using a composite objective (i.e. $L1 + STFT + Mel$).

Chapter 5

Results

5.1 Evaluation Metrics

The performance of the models in generating enhanced speech is evaluated with well-known objective metrics. Each measurement used compares the enhanced signal to the clean reference of the VTCK + DAPS test set files. The metrics, their meaning and their range of values are as follows:

- PESQ: Perceptual Evaluation of Speech Quality. The implementation used is a python wrapper for the PESQ score calculation C routine [1]. It ranges from -0.5 to 4.5. The higher the score, the better.
- LSD: Log-Spectral Distance. It takes values between 0 and ∞. The lower the value, the better. Mathematically it is defined as below:

$$LSD = \frac{1}{L}\sum_{l=1}^{L}\sqrt{\frac{1}{K}\sum_{k=1}^{K}(Y(l,k)-\hat{Y}(l,k))^2} \tag{5.1}$$

  where Y and $\hat{Y}$ are the log-spectral power magnitudes of the ground truth and the estimate, respectively. The K represents the number of frequencies in a frame and the L is the number of frames of a speech utterance.

[1] `https://github.com/vBaiCai/python-pesq`

- SSNR: Segmental Signal to Noise Ratio. This has been computed with the implementation used in [6][2], which is a python re-implementation of the algorithm in [72]. It is in the range $[-10, 35]$. The higher the value, the better.

5.2 Model Comparison

Tables 2 and 3 show the results of the metrics evaluated in all the experiments conducted. The first thing that becomes clear is that these results differ significantly between the test sets. For instance, PESQ values are always higher in the VCTK test set, and LSD values are always lower in the DAPS test set. This is not entirely surprising given the essential difference in the procedures used in each dataset to create their versions.

ID	Variant	Loss	PESQ	LSD	SSNR
-	Noisy and Narrow-Band	-	3.023	3.387	1.697
00	Baseline	$L1$	3.336	1.940	9.410
01	Spectral Losses	$L1 + STFT$	3.473	2.729	9.072
02		$L1 + Mel$	3.475	2.711	9.093
03		$STFT$	3.461	2.741	-4.743
04		Mel	3.456	2.732	-7.193
05		$STFT + Mel$	3.475	2.734	-4.766
06		$L1 + STFT + Mel$	3.476	2.717	8.998
07	Noise Input Vector	$L1$	3.348	**1.939**	9.214
08		$L1 + STFT$	3.464	2.756	9.149
09		$L1 + Mel$	3.483	2.726	9.085
10		$STFT$	3.467	2.705	3.311
11		Mel	**3.499**	2.738	0.899
12		$STFT + Mel$	3.467	2.676	3.771
13		$L1 + STFT + Mel$	3.472	2.752	8.955
14	Attention Mechanism	$L1$	3.313	1.960	9.351
15		$L1 + STFT + Mel$	3.461	2.731	9.122
16	SEGAN+ Variations	$L1$	3.346	1.946	**9.436**
17		$L1 + STFT + Mel$	3.472	2.724	9.167

Table 2: VCTK Results. Best performances are shown in bold.

[2] https://github.com/santi-pdp/segan_pytorch/blob/master/segan/utils.py

ID	Variant	Loss	PESQ	LSD	SSNR
-	Noisy and Narrow-Band	-	2.461	3.057	-5.027
00	Baseline	$L1$	2.000	1.851	**-0.541**
01	Spectral Losses	$L1 + STFT$	3.178	1.071	-1.575
02		$L1 + Mel$	3.167	1.053	-1.831
03		$STFT$	3.234	1.043	-2.747
04		Mel	**3.252**	1.046	-5.667
05		$STFT + Mel$	3.251	1.058	-2.613
06		$L1 + STFT + Mel$	3.211	1.043	-2.084
07	Noise Input Vector	$L1$	2.118	1.751	-0.599
08		$L1 + STFT$	2.865	1.067	-1.693
09		$L1 + Mel$	2.910	1.047	-1.708
10		$STFT$	3.015	1.033	-2.593
11		Mel	2.970	1.041	-4.842
12		$STFT + Mel$	3.126	1.060	-2.490
13		$L1 + STFT + Mel$	2.992	**1.014**	-1.852
14	Attention Mechanism	$L1$	1.990	1.879	-0.637
15		$L1 + STFT + Mel$	3.217	1.058	-2.054
16	SEGAN+ Variations	$L1$	2.062	1.884	-0.652
17		$L1 + STFT + Mel$	2.927	1.072	-1.949

Table 3: DAPS Results. Best performances are shown in bold.

