YAŞAR UNIVERSITY

GRADUATE SCHOOL

MASTER THESIS

DETECTOR-DRIVEN SPEECH BACKGROUND NOISE REMOVAL WITH CONVOLUTIONAL NETWORKS

CEM AYAR

THESIS ADVISOR: ASSIST. PROF. (PHD) ARMAN SAVRAN

COMPUTER ENGINEERING

PRESENTATION DATE: 16.08.2022

BORNOVA / İZMİR
AUGUST 2022

We certify that, as the jury, we have read this thesis and that, in our opinion, it is fully adequate, in scope and quality, as a thesis for the degree of Master of Science.

Jury Members: **Signature:**

Assist. Prof. (PhD) Arman SAVRAN
Yaşar University ....

Assist. Prof. (PhD) Umut AVCI
Yaşar University ....

Assist. Prof. (PhD) Nesli ERDOĞMUŞ
Izmir Institute of Technology ....

Prof. (PhD) Yücel Öztürkoğlu
Director of the Graduate School

ABSTRACT

DETECTOR-DRIVEN SPEECH BACKGROUND NOISE REMOVAL WITH CONVOLUTIONAL NETWORKS

Ayar, Cem
MSc, Computer Engineering
Advisor: Assist. Prof. (PhD) Arman SAVRAN
August 2022

Speech background noise is a common issue, which has become especially important with the increasing popularity of online meetings and live internet broadcasting. Recently, Deep Neural Networks (DNNs) have shown to be highly successful in the suppression of a wide variety of background noise types without requiring more than one microphone. However, such deep models which consume substantial resources cause many real-life applications to become expensive, burdensome or sometimes impractical. This thesis proposes a solution to mitigate the problem by de-activating a high performance DNN when there is no significant noise, that is, by a detector-driven noise removal approach. First, we optimized a modern time-domain convolutional neural network (CNN), known as Conv-TasNet, regarding the efficiency and performance. Then, a CNN-based noisy-speech detector was designed and evaluated with different size and resolution variations for the detector-driven scheme. We found that the optimal detector has only a 2% computation load of the optimal Conv-TasNet, with a very low noisy-speech miss-rate causing only negligible performance drop. Thus, having successful noisy-speech detection with this minor computation overhead, we justified our detector-driven approach for possible substantial gains in efficiency. This efficiency gain is inversely proportional to noise occurrence probability. Besides, we have also shown that, by automatic identification of already clean-speech, slight degradations due to occasional processing artifacts can be avoided.

Keywords: Speech background noise removal, speech enhancement, Conv-TasNet, noisy-speech detector, noise canceling, CNN

ÖZ

SAPTAYICI-GÜDÜMLÜ KONUŞMA ARKA PLANI GÜRÜLTÜSÜNÜN EVRİŞİMSEL AĞLAR İLE GİDERİLMESİ

Ayar, Cem
Yüksek Lisans Tezi, Bilgisayar Mühendisliği
Danışman: Dr. Öğr. Üyesi. Arman SAVRAN
Ağustos 2022

Konuşma arka planı gürültüsü, çevrimiçi toplantıların ve canlı internet yayınlarının artan popülaritesi ile özelikle önem teşkil eden, yaygın bir sorundur. Son zamanlarda, Derin Sinir Ağlarının (DSA), geniş bir yelpazedeki arka plan gürültü çeşitlerinin bastırılmasında, birden fazla mikrofon gerektirmeden yüksek başarı elde ettiği gösterilmiştir. Ancak, ciddi kaynak tüketen böyle derin ağlar birçok gerçek hayat uygulamasının pahalı, külfetli veya bazen kullanışsız olmasına yol açar. Bu tez, problemi hafifletmek için, yüksek başarımlı bir DSA'yı, kayda değer gürültü olmayan zamanlarda devre dışı bırakan, yani saptayıcı-güdümlü bir gürültü giderme yaklaşımı ile, bir çözüm önermektedir. İlk olarak, Conv-TasNet olarak bilinen zaman alanında çalışan modern bir evrişimsel sinir ağı (ESA), verimlilik ve başarımına göre eniyilenmiştir. Sonra, ESA-temelli bir gürültülü konuşma saptayıcı tasarlanmış ve farklı büyüklük ve çözünürlük varyasyonları ile saptayıcı-güdümlü tasarı için değerlendirilmiştir. Optimum saptayıcının, optimum Conv-TasNet'in hesaplama yükünün sadece %2'sine sahip olduğu ve çok düşük gürültülü konuşma ıskalama oranı ile sadece ihmal edilebilir bir başarım düşüşüne neden olduğu bulunmuştur. Böylece, bu önemsiz hesaplama yükü ile başarılı bir şekilde gürültülü konuşma saptayarak, saptayıcı-güdümlü yaklaşımımızın muhtemel önemli verimlilik kazanımları için kullanılabileceğini doğruladık. Bu verimlilik kazanımı gürültü oluşma olasılığı ile ters orantılıdır. Bunun yanında, zaten temiz olan konuşmanın otomatik olarak tanımlanmasıyla, ara sıra oluşan işleme kusurlarının yol açtığı hafif bozulmalardan sakınılabileceğini de gösterdik.

Anahtar Kelimeler: Konuşma arka planı gürültüsünün giderilmesi, konuşma iyileştirme, Conv-TasNet, gürültülü konuşma saptayıcı, gürültü engelleme, CNN

ACKNOWLEDGEMENTS

First of all, I would like to thank my supervisor Assist. Prof. (Ph.D.) Arman Savran for his guidance, friendly chats, unwavering support, and patience that made this work possible during this study.

Also, I would like to express my enduring love to my parents, who are always supportive, loving, and caring to me in every possible way in my life.

Finally, from the bottom of my heart, I would like to thank my darling Rabia Deniz Sandıkçı. She always supported me and gave me morals when I felt depressed.

Cem Ayar
İzmir, 2022

TEXT OF OATH

I declare and honestly confirm that my study, titled "DETECTOR-DRIVEN SPEECH BACKGROUND NOISE REMOVAL WITH CONVOLUTIONAL NETWORKS" and presented as a Master's Thesis, has been written without applying to any assistance inconsistent with scientific ethics and traditions. I declare, to the best of my knowledge and belief, that all content and ideas drawn directly or indirectly from external sources are indicated in the text and listed in the list of references.

Cem Ayar
16.08.2022

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

SYMBOLS AND ABBREVIATIONS

ABBREVIATIONS:

DNN	Deep Neural Network
MMSE	Minimum Mean Square Error
LMS	Least Mean Squares
RLS	Recursive Least Squares
STFT	Short-Time Fourier Transform
ISTFT	Inverse Short-Time Fourier Transform
CNN	Convolutional Neural Networks
FCNN	Fully Convolutional Neural Networks
RNN	Recurrent Neural Networks
GAN	Generative Adversarial Networks
DAE	Deep Autoencoder
L-MMSE	Logarithmic Minimum Mean Square Error
Conv-TasNet	Fully - Convolutional Time-domain Audio Separation Network
TasNet	Time-domain Audio Separation Network
SNR	Signal-to-Noise Ratio
SDR	Signal-to-Distortion Ratio
SI-SNR	Scale Invariant Signal to Noise Ratio
LSTM	Long Short-Term Memory
VAD	Voice Activity Detection
TCN	Temporal Convolutional Network
VBD	VoiceBank Demand Dataset
TP	True Positive
TN	True Negative
FP	False Positive

FN	False Negative
TPR	True Positive Rate
TNR	True Negative Rate
FPR	False Positive Rate
FNR	False Negative Rate
ROC	Receiver Operation Characteristic
AUC	Area Under Curve
PPR	Positive Prediction Rate
DNS	Deep Noise Suppression

SYMBOLS:

Hz	Hertz
kHz	Kilohertz
dB	Decibel

CHAPTER 1
INTRODUCTION

In the last decade, online gatherings and live internet broadcasting have become quite common. People often join online business meetings, lectures, or seminars, talk with family and friends, or attend social events, remotely from their mobile phones or personal computers. Especially the current pandemic conditions have enormously contributed to the popularity of such remote social connections and even seem to be more common in the future. However, background noise can severely degrade or hinder the experience; for example, when one or more participants join from crowded environments like a café and airport, or from home with a working vacuum cleaner or chirping birds behind. Speech background noise is also an issue for other applications, like live broadcasting, phones, hearing aids, or recordings. With the wide variety of speech background noise types, background noise cancellation has been an active research topic for decades.

Although great progress has been made in background noise removal, it is not commonly available in devices, or if available, usually with mild suppression capabilities or requires more than one audio acquisition channel. Currently, high performance single-channel noise suppression is possible only with deep neural network models; however, with the drawback of quite demanding computation and energy consumption. They usually require powerful local GPUs (*NVIDIA*, n.d.) or cloud services (*Noise Cancelling App & Echo Reduction Software | Krisp*, n.d.). This thesis aims at a high performance single-channel DNN solution that works on a wide variety of speech background noise scenarios, however, by reducing the computations depending on the noise to gain substantial efficiency.

1.1 Problem Statement

Commonly, noise removal methods run all the time after being switched on. However, in the presence of background noise that occasionally occurs, resources would be wasted due to a complex DNN running for no-noise intervals. For instance, imagine while you are in an online meeting, there is some construction work going on outside, and it makes you turn on the noise removal. But after the construction is suspended during your meeting, the noise removal will continue to run and consume computation resources and battery power if you forget to turn it off. Now imagine the presence of background noise is automatically detected with very lightweight computation. Then the expensive suppression module could be avoided when there is no significant noise. This example demonstrates the main motivation of the thesis.

We conjecture that background noise can be detected cheaply, and it can conveniently drive a high-performance noise suppression module. In other words, when noise is detected, suppression is activated; otherwise, it is stopped. This way, depending on the availability of background noise, we can reduce the related computations substantially. Therefore, this thesis aims to develop a detector-driven method for efficient noise cancellation. For that purpose, we limit our design to convolutional neural networks (CNNs) since state-of-the-art CNNs (Luo & Mesgarani, 2019; Pandey & Wang, 2019a; Park & Lee, 2017) can achieve high noise suppression performance; however, more efficiently compared to alternatives, like recurrent neural networks (RNNs) (Sun et al., 2017; Weninger et al., 2015), and generative adversarial networks (GANs) (Fu et al., 2019, 2021; Pascual et al., 2017).

1.2 Contributions

In this thesis, we show that a lightweight CNN can detect speech background noise with a satisfactory performance and thus improve efficiency considerably. To the best of our knowledge, there is no such study in the literature. First, we find an optimal set of hyper-parameters regarding the performance and complexity using the convolutional time-domain audio separation network (Conv-TasNet) (Luo & Mesgarani, 2019). Then we design a simple CNN noisy-speech detector and evaluate it for different sets of architecture configurations over noisy and clean-speech clips.

We find that an optimal Conv-TasNet can achieve 19.9 dB SI-SNR (scale-invariant signal-to-noise ratio) on noisy clips with 670 MFLOPS (mega floating-point operations per a single forward pass). On the other hand, our most efficient noisy-speech detector requires only 7.3 MFLOPS overhead. By successfully disabling when there is no background noise, it reduces the computations to about 350 MFLOPS on average over a balanced set of clean and noisy-speech clips; on noisy and clean clips, the MFLOPS are about 660 and 32, respectively. At first glance, it seems like a twofold increase in efficiency. However, gain in efficiency is inversely proportional to background noise occurrence probability. Thus, depending on the scene, detector-driven removal can provide dramatic economy in practice by preventing unnecessary processing in the absence of noise.

Second, as a by-product, we found out that Conv-TasNet slightly degrades the quality when applied to clean-speech. Therefore, our detector also provides some small improvement on the quality by not allowing Conv-TasNet to process already clean-speech.

1.3 Thesis Outline

The rest of the thesis is organized as follows. In Chapter 2, we review the related literature, discussing the alternative models used for background noise removal. Thus, based on the literature, we justify the chosen DNN model that constitutes the basis of our design. Chapter 3 elaborates on speech background noise, stressing some typical noise types. We describe the proposed detector network and the detector-driven suppression in detail in Chapter 4. Chapter 5 is devoted to experimental evaluations with a discussion of the results. Finally, in Chapter 6, conclusions and future work are given.

CHAPTER 2
RELATED WORK

In this chapter, first, we survey related speech background noise removal methods. Then we briefly review voice activity detection methods as it is related to noisy-speech versus clean-speech classification.

2.1 Speech Background Noise Removal

The audio noise-cancellation problem has been studied for decades, as it is a common issue with telephones, radios, hearing aids, etc. In this section, we review the approaches in the literature that aim to clean audio signals from noise in general, focusing on speech background noise removal methods.

Figure 2.1 shows the taxonomy of speech enhancement(SE) methods. The methods are categorized into four main classes: conventional, adaptive filtering, machine learning, and multimodal.

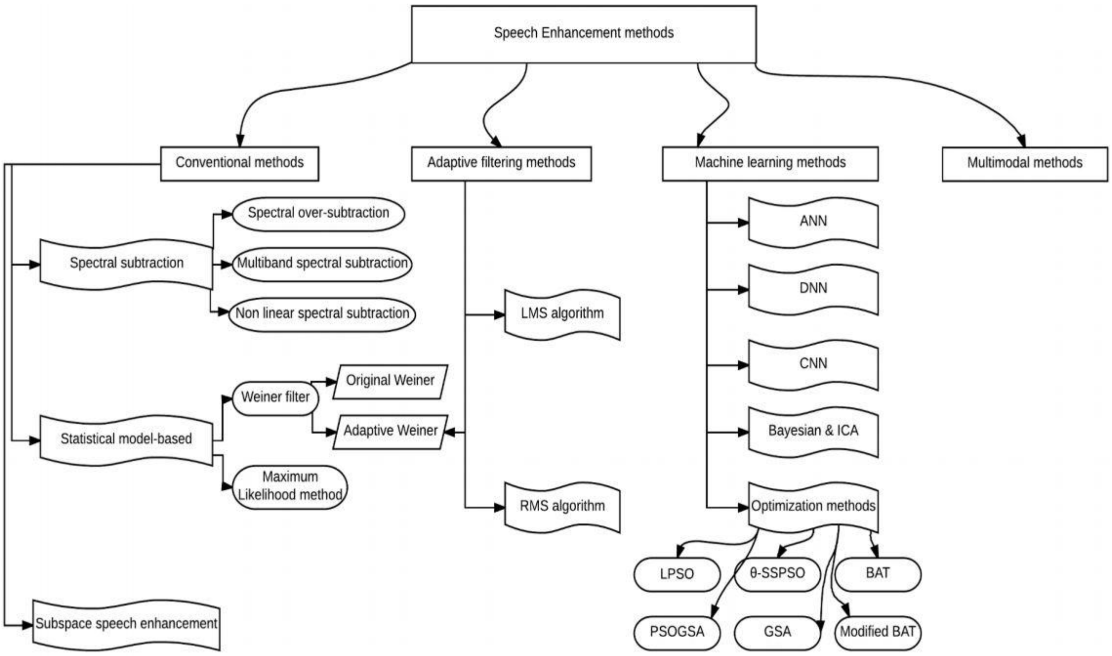


Figure 2.1. Speech Enhancement Taxonomy, up to 2018 (M. et al., 2018)

In the conventional category, spectral subtraction methods (Boll, 1979) are one of the oldest methods. They are realized by estimating and suppressing the noise frequencies. Spectral subtraction can be carried out by non-linear, multiband, and over-subtraction methods (Colored, 2002). In general, they are mainly effective at stationary noise but not in nonstationary cases, though a number of improved variants have also been proposed (Hu & Loizou, 2002; Udrea & Ciochina, 2003; L.-P. Yang & Fu, 2005). Other approaches in the conventional category are statistical models and subspace methods. Statistical models are closely related to the adaptive filtering methods, which have been proposed to overcome the ineffectiveness of the conventional methods at-non-stationary noise. A prominent statistical model is Wiener filtering (Wiener et al., 1949), which models the noise as additive white Gaussian noise and works in the spectral domain by the Short Time Fourier Transform (STFT).

The adaptive filtering methods do not need prior information about the noise types, unlike the conventional approaches, and are good at dealing with nonstationary noise. For instance, an adaptive version of Wiener filtering, which works in the time domain, has been proposed (El-Fattah et al., 2008). The major methods in the adaptive filtering category are Least Mean Squares (LMS) (Stearns, 1985) and Recursive Least Squares (Engel et al., 2004). LMS aims at reducing the difference between noisy and clean sounds by mean square error minimization, and it has been used widely. On the other hand, the later RLS method reduces a weighted squared error; however, it is computationally more complex than LMS. A comparative evaluation of LMS and RLS methods is available in (Rakesh & Kumar, 2015). One drawback of these methods is that they need at least two microphones for high performance, which is not applicable in this thesis as we aim at solving the problem only by one microphone, i.e., on single-channel audio.

Progress in machine learning has led to superior solutions. Bayesian methods, optimization-based methods, artificial neural networks (shallow neural networks), and, more recently, DNNs are the major approaches in this category. More specifically, DNNs, such as Convolutional Neural Networks (CNNs) (lu et al., 2013; Park & Lee, 2017), Recurrent Neural Networks (RNNs) (Sun et al., 2017; Weninger et al., 2015), Generative Adversarial Networks (GANs) (Fu et al., 2019, 2021; Pascual et al., 2017), Transformer (Koizumi et al., 2021; Subakan et al., 2021, 2022) models have surpassed other machine learning methods. As an early DNN work, a CNN model in the form of a deep autoencoder (lu et al., 2013) is applied to learn both speech and noise characteristics after training by clean and noisy-speech clip pairs. It was successful at the suppression of complex noise characteristics where traditional methods failed. This early work has also shown that increasing the depth of the network and training set size improve the quality, and training with clean-noisy pairs yields better performance than training with individual clips. Some other notable early noise-cancellation DNNs, include the use of an Ideal Ratio Mask (Hummersone et al., 2014) and a regression model (Xu et al., 2014, 2015) have shown by both objective and subjective evaluations that their regression DNN obtained much better performance than shallow neural networks and statistical model-based methods.

More recent studies have frequently proposed CNN-based methods due to their high performances as well as high computation efficiencies. For instance, Park & Lee (2017) has proposed a lightweight CNN to suppress babble noise suitable for embedded systems, e.g., to be used in hearing aids. Their model is called Redundant Convolutional Encoder-Decoder (R-CED). By involving an initial STFT stage, it is 68 times smaller than a competing fully connected network and 12 times smaller than a competing RNN (Park & Lee, 2017).

In general, until recent years, RNNs had been claimed to be considerably better performing than CNNs. A prominent example is TasNet (Luo & Mesgarani, 2018) which has been developed for speaker separation but solves the background noise removal in the special case of one speaker. At the same time, some researchers (Braun et al., 2021; Hu et al., 2020; Pandey & Wang, 2019b; Zhao et al., 2018) have proposed combining CNN and RNN by using the CNN as a feature extractor and the RNN for modeling inter-frame relations. Yet it has also been shown that a sophisticated design that applies RNN independently at sub-bands can attain real-time performance (Hao et al., 2021). Recently, the time domain RNN-based TasNet model (Luo & Mesgarani, 2018) was transformed into a convolutional network, named as Conv-TasNet (Luo & Mesgarani, 2019), which uses a learnable encoder/decoder instead of STFT/ISTFT by replacing the RNN separation structure by stacked dilated temporal convolutions, which has been inspired by speech denoising wavenet (Rethage et al., 2018). Moreover, they employ depthwise separable convolutions to lower the computational load substantially. As a result of these transformations, real-time operation with low latency became possible due to small frame lengths. After its success, different authors have made more investigations and improvements to Conv-TasNet (Heitkaemper et al., 2020; Koizumi et al., 2021; Koyama et al., 2020; Sonning et al., 2020). Since this thesis focuses on efficiency, we also develop our method using the highly successful Conv-TasNet model.

Note that there are other high-performant noise removal networks as well, based on GANs (Fu et al., 2019, 2021; Pascual et al., 2017) or attention/transformers (Koizumi et al., 2021; Subakan et al., 2021, 2022). Though these recent approaches may offer better performances, the complexities are much higher. For example, in Subakan's (2021 and 2022) model sizes are about ten times larger. Therefore, we have not considered those models in this thesis.

2.2 Voice Activity Detection

We have not found a prior study directly on the noisy-speech detector as we develop in this thesis. However, there is substantial prior research on closely related topics like voice activity detection (VAD) and environmental noise classification. The history of VAD is rather long. Initial VAD algorithms (Greenwood & Kinghorn, 1999; Tanyer & Ozer, 2000; Tucker, 1992) are computationally very simple as they have been developed based on simple measures like short time energy, spectral flatness, periodicity, and zero crossing rate. Nevertheless, these methods suffer from low SNRs and nonstationary background noise. Later methods aimed at overcoming these limitations by statistical techniques (Almajai & Milner, 2008; J.-H. Chang et al., 2006; Ramirez et al., 2005; Ramírez et al., 2004; Sohn et al., 1999). Although these methods were successful to some extent in finding practical applications, they usually fail with the presence of complex background noise. State-of-the-art is based on CNNs (S.-Y. Chang et al., 2018; Sehgal & Kehtarnavaz, 2018) with superior performance on complex background noise. Also, a rather different approach to voice activation is by the guidance of environmental noise classification (Hwang et al., 2015; Ting et al., 2021) for improved performance under a wide range of noise types, which again are based on DNNs.

CHAPTER 3
SPEECH BACKGROUND NOISE

In this chapter, we briefly explain the main noise characteristics depending on noise classes which have commonly been treated in the speech background noise removal literature.

In general, sound is characterized by means of its loudness and frequency. Loudness is the power measured in Decibels (dB). Human speech is typically in the range of 50-60 dB. High dB levels can be disturbing, often intolerable, and can even cause health problems. For example, a chain saw produces 110 dB sound, and a motorcycle has about 80-90 dB levels. These types of sounds are unwanted when people are speaking and thus accepted as noise. Figure 3.1 shows some common noise sources with their typical dB ranges.

The number of vibrations in a second is called frequency measured in the units of hertz (Hz). Humans can only hear between 20 Hz and 20 kHz. Sounds of frequencies higher than 20 kHz are called as ultrasound, and sounds of frequencies less than 20 Hz are called as infrasound. Audio signals are often visualized with their frequency spectrums in addition to their waveforms since we perceptually distinguish sound waves based on the frequency as there is often some unique pattern due to the power concentrated on certain frequencies. It is calculated by STFT and has three dimensions: the horizontal axis is time, the vertical axis is frequency, and color tones represent power variations. In Figure 3.2, we see how different a clean-speech signal can be compared to its noisy counterparts by means of waveform and frequency spectrum visualizations, where power increases from blue to red tones. We observe that with decreasing signal-to-noise ratio (SNR), the patterns of the original clean-speech signal become less recognizable. SNR is a measure that compares the power of the signal against background noise by a simple ratio:

$$SNR = \frac{\boldsymbol{P}signal}{\boldsymbol{P}noise}. \qquad (1)$$

Due to its very wide dynamic range, SNR is commonly expressed in the logarithmic decibel scale:

$$S\mathrm{NR}_{\mathrm{dB}} = 10\log_{10}(\mathrm{SNR}). \qquad (2)$$

We can divide noise classes into four basic categories as stationary background noise, unwanted human speech, intermittent noise, and impulsive (burst) noise.

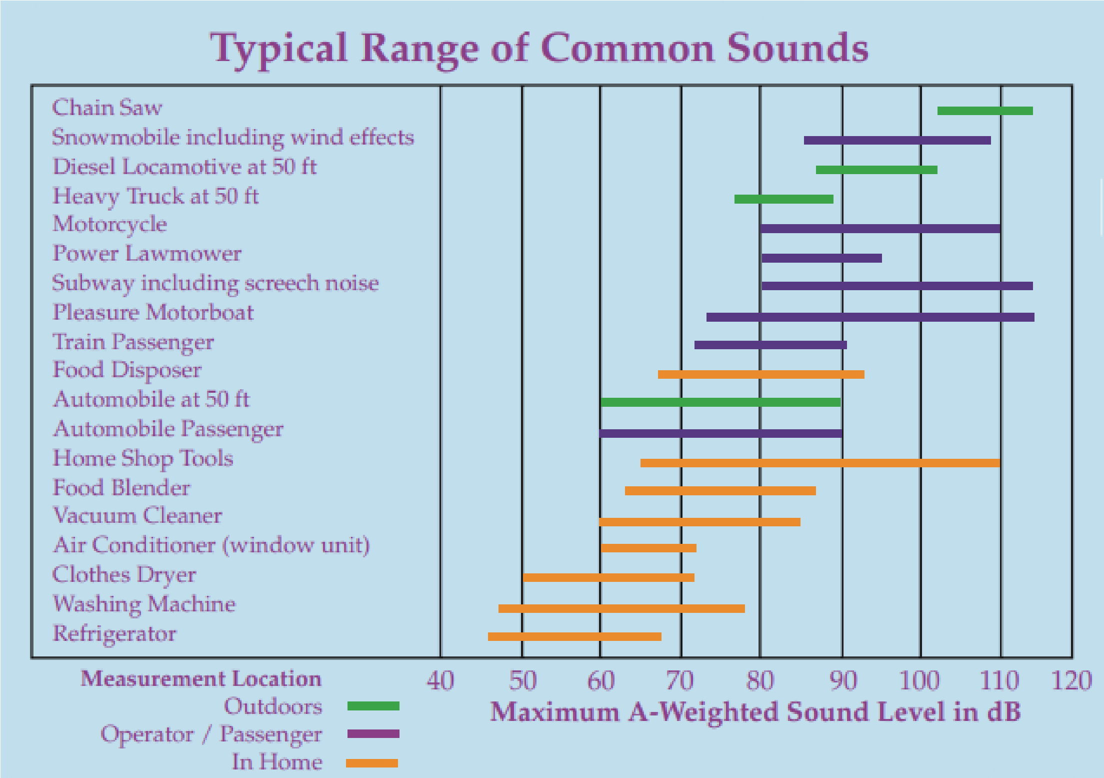


Figure 3.1. Typical Range of Common Sounds (Suter, 1991)