It can be seen that the use of spectral supervision in the optimization objective has improved the PESQ score, especially in the DAPS dataset. The models that include an $L1$ loss in the raw waveform encourage optimization of the SSNR metric, which in fact can be seen in the results, especially for the VCTK dataset.

Intuitively, it would be expected that the models that have used spectral components in the training objective would perform better in terms of the LSD metric. Although the results on the DAPS dataset support this reasoning, the effect on the VCTK dataset has been the opposite.

The concatenation of the input sample with a noise vector has a modest effect on the objective metrics. However, at the listening level, the impact is much more noticeable. A number of listening examples are available online[3]. Figure 9 shows how this variant generates more coherent high-frequency content.

[3] `https://jdavibedoya.github.io/SE_Wave-U-Net/`

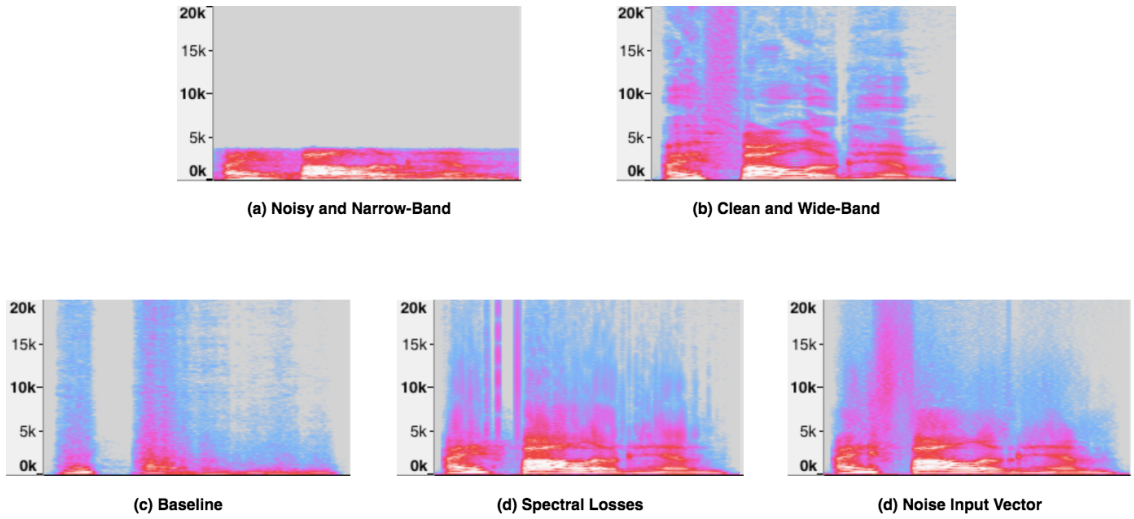

Figure 9: Comparing spectrograms. (a) Noisy and Narrow-Band input. (b) Clean and Wide-Band ground truth. (c) Baseline. (d) Spectral Loss ($STFT + Mel$). (e) Noise Input Vector and Spectral Loss ($STFT + Mel$).

The experiments carried out with the model variants inspired by the attention mechanism and the SEGAN+ architecture, have yielded results similar to those obtained with the original Wave-U-Net architecture using the same losses. Also, these have made the models larger and converge slower.

Chapter 6

Conclusions

This thesis has studied the performance of the Wave-U-Net deep learning network to simultaneously solve the conventional targets of speech enhancement, i.e. denoising, dereverberation, decoloration and bandwidth extension. Among the explored model variations, spectral losses are those that have had the most favorable impact on the results, both in terms of objective metrics and at the listening level. Also, concatenating the input audio with a noise vector in the network has led to the generation of a more coherent high-frequency content in the output signal.

Although the models that used spectral components in the training objective were able to improve the quality of the input samples for both data distributions, the results were particularly better for data with artificially mixed noise (VCTK) compared to those obtained for data recorded in real environments (DAPS). The foregoing prompts reflection on the advisability of using artificially generated low-quality data for training models with these targets.

For future work, better loss functions should be investigated, such as spectral optimization objectives that consider the phase information in the complex $STFT$ [73], and those provided by generative adversarial networks [6, 7, 9]. Another next step in this research is to implement stronger variations of the Wave-U-Net auto-encoder such as the inclusion of an LSTM layer in the bottleneck, similar to that of the DEMUCS architecture [74].