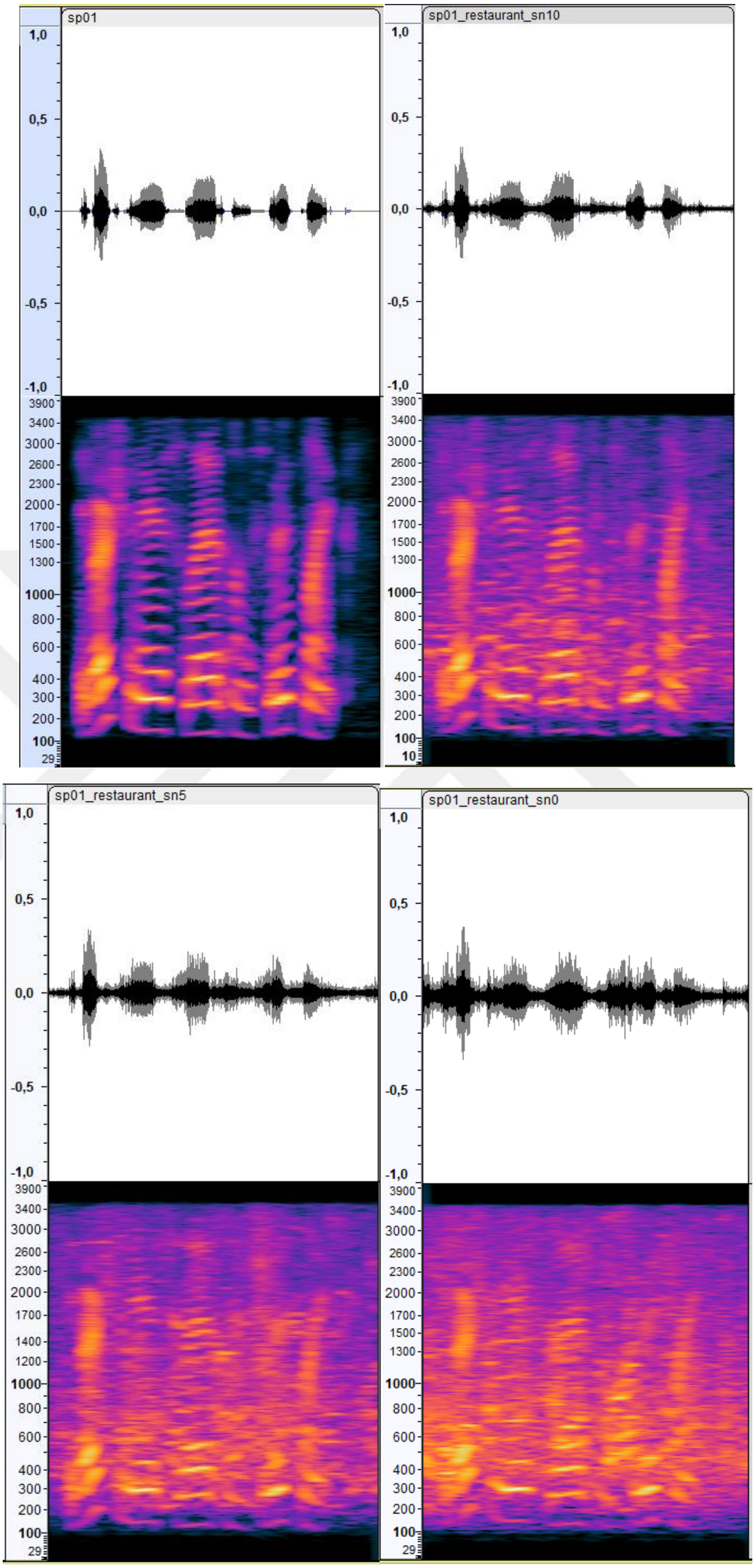


Figure 3.2 Clean-speech (top-left), restaurant background noise SNR 10-5-0 (left-to-right and top-to-bottom) (Pearce & Hirsch, 2000)

3.1 Stationary Noise

With stationary noise, the power level of noise is assumed constant over time. Thus, simple statistics like mean and variance can easily estimate noise characteristics. Therefore, stationary noise had been the focus of the early work, as reviewed in Chapter 2. Examples are white noise, babble noise, and machinery noise in Figure 3.2. However, nonstationary noise is encountered more often, like sounds of cars passing on the street, barking dogs, and chirping birds (see Figure 3.3).

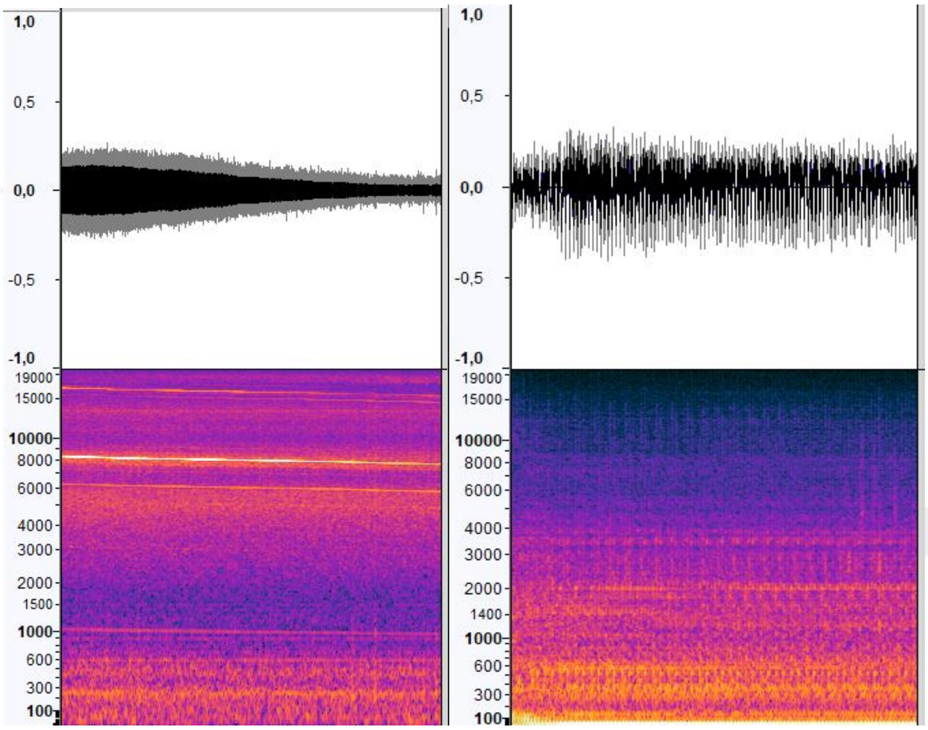


Figure 3.2 Drilling a metal (left), engine idle (right) (Pearce & Hirsch, 2000)

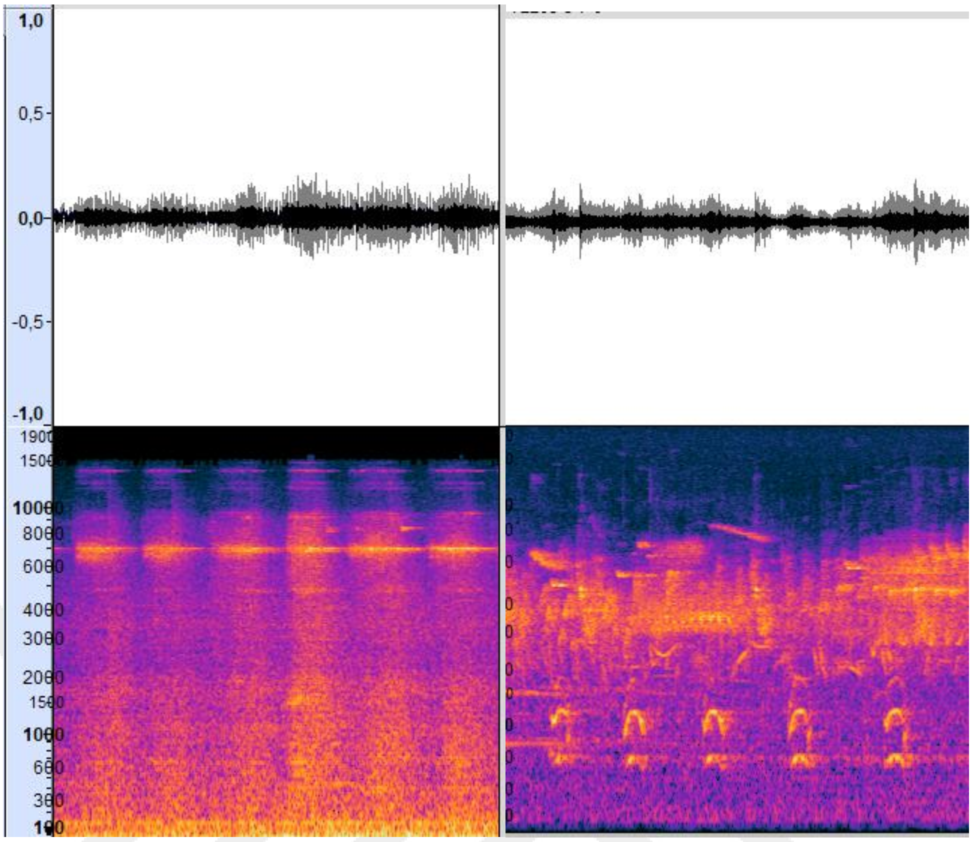


Figure 3.3 Cutting an iron rod (left), and mixed dog-bird noise (right) (Pearce & Hirsch, 2000)

3.2 Unwanted Human Speech

Unwanted human speech is the most challenging noise type for removal due to the obvious similarity between the signal and the noise characteristics. If the people speaking in the environment are far from the microphone, like in an open area, the problem is usually rather manageable. Otherwise, speaker separation models are needed. In Figure 3.4, the environmental sound of children playing in the park is visualized.

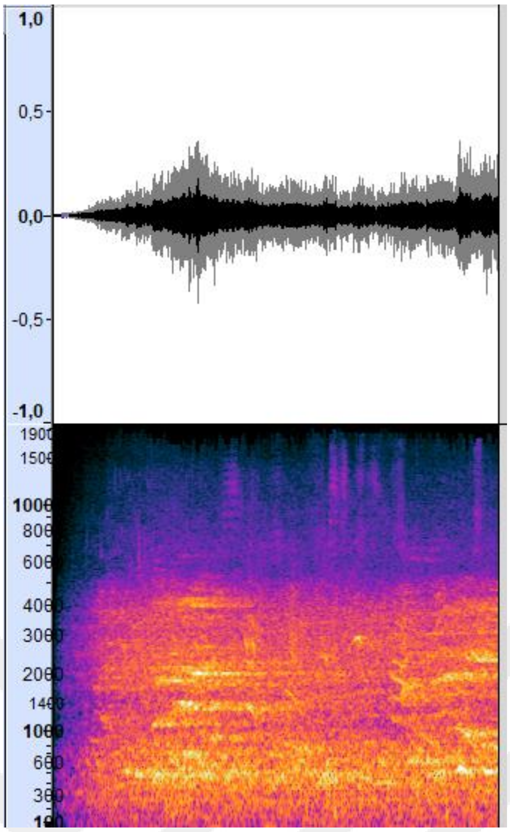


Figure 3.4 Children sound (Pearce & Hirsch, 2000)

3.3 Intermittent Noise

Intermittent noise increases and decreases in a short period. For example, an ambulance passing by quickly, airplanes passing above the house, etc. Figure 3.5, visualizations of ambulance siren sound, and the same with people talking.

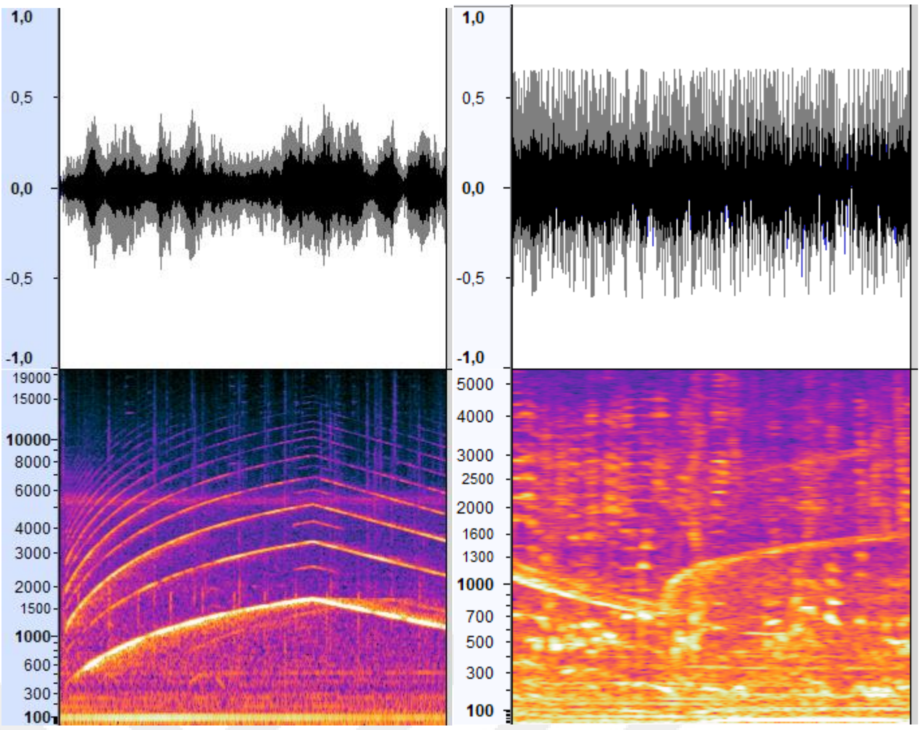


Figure 3.5 Ambulance siren (left), ambulance siren with people talking (right) (Pearce & Hirsch, 2000)

3.4 Impulsive Noise (Burst Noise)

Impulsive noise rises suddenly, stays for a few seconds, or occurs several times. For example, hammer noise, gunshot, glass hitting on marble ground, and car horns. Figure 3.6 shows visualizations of car signal sound and gunshot sound.

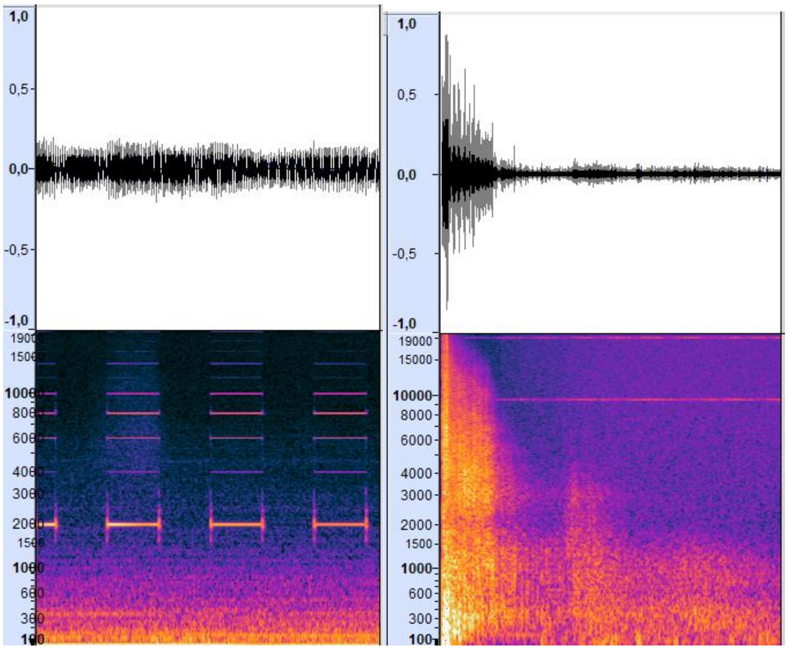


Figure 3.6 Car signal sound (left) and gunshot (right) (Pearce & Hirsch, 2000)

CHAPTER 4
DETECTOR-DRIVEN BACKGROUND NOISE REMOVAL

We aim at improved efficiency by activating a noise removal network only when there is significant noise. For this purpose, we build a noisy-speech detector to drive a background noise suppression network. If no noise is detected, the input signal is deemed to be clean and thus is kept unchanged as the output signal. This is depicted in Figure 4.1. Our noisy-speech detector is convolutional and described in Section 4.1, and the noise removal network is Conv-TasNet, explained in Section 4.2.

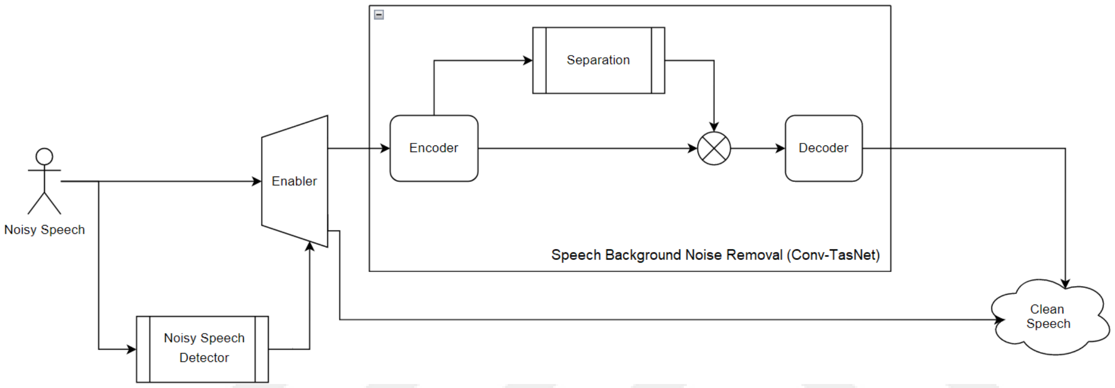


Figure 4.1 Detector-driven Model Overview

4.1 Convolutional Background Noisy-speech Detector

Our noisy-speech detector model is a lightweight convolutional network. The architecture is given in Figure 4.2. We use four convolution layers, one pooling layer, and one dense layer, with a softmax activation at the output. The model's input shape is 1 x 16000, corresponding to two seconds of audio due to the sampling rate of 8 kHz. After the convolution layers, a global pooling layer performs averaging over the input volume for each channel. Then a fully connected dense layer is applied to produce a detection outcome. At the output, "1" means that input is classified as noisy, and "0" is as clean.

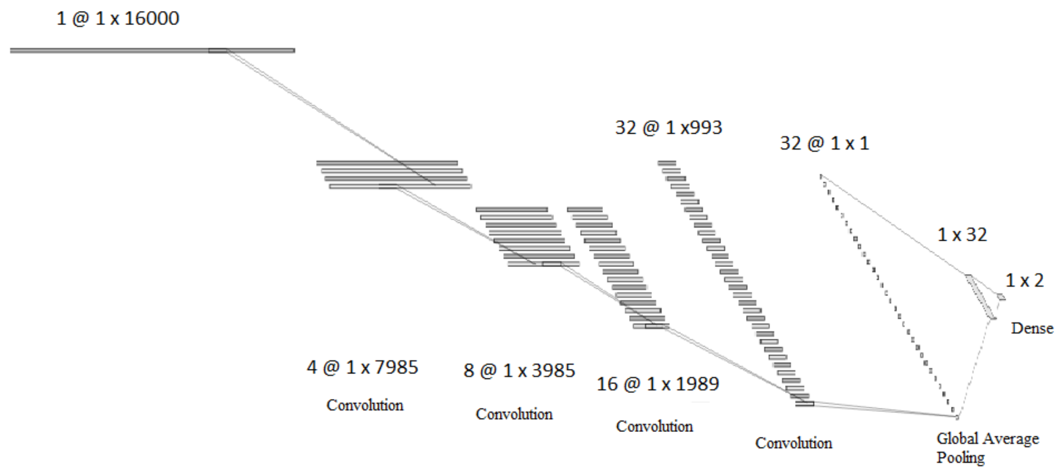


Figure 4.2 Noisy-speech detector network with example layer sizes

Using only four 1-D convolution layers and one small dense layer, we designed a very light network that can yield good detection performances for our task, as will be shown in the experiments in Chapter 5. We empirically find some good distributions of layer resolution and channels count across convolutional layers.

4.2 Fully Convolutional Time-domain Audio Separation Network

Conv-TasNet (Luo & Mesgarani, 2019) is the fully convolutional version of TasNet (Luo & Mesgarani, 2018) that has a trainable encoder-decoder and a separation module, as shown in Figure 4.3. Due to some issues, the authors have re-designed the separation module with substantial modifications. First, the original TasNet's separation module had been implemented based on Long Short-Term Memory (LSTM), which causes long training durations. The second issue is the rapid rise of the test time computation cost with increasing model complexity. Third, long term dependency modeled by LSTM can also cause inconsistent accuracies. Because of these three problems, the authors have replaced the separation module with dilated casual convolutional layers. There have also employed STFT and inverse STFT in place of the encoder and decoder modules, respectively, as an alternative to Conv-TasNet.

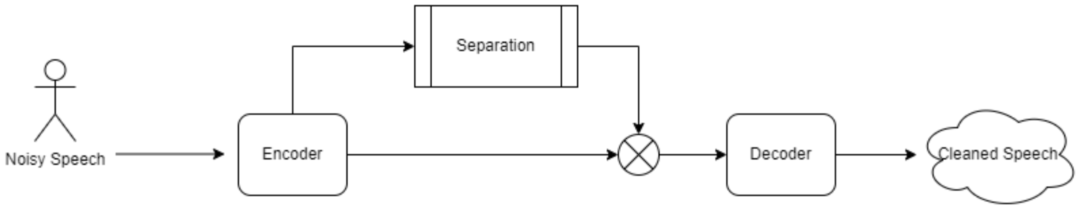


Figure 4.3. Block Diagram of Conv-TasNet

Conv-TasNet has three parts, which are encoder, separator, and decoder. First, the encoder module transforms the speech waveform into a feature space by 1-D convolution. Then, the separator takes the encoder output and calculates a mask to separate speech and noise sources via element-wise multiplication. Finally, after masking out noise-related features, the decoder module reconstructs from the masked features a clean waveform (see Figure 4.4).

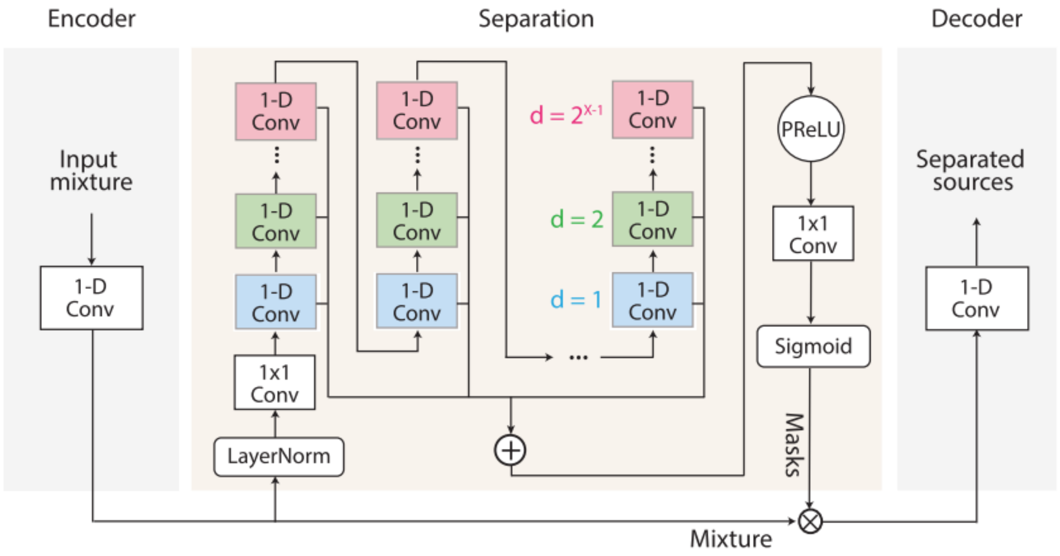


Figure 4.4. Overview of Conv-TasNet (Luo & Mesgarani, 2019)

The model design assumes the additive noise model, $\mathrm{X(n)} = \mathrm{S(n)} + \mathrm{N(n)}$, where $\mathrm{X(n)}$ is the input signal, $\mathrm{S(n)}$ is the speech signal, and $\mathrm{N(n)}$ is the noise signal at time step n. The signal can be divided into $\boldsymbol{T}$ frames of length $\boldsymbol{L}$; thus, it is represented by $\mathrm{X} \in \mathrm{R}^{\mathrm{LxT}}$.

4.2.1 Encoder and Decoder

The encoder is a trainable 1-D convolutional block that has **N** filters. It is used to represent input signal in feature space, i.e., to transform $\mathrm{X} \in \mathrm{R}^{\mathrm{LxT}}$ to $\mathrm{W} \in \mathrm{R}^{\mathrm{NxT}}$ by $\mathrm{U} \in \mathrm{R}^{\mathrm{NxL}}$ as

$$W = UX . \tag{3}$$

The decoder does the opposite of the encoder, i.e., reconstructs the signal from its feature space representation by means of transposed convolutional layers. Before reconstruction, the mask generated by the separator is applied to the encoded signal to separate the speech from background noise. Then the estimated speech signal is reconstructed by multiplying the separated signal in the feature space, $\mathrm{Z} \in \mathrm{R}^{\mathrm{NxT}}$, by $\mathrm{V} \in \mathrm{R}^{\mathrm{LxN}}$ as

$$\hat{S} = VZ . \tag{4}$$

4.2.2 Separator

As described in Figure 4.4, this module takes the encoder output $\mathrm{W} \in \mathrm{R}^{\mathrm{NxT}}$, estimates a separation mask using 1-D stacked dilated convolution layers (also called Temporary Convolution layers (TCN)) represented by $\mathrm{M} \in \mathrm{R}^{\mathrm{NxT}}$, then performs element-wise multiplication:

$$Z = M \circ W. \tag{5}$$

However, before the convolutions, first, layer normalization is performed due to its benefits in training as well as for the generalization performance (Ba et al., 2016). Also, after the dilated convolutions, the parametric rectified linear unit (PRelu) activation function, which includes a learnable parameter, is applied (Y.-D. Zhang et al., 2018).