Chapter 7

Reproducibility

All the code needed to reproduce the experiments is openly available on GitHub: `https://github.com/jdavibedoya/SE_Wave-U-Net`. This code is based on the original Wave-U-Net implementation for audio source separation [1].

The repository description includes direct links to pre-trained weights of the explored model variants and to the datasets used.

[1] `https://github.com/f90/Wave-U-Net`

Bibliography

[1] Benesty, J., Makino, S. & Chen, J. *Speech Enhancement* (Springer, 2005).

[2] Vincent, E., Virtanen, T. & Gannot, S. *Audio Source Separation and Speech Enhancement*, chap. 1 (John Wiley & Sons, 2018).

[3] Williamson, D. S. & Wang, D. Time-Frequency Masking in the Complex Domain for Speech Dereverberation and Denoising. *IEEE/ACM Transactions on Audio, Speech, and Language Processing* (2017).

[4] Macartney, C. & Weyde, T. Improved Speech Enhancement with the Wave-U-Net (2018). `1811.11307`.

[5] Giri, R., Isik, U. & Krishnaswamy, A. Attention Wave-U-Net for Speech Enhancement. In *WASPAA* (2019).

[6] Pascual, S., Serrà, J. & Bonafonte, A. Time-Domain Speech Enhancement Using Generative Adversarial Networks. *Speech Communication* (2019).

[7] Hao, X. *et al.* Time-Domain Neural Network Approach for Speech Bandwidth Extension. In *ICASSP* (2020).

[8] Feng, B., Jin, Z., Su, J. & Finkelstein, A. Learning Bandwidth Expansion Using Perceptually-Motivated Loss. In *ICASSP* (2019).

[9] Eskimez, S. E. & Koishida, K. Speech Super Resolution Generative Adversarial Network. In *ICASSP* (2019).

[10] Lim, T. Y., Yeh, R. A., Xu, Y., Do, M. N. & Hasegawa-Johnson, M. Time-Frequency Networks for Audio Super-Resolution. In *ICASSP* (2018).

[11] Berouti, M., Schwartz, R. & Makhoul, J. Enhancement of speech corrupted by acoustic noise. In *ICASSP* (1979).

[12] Jae Lim & Oppenheim, A. All-Pole Modeling of Degraded Speech. *IEEE Transactions on Acoustics, Speech, and Signal Processing* (1978).

[13] Ephraim, Y. Statistical-Model-Based Speech Enhancement Systems. *Proceedings of the IEEE* (1992).

[14] Dendrinos, M., Bakamidis, S. & Carayannis, G. Speech Enhancement from Noise: A Regenerative Approach. *Speech Communication* (1992).

[15] Ephraim, Y. & Van Trees, H. L. A Signal Subspace Approach for Speech Enhancement. *IEEE Transactions on Speech and Audio Processing* (1995).

[16] Lu, X., Tsao, Y., Matsuda, S. & Hori, C. Speech Enhancement Based on Deep Denoising Autoencoder. In *INTERSPEECH* (2013).

[17] Xu, Y., Du, J., Dai, L. & Lee, C. A Regression Approach to Speech Enhancement Based on Deep Neural Networks. *IEEE/ACM Transactions on Audio, Speech, and Language Processing* (2015).

[18] Narayanan, A. & Wang, D. Ideal Ratio Mask Estimation Using Deep Neural Networks for Robust Speech Recognition. In *ICASSP* (2013).

[19] Wang, Y., Narayanan, A. & Wang, D. On Training Targets for Supervised Speech Separation. *IEEE/ACM Transactions on Audio, Speech, and Language Processing* (2014).

[20] Maas, A. *et al.* Recurrent Neural Networks for Noise Reduction in Robust ASR. In *INTERSPEECH* (2012).

[21] Weninger, F., Hershey, J. R., Le Roux, J. & Schuller, B. Discriminatively Trained Recurrent Neural Networks for Single-Channel Speech Separation. In *GlobalSIP* (2014).

[22] Weninger, F. *et al.* Speech Enhancement with LSTM Recurrent Neural Networks and its Application to Noise-Robust ASR. *LVA/ICA* (2015).

[23] Erdogan, H., Hershey, J. R., Watanabe, S. & Le Roux, J. Phase-Sensitive and Recognition-Boosted Speech Separation Using Deep Recurrent Neural Networks. In *ICASSP* (2015).