The separator is shown in more detail in Figure 4.7. The initial part of the separation module has a 1x1 convolution block in order to transform from N channels data to B channels data. Then the TCN architecture with **X** convolutional layers of dilation factors $\mathbf{1}, \mathbf{2}, \ldots, \mathbf{2}^{\mathbf{X-1}}$ and **R** repetitions is applied. Hence, we have $\mathbf{X}\ \boldsymbol{x}\ \mathbf{R}$ times 1-D blocks in the separation module.

Figure 4.5 and Figure 4.6 show some generic dilated convolution examples from literature. The model's receptive field is enlarged by using dilated convolutions. Large receptive fields are necessary for good performance. We need to have big kernels for large receptive fields with normal convolutions. However, this quickly increases the number of parameters. Instead, dilated convolutions are applied, which enlarge the receptive field without increasing the kernel sizes. This way, without recurrent models like LSTM, long term patterns can be conveniently learned via very simple models. In this thesis, we employ non-casual convolutions, but it is possible to apply casual convolutions as well to avoid delays as they do not require future samples for prediction. By accessing future frames, non-causal convolutions obtain better performances (Rethage et al., 2018).

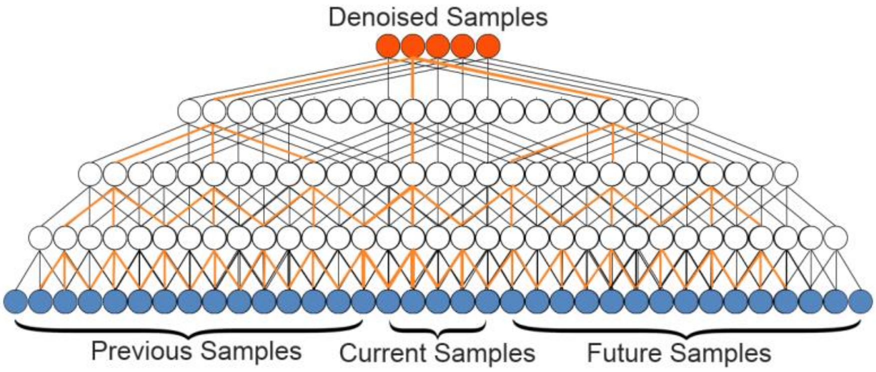


Figure 4.5 Dilated non-casual convolution with dilation factor of 8 (orange lines: non-causal, dilated convolutions predicting a single sample) (Rethage et al., 2018)

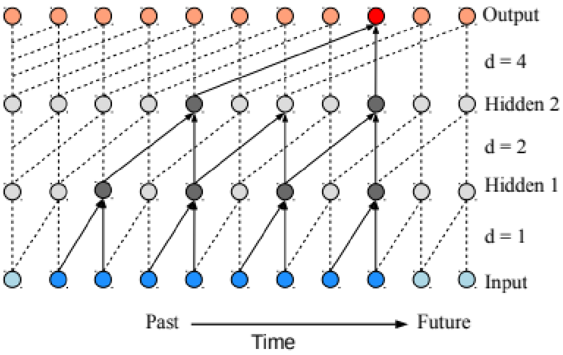


Figure 4.6 Dilated casual convolution with dilation factor of 4 and filter size of 2 (Pandey & Wang, 2019a)

In Figure 4.7, we draw a more detailed version of the separation module shown in Figure 4.4. The blue 1-D conv blocks are of one dilation, and the dilation extent while going up in the hierarchy.

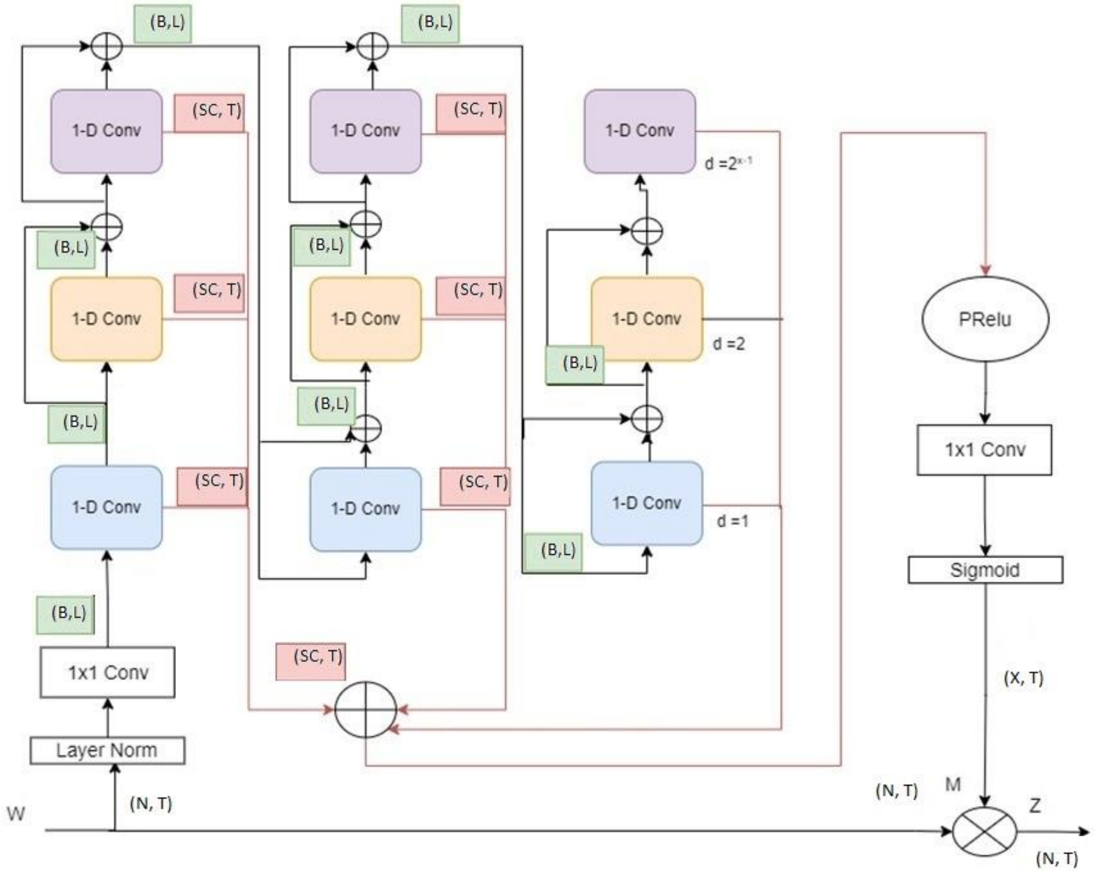


Figure 4.7. Detailed version of the separator Module in Conv-TasNet when X:3 and R:3 (B: number of channels in the bottleneck and residual paths' 1x1-conv blocks, SC: number of channels in skip-connection paths' 1x1-conv blocks, T: number of frames)

Figure 4.8 shows the structure of the 1-D Conv block. It has two outputs: skip-connection path and residual path. Skip-connection paths of all the 1-D Conv blocks are later summed up and form the output of the separation module. On the other hand, each residual path is fed to the following 1-D Conv block.

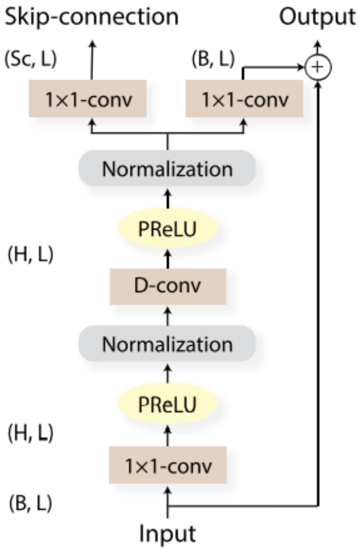


Figure 4.8. 1-D Block Architecture (H: Number of channels in conv. blocks, B: Number of channels in skip-connection paths' 1x1-conv blocks, L: Length of the filters in samples) (Luo & Mesgarani, 2019)

Rather than normal convolution operations, the authors replaced all convolutions with depthwise separable convolution (Chollet, 2017) to considerably reduce the number of learnable parameters and floating-point operations.

CHAPTER 5
EXPERIMENTAL RESULTS

In this chapter, we present our experimental evaluations, which justify the use of a detector for significant efficiency gain. First, we describe the dataset employed for the experimentation in Section 5.1. Then, in Section 5.2, we compare the performance and efficiency of different Conv-TasNet models and pick the optimal one. In Section 5.3, we evaluate different hyper-parameters of our noisy-speech detector. Finally, we evaluate the proposed detector-driven approach in terms of performance and efficiency in Section 5.4.

5.1 Datasets

There are several common datasets used for background noise removal studies in the literature, differing in use purposes, content, and size. We use a combination of two datasets for our experimental evaluation. These datasets are Voice Bank (Veaux et al., 2013) for clean-speech sources and Demand (Thiemann et al., 2013a) for background noise sources, which were mixed by Valentini-Botinhao (2017). Note that many prior studies have also employed this combined dataset (Chao et al., 2022; Fu et al., 2021; Koyama et al., 2020; Pascual et al., 2017; Rethage et al., 2018; Yu et al., 2021).

5.1.1 Voice Bank Dataset

The University of Edinburgh developed the Voice Bank corpus (Veaux et al., 2013) for people with speech impairment. It has also been used for speech enhancement in Koyama (2020). It contains British English and more than 300 hours of recordings from nearly 500 speakers. The age-gender and geographical distributions are shown in Figure 5.1 and Figure 5.2, respectively.

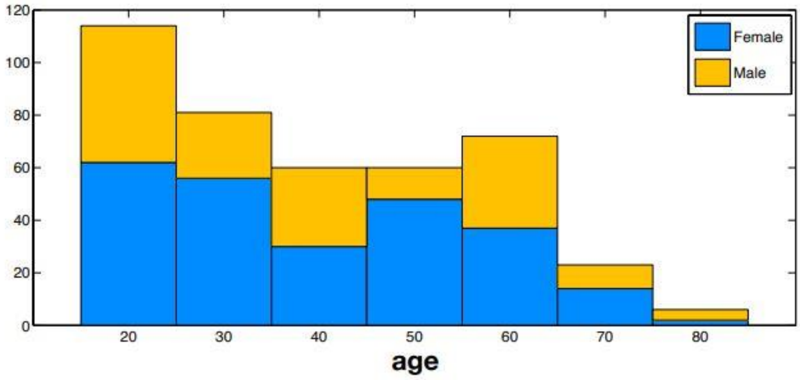


Figure 5.1 Voicebank age distributions (Veaux et al., 2013)

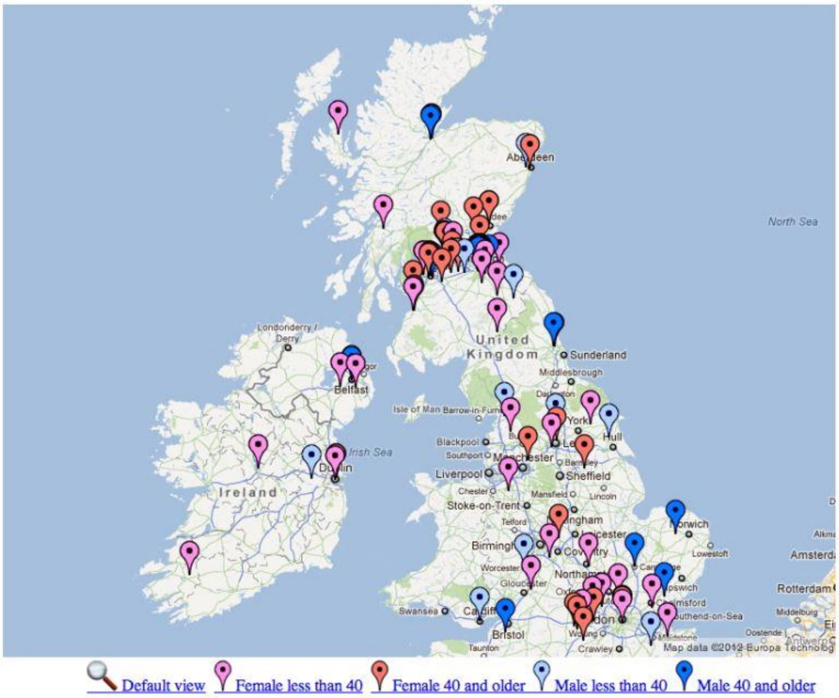


Figure 5.2 Voicebank location distributions (Veaux et al., 2013)

5.1.2 Demand Dataset

The Diverse Environments Multi-Channel Acoustic Noise Database (DEMAND) Dataset (Thiemann et al., 2013a) was designed and collected by Thiemann (2013). Unlike other datasets, it contains multi-channel (16-channel) environmental noise samples (can be downloaded at http://www.irisa.fr/metiss/DEMAND (Thiemann et al., 2013b)).

Noise clips were sampled at 48 kHz by planar 16 microphones in six categories. Clips from four categories (Domestic, Office, Public, and Transportation) were recorded indoors, and the rest (Street and Nature) were outdoors. Each clip is five minutes long.

There are six environment sound categories involving three different scene audio clips per category, resulting a total of 18 clips as described in Table 5.1.

Table 5.1 Noise Database Structure

Category	Environment	Description
Domestic	DKITCHEN	inside a kitchen during the preparation of food
	DLIVINGR	inside a living room
	DWASHING	domestic washroom with washing machine running
Office	OHALLWAY	a hallway inside an office building with occasional traffic
	OMEETING	a meeting room while the microphone array is discussed
	OOFFICE	a small office with three people using computers
Public	PCAFETER	a busy office cafeteria
	PRESTO	a university restaurant at lunchtime
	PSTATION	the main transfer area of a busy subway station
Transportation	TBUS	a public transit bus
	TCAR	a private passenger vehicle
	TMETRO	a subway
Nature	NFIELD	a sports field with activity nearby
	NPARK	a well-visited city park
	NRIVER	a creek of running water
Street	SCAFE	the terrace of a cafe at a public square
	SPSQUARE	a public town square with many tourists
	STRAFFIC	a busy traffic intersection

5.1.3 Pre-processing and Partitioning of the Dataset

We use the mixed dataset of (Valentini-Botinhao, 2017), who combined the Voice Bank clips with the environmental sounds in the DEMAND dataset. It is called as the VBD dataset in the literature and has been employed by many authors (Chao et al., 2022; Fu et al., 2021; Koyama et al., 2020; Pascual et al., 2017; Rethage et al., 2018; Yu et al., 2021). It contains 30 speakers with an equal number of males and females. Originally, the VBD dataset is sampled at 48 kHz. The lengths of the clips vary between one to six seconds. Due to the memory constraints during our experimentations, we down-sampled them at 8 kHz and trimmed the duration to two seconds. However, we repeated the same clip to fill the gap when a clip was shorter than two seconds.

VBD dataset has two partitions as training and test sets. The training set has 11572 clips from 28 speakers (14 males and 14 females) for ten noise types: two of them are artificially generated, and the rest are environment sounds. Artificial ones are the babble and speech-shaped noise samples, whereas recordings from cafeteria, kitchen, meeting, metro, restaurant, busy subway station, traffic, and car scenes are used as environment sounds. These noise types are described in Table 5.1. The test set has 824 clips from two subjects (one male and one female) for five noise types: bus, café, living room, office, and town square environments. Also, while the training set has four SNR levels (0 dB, 5 dB, 10 dB, 15 dB), test set has five SNR levels (2.5 dB, 7.5 dB, 12.5 dB, 17.5 dB) different from the training set. Thus, having noise types and SNR levels of the test set unseen in the training set, experimental evaluations of the generalization ability become more realistic and reliable. However, since we also need a validation set in our experiments, following prior studies in the literature (Koyama et al., 2020), we took 300 clips from the training set for validation by picking an equal number of clips from each noise type and SNR level. This slightly reduces our training set size to 11272 clips.

5.2 Performance and Complexity Comparison of Conv-TasNet Models

In this chapter, six different model configurations of Conv-TasNet are tested. Based on these experiments, we determine an optimal set of hyper-parameters regarding the performance and efficiency, which will be employed in our detector-driven experiments.

The hyper-parameters that determine the model's architecture are listed in Table 5.2. According to the table, for instance, if L=32, frame duration is 4 ms due to the 8 kHz sampling rate (32/8000 = 0.004). Similarly, with St = 16, the stride size is 2 ms. Thus, at every 2 ms step, the encoder yields an N-dimensional feature vector. B, H, and Sc are parameters of 1-D conv blocks; X and R adjust the number of separation blocks, which is visualized with X = 3 and R = 3 in Figure 4.7. If X is increased, the separation module can see a longer duration due to increased receptive field size. Therefore, the model can get information about all the frames in a clip when X is big enough so that the receptive field size becomes equal to or bigger than the clip size.

Table 5.2 Architecture hyper-parameters of the Conv-TasNet

Symbol	Description
N	Number of filters in the encoder
L	Length of the filters (in samples)
St	A stride of the convolution. (Overlapping, default L/2)
B	Number of channels in the bottleneck and the residual paths' 1x1-conv blocks
Sc	Number of channels in skip-connection paths' 1x1-conv blocks
H	Number of channels in convolutional blocks
P	Kernel size in convolutional blocks
X	Number of convolutional blocks in each repeat
R	Number of repeats

Often there is a trade-off between complexity and performance. This trade-off has been shown for the main hyper-parameters of the Conv-TasNet model (Luo & Mesgarani, 2019), as explained below.

- **L:** We can increase the model's performance by choosing the filters' size (L) low. However, as this will cause too many frames because of short length frames, it considerably increases the training time.
- **B, H:** Commonly, the number of bottleneck (B) channels are chosen as small, and the number of channels in the convolutional blocks (H) is bigger. For instance, the ideal ratio of H/B was found to be about 5 in (Sandler et al., 2018).
- **Sc:** Increasing the number of channels in the skip-connections block (Sc) improves performance but can greatly increase the model complexity. Therefore, a compromise must be made between performance and complexity.
- **R:** A larger receptive field size usually results in higher performance due to more information captured. An effective way of increasing the receptive field size is to increase the number of repetitions (R) since it creates a deeper model hence more learning capacity.

In the Conv-TasNet table in Appendix, we list the three complexity measures, i.e., FLOPS, number of total learnable parameters, and estimated memory requirements, for varying X and R values but fixing B, H, and Sc.

In the experiments, our evaluation metric is Scale-Invariant Signal-to-Noise Ratio (SI-SNR), and the loss function is negated SI-SNR. SI-SNR is commonly employed in speech enhancement and speaker separation studies (Le Roux et al., 2019; Ma et al., 2020). It brings scale-invariance to SNR by scale normalization depending on the clean-speech source as formulated below.

$$s_{target} = \frac{\langle \hat{s}, s \rangle s}{||s||^2} \quad (6)$$

$$e_{noise} = \hat{s} - s_{target} \quad (7)$$

$$SI - SNR = 10 \log_{10} \frac{||s_{target}||^2}{||e_{noise}||^2} \quad (8)$$

where S and Ŝ are the clean and estimated speech signals, respectively.

Table 5.3 shows all the configurations that we tested. By adequate adjustment of the hyper-parameters, these models vary according to the number of learnable parameters and according to the total number of floating-point operations required per a single forward pass as total multiplications and additions (FLOPS). FLOPS are calculated by using the PyTorch library (Paszke et al., 2017), setting the batch size to one on the test set. We express this computation load throughout the thesis in the units of mega FLOPS (MFLOPS). While the learnable parameters affect the model capacity, hence the performance, as well as the training duration, MFLOPS affects test time efficiency and speed. Here, the model named as C6 is based on the hyper-parameters of the best performing model in Luo & Mesgarani (2019), except for L, which is set as 32 instead of 16. C6 has 5 million learnable parameters. In the code names, the smaller value of the integer suffix means the Conv-TasNet model is simpler according to the number of learnable parameters. As seen in Table 5.3 that, model complexity and MFLOPS vary similarly since we adjust different model configurations accordingly. However, Table 5.3 also shows a different model named as S, which stands for STFT encoder/decoder in place of learnable convolutional encoder/decoder. We also experiment with it since it is often chosen as an efficient model due to the STFT, as can be seen from the MFLOPS calculation in Table 5.3. However, although it has very low test-time complexity, the number of learnable parameters is at the same level as C6.

Table 5.3 Hyper-parameter values for different values of total learnable parameters of convolution Conv-TasNet. (S: replacement of decoder/encoder by STFT/ISTFT, MFLOPS: total mega floating-point operations per a single forward pass)

	C1	C2	C3	C4	C5	C6	S
Learnable Parameters	308K	718K	822K	1.2M	1.3M	5M	5M
MFLOPS	270	670	770	1130	1280	9580	610
Hypers Parameters							
N	512	512	512	512	512	512	512
L	32	32	32	32	32	32	192
St	16	16	16	16	16	16	128
B	64	64	64	64	64	128	128
Se	64	64	64	64	64	128	128
H	256	256	256	256	256	512	512
P	3	3	3	3	3	3	3
X	4	4	7	7	8	8	8
R	1	3	2	3	3	3	3

In Table 5.4, we show SI-SNR scores obtained after training. For the sake of experimentation time, we used the trained encoder/decoder modules of C6 for all the convolutional models as pre-trained encoder/decoder, which are shown in the "Pre-trained" sections in Table 5.4. However, for C2 and C4, we also experimented without using pre-trained modules, as shown in the "Not pre-trained" part of Table 5.4. We see that use of pre-trained modules causes a small reduction in performance, and moreover, it did not help much to reduce the training time in practice. When we compare the results of the STFT-based (Short-time Fourier Transform) encoder/decoder, we see that our training with STFT clearly could not compete with learnable convolutional encoder/decoder models; therefore, it is not considered in the rest of the thesis. According to Table 5.4, while C6 is the best performing model with 20 dB SI-SNR on the test set, C2 and C4 are very close follow ups with 19.9 dB.

Interestingly, while validation scores are lower than the training set scores for all the models, we observe that test scores are the highest. This starkly contrasts common expectations since generalization performance on the test set is usually significantly lower. However, there is a special nuance in our evaluation protocol, which is the dB levels chosen for the test to make the test set more different than the training set, in addition to using a completely different set of environment noise categories in testing. As detailed in Section 5.1.3, on average, the SNR of the test set is 2.5 dB higher than that of the training set. Therefore, we believe this is the main reason for higher test set SI-SNR scores, even though the test set noise types are unseen during training.

On the other hand, we see from Table 5.3 that C2 is the most efficient among the three, as C2 requires 670 MFLOPS, C4 requires 770 MFLOPS, and C6 requires 9580 MFLOPS. Therefore, in the rest of the thesis, we conduct our evaluations using C2.

Table 5.4 Training, validation, and test scores (SI-SNR) of different Conv-TasNet configurations. Pre-training is by re-using the trained encoder/decoder of C6. S: replacement of decoder/encoder by STFT/ISTFT.

		C1	C2	C3	C4	C5	C6	S
Training Score	Pre-trained	16.4	16.5	17.1	17.8	--		
	Not pre-trained	--	18.0	--	18.1	16.9	**18.4**	16.2
Validation Score	Pre-trained	16.2	16.6	17.0	17.0	--		
	Not pre-trained	--	16.8	--	17.2	16.6	**17.3**	13.6
Test Score	Pre-trained	19.4	19.8	19.7	19.7	--		
	Not pre-trained	--	19.9	--	19.9	19.8	**20.1**	16.3

In Table 5.5, we show SI-SNR performances of Conv-TasNet models from the literature together with our C6 and C2 models (results from a few other papers are not included since they report with different performance scores). Note that there are differences in model hyper-parameters, protocols, datasets, and pre-processing. Therefore, the results are not directly comparable. For instance, some studies do re-sampling at 16 kHz and others at 8 kHz; authors trim the audio clips to different lengths, or studies with WSJ0-2mix involve separation task of two different speakers from each other. We also list the model sizes and employed datasets in Table 5.5. Despite of these differences in the experiment setups, this table still provides some useful information for the assessment of our results. The scores approximately range from 15 dB to 20 dB, and the performance usually drops with increased difficulty (e.g., when the task involves speaker separation). The closest to our work is of (Koyama et al., 2020), as they carried out the same task on the same dataset with a similar protocol. Therefore, their performance score is not surprisingly the closest to ours. Their model with 5.1 M parameters obtains 19.0 dB, while our C6 model, which is equivalent in size, obtains 20.1 dB. The small score gap might be due to some differences in the experiment setups, model architectures, and training processes. Thus, Table 5.5 shows that the performance of our Conv-TasNet is comparable with the literature and that the slight performance drop with C2, after our drastic simplification, is negligible.

Table 5.5 Conv-TasNet performances in the literature. There are differences in model hyper-parameters, protocols, datasets and pre-processing.

Author	Model Size	SI-SNR Score	Dataset
(Deng et al., 2020)	5.0M	14.4	WSJ0-2MIX
(Koizumi et al., 2021)	17.8M	15.3	LibriVox + freesound
(Luo & Mesgarani, 2019)	5.1M	15.3	WSJ0-2MIX
(Kadıoğlu et al., 2020)	9.7M	16.3	WSJ0-2MIX
(G.-P. Yang et al., 2019)	10.0M	16.6	WSJ0-2MIX
(Kadıoğlu et al., 2020)	9.7M	17.1	LibriTTS
(Koyama et al., 2020)	5.1M	19.0	VBD Dataset
C2 (ours)	0.7M	19.9	VBD Dataset
C6 (ours)	5.1M	20.1	VBD Dataset

Interestingly, scores in the range of 25-30 on the clean-speech clips indicate that there may also be significant corruptive effects on the speech signal due to the suppression artifacts. In fact, although we observed some very slight cracking-like artifacts, in terms of audio perception, they are rather negligible. Including clean-speech in the suppression network might perhaps help to mitigate such artifacts. Nevertheless, these cases can be automatically avoided if the absence of background noise is detected. (see in Figure 5.3).