[24] Xia, B. & Bao, C. Speech Enhancement with Weighted Denoising Auto-Encoder. In *INTERSPEECH* (2013).

[25] Shivakumar, P. G. & Georgiou, P. Perception Optimized Deep Denoising AutoEncoders for Speech Enhancement. In *INTERSPEECH* (2016).

[26] Fu, S., Wang, T., Tsao, Y., Lu, X. & Kawai, H. 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* (2018).

[27] Pascual, S., Bonafonte, A. & Serrà, J. SEGAN: Speech Enhancement Generative Adversarial Network (2017).

[28] Park, S. R. & Lee, J. A Fully Convolutional Neural Network for Speech Enhancement (2016).

[29] Rethage, D., Pons, J. & Serra, X. A wavenet for speech denoising. In *ICASSP* (2018).

[30] Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V. & Courville, A. C. Improved Training of Wasserstein GANs. In *Advances in Neural Information Processing Systems 30* (Curran Associates, Inc., 2017).

[31] Qin, S. & Jiang, T. Improved Wasserstein Conditional Generative Adversarial Network Speech Enhancement. *EURASIP Journal on Wireless Communications and Networking* (2018).

[32] Higuchi, T., Kinoshita, K., Delcroix, M. & Nakatani, T. Adversarial Training for Data-Driven Speech Enhancement Without Parallel Corpus. In *ASRU* (2017).

[33] Stoller, D., Ewert, S. & Dixon, S. Wave-U-Net: A Multi-Scale Neural Network for End-to-End Audio Source Separation (2018). 1806.03185.

[34] Jansson, A. *et al.* Singing Voice Separation with Deep U-Net Convolutional Networks (2017).

[35] Ronneberger, O., Fischer, P. & Brox, T. U-Net: Convolutional Networks for Biomedical Image Segmentation (2015). 1505.04597.

[36] Veaux, C., Yarnagishi, J. & MacDonald, K. CSTR VCTK Corpus: English Multi-speaker Corpus for CSTR Voice Cloning Toolkit. *The Centre for Speech Technology Research (CSTR)* (2016).

[37] Thiemann, J., Ito, N. & Vincent, E. DEMAND: A Collection of Multi-Channel Recordings of Acoustic Noise in Diverse Environments (2013).

[38] Valentini-Botinhao, C. Noisy Speech Database for Training Speech Enhancement Algorithms and TTS Models (2016). University of Edinburgh. School of Informatics. Centre for Speech Technology Research (CSTR).

[39] Germain, F. G., Chen, Q. & Koltun, V. Speech Denoising with Deep Feature Losses (2018). 1806.10522.

[40] He, K., Zhang, X., Ren, S. & Sun, J. Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification (2015). 1502.01852.

[41] Ioffe, S. & Szegedy, C. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift (2015). 1502.03167.

[42] Larsen, E. & Aarts, R. *Audio Bandwidth Extension: Application of Psychoacoustics, Signal Processing and Loudspeaker Design* (Wiley, 2005).

[43] Dietz, M., Liljeryd, L., Kjorling, K. & Kunz, O. Spectral Band Replication, a Novel Approach in Audio Coding. In *Audio Engineering Society Convention* (2002).

[44] Kun-Youl Park & Hyung Soon Kim. Narrowband to Wideband Conversion of Speech Using GMM Based Transformation. In *ICASSP* (2000).

[45] Chennoukh, S., Gerrits, A., Miet, G. & Sluijter, R. Speech Enhancement Via Frequency Bandwidth Extension Using Line Spectral Frequencies. In *ICASSP* (2001).

[46] Seo, H., Kang, H. & Soong, F. A Maximum a Posterior-Based Reconstruction Approach to Speech Bandwidth Expansion in Noise. In *ICASSP* (2014).

[47] Jax, P. & Vary, P. Artificial Bandwidth Extension of Speech Signals Using MMSE Estimation Based on a Hidden Markov Model. In *ICASSP* (2003).

[48] Chen, G. & Parsa, V. HMM-Based Frequency Bandwidth Extension for Speech Enhancement Using Line Spectral Frequencies. In *ICASSP* (2004).

[49] Bauer, P. & Fingscheidt, T. An HMM-Based Artificial Bandwidth Extension Evaluated by Cross-Language Training and Test. In *ICASSP* (2008).