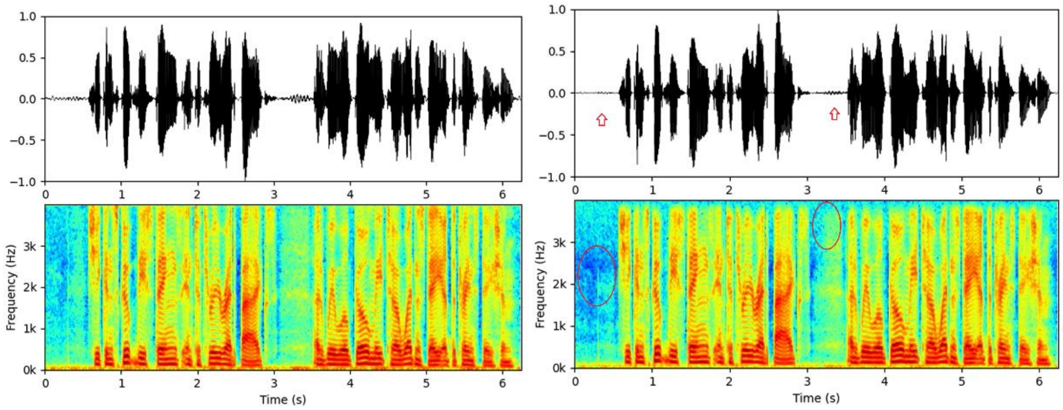


Figure 5.3 Spectrogram and waveform of an input clean-speech (right) clip and its processed output (left).

Figure 5.3 shows the output of the C2 model for an already clean-speech clip. Though marked with circles and arrows show some visible artifacts; however, it is easier to notice the differences by listening to the clips.

Figure 5.4 compares these models over different noise categories and clean-speech based on SI-SNR scores. We see that, in general, competition over the noise categories is quite similar to competition over the whole noisy set with some expected degree of variations. As anticipated, with decreasing SNR, the output of all the models degrade. The performances are lower in café and living room environments compared to office, square, and bus environments. The reason might be the strong interference of other speakers in the café and living room environments. As explained in Section 3.2, suppression of unwanted human speech is quite challenging.

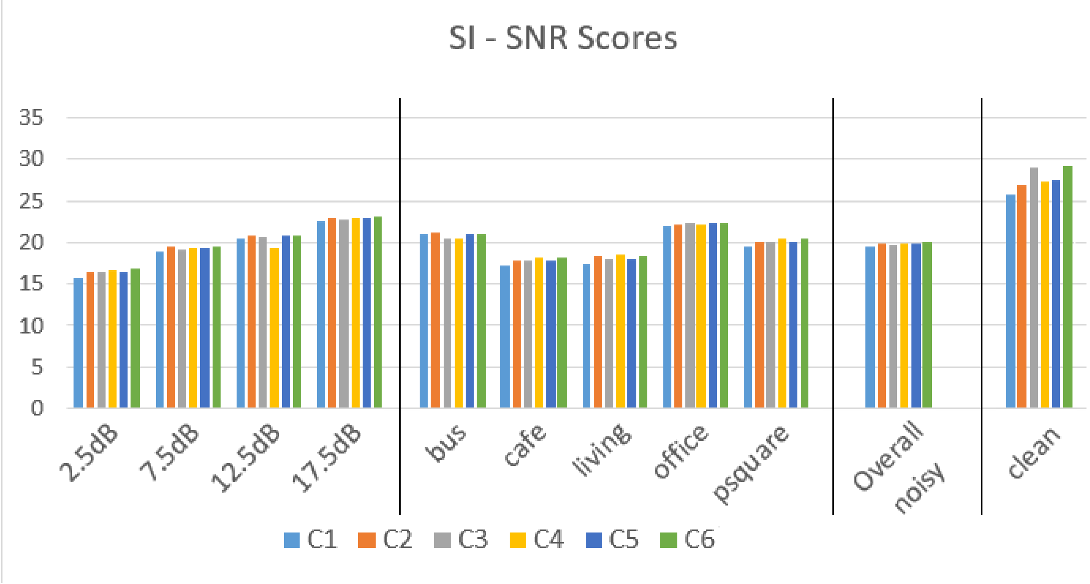


Figure 5.4 SI-SNR scores (higher is better) on the test are shown for sub-categories of the sample set divided according to SNR dB level (higher means less noise power) and noise type.

We show some example noise removal results for the "bus" environmental noise by means of spectrograms and waveforms at SNR of 17.5 dB in Figure 5.5, at SNR of 12.5 dB in Figure 5.6, and at SNR of 2.5 dB in Figure 5.7. In Figure 5.5, there are almost no significant differences between clean, noisy, and processed clips due to very low noise power. In Figure 5.6, which is of mid-level noise power, we clearly observe successful background noise removal. In the most difficult example due to high-power noise that is shown in Figure 5.7, again, we clearly see the cancellation of the noise, though this time, there also happens some degradation of the speech signal but without breaking the intelligibility.

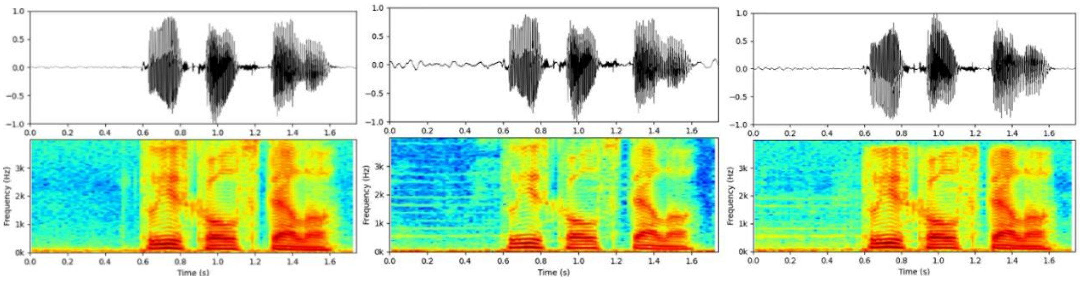


Figure 5.5 Audio waveforms and spectrograms of clean-speech (left) noisy-speech of bus noise at an SI-SNR level of 15.42 dB (middle), and cleaned-speech SI-SNR:18.66 (left)

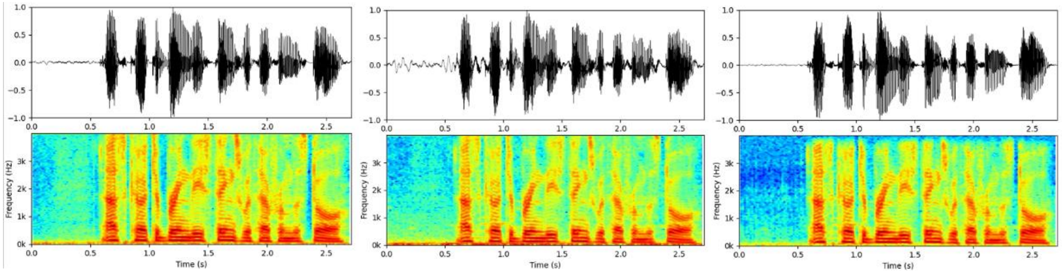


Figure 5.6 Audio waveforms and spectrograms of clean-speech (left) noisy-speech of bus noise at an SI-SNR level of 11.3 dB (middle), and cleaned-speech SI-SNR: 17.43 (left)

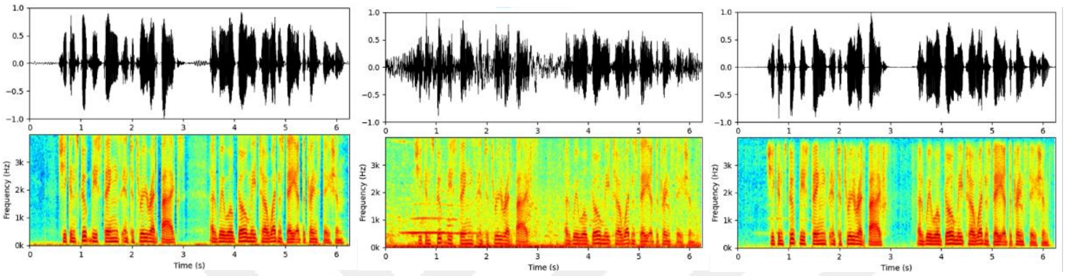


Figure 5.7 Audio waveforms and spectrograms of clean-speech (left) noisy-speech of bus noise at an SI-SNR level of 1.78 dB (middle), and cleaned-speech SI-SNR: 15.91 (left)

5.3 Evaluation of Noisy-speech Detection

We evaluate different configurations for our noisy-speech detector based on false negatives, false positives, receiver operating characteristics (ROC) curve, and the F1 score. Noisy-speech is represented by the positive label and clean-speech by the negative label. Thus, false positive means that when the clip is clean, it is predicted as noisy, and false negative means that when the clip is noisy, it is predicted as clean. Table 5.6 explains their calculations. High FNR means that we miss noisy clips. Therefore, they would not be cleaned, and the output quality would be bad, resulting in poor performance.

On the other hand, high FPR means that many clean clips are detected as noisy, which would unnecessarily call the expensive noise suppression network and possibly introduce some artifacts on the clean audio, as shown in Section 5.2. To avoid missing noisy clips without causing considerable FPR, we aim at 1% FNR, i.e., a very low rate. We realize this by finding a threshold value that satisfies this condition on the validation set. Then based on the threshold, we evaluate FNR and FPR values on the test set.

While FNR, FPR, or scores like accuracy are threshold dependent, we also calculate the area under the ROC curve (AuC) as a threshold-independent measure. Here, ROC is based on FNR, and FPR measurements vary across all possible threshold values. Moreover, we also evaluate the F1-score as an alternative measure, which is the harmonic mean of precision and recall, calculated as below.

$$\text{Precision} = \frac{TP}{TP+FP} \tag{9}$$

$$\text{Recall} = \frac{TP}{TP+FN} \tag{10}$$

$$\text{F1} = 2\,.\,\frac{Precision * Recall}{Precision + Recall}\,. \tag{11}$$

Table 5.6 Detector evaluation measures

Label	Explanation
TP (True Positive)	Label and prediction are noisy.
TN (True Negative)	Label and prediction are clean.
FP (False Positive)	Label is clean, and prediction is noisy.
FN (False Negative)	Label is noisy, and prediction is clean.
N (All Negatives)	FP + TN
P (All Positives)	FN + TP
TPR (True Positive Rate)	TP / P = 1 - FNR
FNR (False Negative Rate)	FN / P = 1 - TPR
TNR (True Negative Rate)	TN / N = 1 - FPR
FPR (False Positive Rate)	FP / N = 1 - TNR
PPR (Positive Prediction Rate)	(TP+FP) / (P+N)

We determined five different noisy-speech detector configurations by altering the number of input and output channels and kernel and stride sizes, as shown in Table 5.7, with their FLOPS and the number of learnable parameters. We see that while the model with code name D1 is the most efficient, D3 is the least efficient.

Table 5.7 Noisy-speech detector configurations (MFLOPS: Mega floating-point operations per a single forward pass, I: input channels, O: output channels, K: kernel size, S: stride)

Model		D1				D2				D3				D4				D5			
Convolutional layers		**I**	**O**	**K**	**S**	**I**	**O**	**K**	**S**	**I**	**O**	**K**	**S**	**I**	**O**	**K**	**S**	**I**	**O**	**K**	**S**
		1	4	32	2	1	4	3	1	1	4	32	1	1	4	32	2	1	4	16	2
		4	8	16	2	4	8	3	1	4	8	16	1	4	8	32	2	4	8	16	2
		8	16	8	2	8	16	3	1	8	16	8	1	8	16	32	2	8	16	16	2
		16	32	4	2	16	32	3	1	16	32	4	1	16	32	32	2	16	32	16	2
Model Size	MFLOPS	7.26				33.39				60.15				29.21				14.83			
	Learnable Parameters	3838				2154				3838				21758				10941			

In the experimentation sets, we add clean clips as many as noisy clips, i.e., the sample size is doubled. Thus, the training set has 22540 samples, the validation set has 600 samples, and the test set has 1648 samples. The training batch size is 16. All models are trained with Adam optimizer, and the learning rate is set to 0.001. The loss function is cross-entropy. We used early stopping to avoid over-learning. Learning curves, confusion matrices and the chosen threshold values of all the detectors are given in Appendix.

Table 5.8 shows performance comparisons of all the detectors. In terms of the threshold-independent AuC, there is no significant difference between the models, as they all have demonstrated very high performance. This also means that their ROC curves are almost identical, very close to the ideal shape. However, according to the threshold-dependent F1 scores, D2 and D4 are the best performers. For all the threshold-dependent evaluations, we fix FNR at 1% on the validation set; that is, we determine the detector thresholds on the validation set. We use a very low FNR rate as for the detector target instead of using FPR; because, above all, the detector should not miss noisy samples for cleaning in order to have good quality output. Only then we should aim at low FPR not to have unnecessary activation of noise removal since there is a trade-off between FNR and FPR.