[50] Song, G.-B. & Martynovich, P. A Study of HMM-Based Bandwidth Extension of Speech Signals. *Signal Processing* (2009).

[51] Kontio, J., Laaksonen, L. & Alku, P. Neural Network-Based Artificial Bandwidth Expansion of Speech. *IEEE Transactions on Audio, Speech, and Language Processing* (2007).

[52] Wang, Y., Zhao, S., Liu, W., Li, M. & Kuang, J. Speech Bandwidth Expansion Based on Deep Neural Networks. In *INTERSPEECH* (2015).

[53] Abel, J. & Fingscheidt, T. Artificial Speech Bandwidth Extension Using Deep Neural Networks for Wideband Spectral Envelope Estimation. *IEEE/ACM Transactions on Audio, Speech, and Language Processing* (2018).

[54] Schmidt, K. & Edler, B. Blind Bandwidth Extension Based on Convolutional and Recurrent Deep Neural Networks. In *ICASSP* (2018).

[55] Li, S., Villette, S., Ramadas, P. & Sinder, D. J. Speech Bandwidth Extension Using Generative Adversarial Networks. In *ICASSP* (2018).

[56] Sautter, J., Faubel, F., Buck, M. & Schmidt, G. Artificial Bandwidth Extension Using a Conditional Generative Adversarial Network with Discriminative Training. In *ICASSP* (2019).

[57] Li, K. & Lee, C. A Deep Neural Network Approach to Speech Bandwidth Expansion. In *ICASSP* (2015).

[58] Liu, B., Tao, J., Wen, Z., Li, Y. & Bukhari, D. A Novel Method of Artificial Bandwidth Extension Using Deep Architecture. In *INTERSPEECH* (2015).

[59] Gu, Y., Ling, Z.-H. & Dai, L.-R. Speech Bandwidth Extension Using Bottleneck Features and Deep Recurrent Neural Networks. In *INTERSPEECH* (2016).

[60] Abel, J., Strake, M. & Fingscheidt, T. A Simple Cepstral Domain DNN Approach to Artificial Speech Bandwidth Extension. In *ICASSP* (2018).

[61] Bachhav, P., Todisco, M. & Evans, N. Latent Representation Learning for Artificial Bandwidth Extension Using a Conditional Variational Auto-Encoder. In *ICASSP* (2019).

[62] Kuleshov, V., Enam, S. Z. & Ermon, S. Audio Super Resolution using Neural Networks (2017). `1708.00853`.

[63] Gu, Y. & Ling, Z.-H. Waveform Modeling Using Stacked Dilated Convolutional Neural Networks for Speech Bandwidth Extension. In *INTERSPEECH* (2017).

[64] Wang, M. *et al.* Speech Super-Resolution Using Parallel WaveNet. In *ISCSLP* (2018).

[65] Ling, Z., Ai, Y., Gu, Y. & Dai, L. Waveform Modeling and Generation Using Hierarchical Recurrent Neural Networks for Speech Bandwidth Extension. *IEEE/ACM Transactions on Audio, Speech, and Language Processing* (2018).

[66] Jin, Z., Finkelstein, A., Mysore, G. J. & Lu, J. FFTNet: A Real-Time Speaker-Dependent Neural Vocoder. In *ICASSP* (2018).

[67] Dauphin, Y. N., Fan, A., Auli, M. & Grangier, D. Language Modeling with Gated Convolutional Networks (2016). `1612.08083`.

[68] Mysore, G. J. 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* (2015).

[69] Shi, W. *et al.* Real-Time Single Image and Video Super-Resolution Using an Efficient Sub-Pixel Convolutional Neural Network (2016). `1609.05158`.

[70] Garofalo, J., Graff, D. P. & Pallett, D. CSR-I (WSJ0) Complete Linguistic Data Consortium. *Philadelphia, USA: LDC* (2007).

[71] Ramires, A., Chandna, P., Favory, X., Gómez, E. & Serra, X. Neural Percussive Synthesis Parameterised by High-Level Timbral Features. In *ICASSP* (2020).

[72] P.C., L. *Speech Enhancement: Theory and Practice* (CRC Press, 2013).

[73] Takaki, S., Nakashika, T., Wang, X. & Yamagishi, J. STFT Spectral Loss for Training a Neural Speech Waveform Model (2018). `1810.11945`.

[74] Defossez, A., Synnaeve, G. & Adi, Y. Real Time Speech Enhancement in the Waveform Domain (2020). `2006.12847`.