Regarding efficiency, the best detector is the one which obtains the lowest FPR with the same FNR level. However, there is also a caveat. Since not all noise instances have the same SNR or perceptual effect, in some cases, false negatives can actually be negligible. Therefore, we will present a final assessment by evaluating the complete detector-driven noise removal in Section 5.3. However, detector-only evaluation is needed to acquire insight into the whole system.

We see in Table 5.8 that even though we set FNR at 1% on the validation set, test set FNRs are significantly higher, varying across the detector models. While D2 is the best in terms of the FNR with 1.58%, which is expected to produce the best detector-driven suppression quality, it has a high FPR of 8.74%. On the other hand, D5 has the lowest FPR with 2.73% and a moderate FNR with 2.43%.

Table 5.8 Detector performances (FPR is for fixed FNR at 1% on the validation set)

Model	D1	D2	D3	D4	D5
F1	0.95	**0.96**	0.92	0.96	0.90
AuC%	100.00	99.00	99.00	99.00	99.00
FNR%	2.55	**1.58**	2.79	3.52	2.43
FPR%	3.76	8.74	3.52	3.28	**2.79**

Finally, we examine the performance of our detectors for different SNR levels as well as at different environment noise types in Figure 5.8. We observe close to 100% true detection for 2.5 dB, 7.5 dB, and 12.5 dB SNR levels, while it suddenly drops to about 90% on average for the 17.5 dB SNR level. This moderate deterioration means that when there is not much noise, roughly at 10% of the time, input will not be processed further by Conv-TasNet. These cases will not have considerable bad effects on the output quality since the SNR is already high but will help to gain some efficiency. Secondly, we observe close to 100% true detection in the "living" category, but a significant detection failure rate is seen for each of the other environment noise types. When we look at Figure 5.8, where Conv-TasNet performances of the same sub-categories are shown, we see a somehow quasi-inverse relationship such that the categories that obtain low true detection rates attain relatively higher SI-SNR scores.

This is rather a positive outcome since even if some noisy clips are missed, we expect usually they will have a minor noise issue.

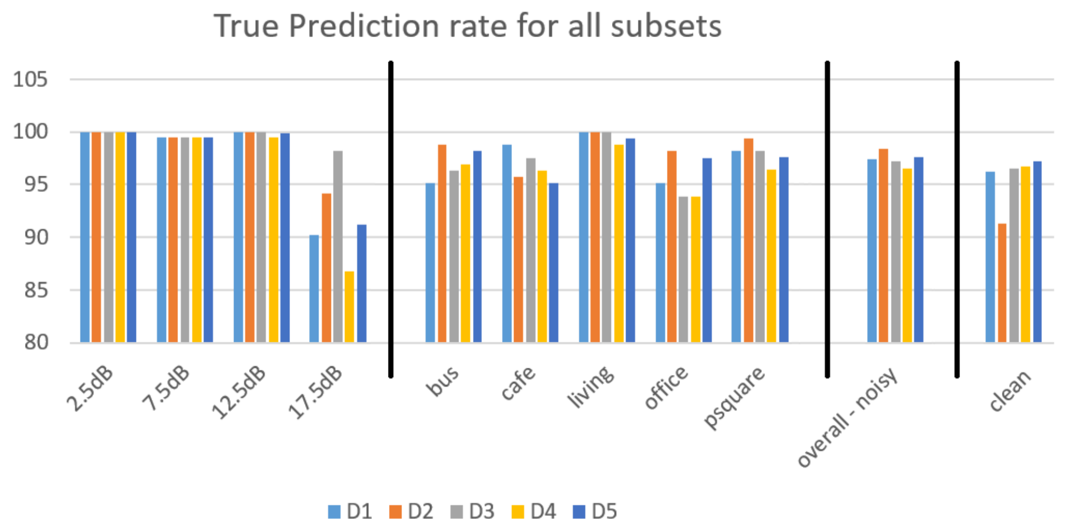


Figure 5.8 True prediction rates of different noise sub-categories of the five detectors for fixed FNR at 1% on the validation set.

5.4 Assessment of Detector-Driven Conv-TasNet

In this section, we use one-by-one the detectors that we have trained in Section 5.3 for the detector-driven noise removal scheme. We evaluate and compare detector-driven noise removal performances together with their test-time efficiencies. Although we know which detector has lower FLOPS than others, this does not mean that the same ordering will be valid for the efficiency comparison of detector-driven models. This is because depending on the FPR and FNR values, computationally much more demanding Conv-TasNet can be activated at a higher or a lower rate. Note that, therefore, FNR and FPR also affect the noise suppression performance. FPR and TPR (i.e., 1-FNR) are the activation rates on the clean and noisy clips, respectively. Moreover, we can use a positive prediction rate (PPR = positive predictions / sample size) for the overall activation rate on the whole test set. As Conv-TasNet is not active all the time due to our detector-driven scheme, we cannot express the computations by counting for a fixed FLOPS as done commonly, but we can calculate an average FLOPS over time. Then, given any activation rate, we can estimate an average FLOPS for the computations of the whole detector-driven model as

$$\mathrm{F} = \mathrm{F_d} + \mathrm{F}_r \cdot A \qquad (12)$$

where A is the activation rate of the noise suppression network, F_r is the FLOPS of the noise suppression network, and F_d is the FLOPS of the detector network (since the detector is active all the time F_d has no multiplier).

In the top section of Table 5.9, we show the estimated FLOPS of detector-driven C2 models for a balanced mix of clean and noisy samples (i.e., on our whole test set) as $\mathrm{F_{PPR}}$ which is calculated by substituting PPR as for the A in Equation 12. Similarly, we show for only the clean samples by substituting with FPR (i.e., for unnecessary activations) and for only the noisy samples by substituting with TPR. On the other hand, the bottom section of Table 5.9 shows the corresponding performances on these three sets. Moreover, in addition to the five detector-driven models, we also show the results of the Conv-TasNet C2 without the application of any detector with the title "No Detector".

We see from $\mathrm{F_{PPR}}$ values that detector-driven models are about two times more efficient than direct noise suppression. $\mathrm{F_{FPR}}$ and $\mathrm{F_{TPR}}$ values tell us that this outcome is due to successful detection with relatively very lightweight (compared to Conv-TasNet) detectors. While direct suppression requires 670 MFLOPS, for instance, on average, D1 has 346.33 MFLOPS and causes only 32.47 MFLOPS due to false detections. The overhead of D1 is 7.3 MFLOPS (Table 5.7), which is about 1% of the Conv-TasNet. Notice that the FLOPS ranking in Table 5.7 is not preserved with the three MFLOPS evaluations in Table 5.9, as there are small differences in the rankings.

$\mathrm{F_{PPR}}$, $\mathrm{F_{FNR}}$ and $\mathrm{F_{TPR}}$ estimates can be used in scene-dependent efficiency evaluations. While $\mathrm{F_{PPR}}$ should be the basis if probabilities of background noise presence and absence are assumed the same; since our test set has equal amounts of these two classes, a weighted summation of $\mathrm{F_{FNR}}$ and $\mathrm{F_{TNR}}$ can be calculated to estimate the computation complexity for a given expected probability of background noise. In short, the efficiency will increase linearly with a smaller probability of noise presence. To show this, let's define the probability of noise presence as p. Then, the probability of clean-speech becomes 1 – p. Now, using Equation 12, we can express an expected average FLOPS for a given detector based on p as

$$\mathrm{F(p)} = [\mathrm{F_d} + \mathrm{F}_r \cdot TPR] \cdot p + [\mathrm{F_d} + \mathrm{F}_r \cdot FPR] \cdot (1 - p) \tag{13}$$

$$= \mathrm{F_d} \cdot [p + 1 - p] + \mathrm{F}_r \cdot [\, TPR\, .p + FPR \cdot (1 - p)] \tag{14}$$

$$= \mathrm{F_d} + \mathrm{F}_r \cdot [\, TPR \cdot p + FPR \cdot (1 - p)] \tag{15}$$

$$= \mathrm{F_d} + \mathrm{F}_r \cdot FPR + \mathrm{F}_r \cdot (TPR - FPR) \cdot p\, . \tag{16}$$

According to the linear relationship between the probability of noise presence and FLOPS given in Equation 16, if noise occurs less frequently, we get increased average efficiency. Hence, there is an inverse linear relationship between efficiency and noise presence probability.

By varying the probability variable value, we can visualize expected FLOPS for a given detector through the lines formulated by Equation 16. This helps to get a concise and informative picture of efficiency variations of alternative detector-driven models with respect to environmental noise occurrence rate, as shown in Figure 5.9. Figure 5.9 compares the five detectors over the full range of noise probabilities for our detector-driven scheme employing the C2 network. We observe different inverse-linear relationships between efficiency and background noise probability. We see that D1 and D5 are the most efficient ones as their MFLOPS trend lines are below others. We also see that D1 becomes relatively more efficient compared to D5 as the noise occurrence rate increases, though only slightly. On the other hand, the most inefficient ones are D2 and D3. While D2 is more efficient with a high noise occurrence rate compared to D3, D3 is better at low rates (having a similar FLOPS background noise probability of around 0.3 where the two trend lines intersect).

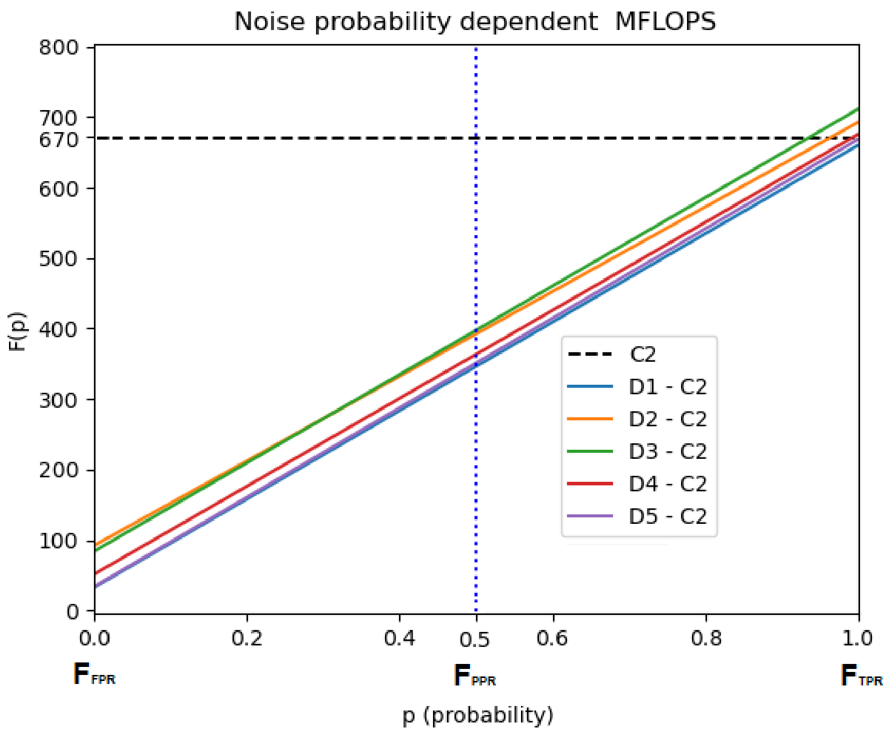


Figure 5.9 Noise occurrence probability dependent MFLOPS trend line of each detector-driven noise removal model, calculated by Equation 16, is shown in a different color. The horizontal dashed line corresponds to Conv-TasNet (C2) MFLOPS without using any detector. F_{FPR}, F_{TPR} and F_{PPR} estimate locations are shown on the probability axis.

When we look at the SI-SNR scores in Table 5.9, we see that detector-driven models obtain much higher overall scores than the direct application of Conv-TasNet, which is rather unexpected at first glance. This is due to moderate SI-SNR values of Conv-TasNet on clean clips, i.e., about 24 dB on average, whereas detector-driven models obtain more than 90 dB. However, very high SI-SNR ranges actually do not correspond to similarly big perceptual differences as there is a large degree of the non-linear relationship between SI-SNR levels and perceptual significance. The perceptual effects are understood the best by listening to the audio clips. In fact, when we calculate SI-SNR similarity between a clip with itself, we get around 170 dB in our implementation due to a numerical adjustment in the logarithm (must be infinity by exact math). However, this does not mean that there is no actual perceptual difference; slight discernable cracking noise occasionally happens, as explained in Section 5.2, where the processing of clean-speech is examined. On the other hand, SI-SNR scores of the detector-driven models on the noisy clips are very close to the direct application of Conv-TasNet; there is only about a 0.1 to 0.2 dB drop in the performances.

Based on all the performance and efficiency scores in Table 5.9, the detector-driven suppression by employing D5 as the detector seems to be the best choice, as it obtains the second lowest FLOPS and the highest SI-SNR. In terms of efficiency only, D1 is the winner, but the FLOPS difference with respect to D5 is negligible, although D1 is a two times more efficient detector, as seen in Table 5.7. D5 requires about 2% MFLOPS of the Conv-TasNet. For these reasons, we prefer the Conv-TasNet C2 driven by the detector D5.

Table 5.9 Performance and efficiency evaluations of detector-driven noise removal networks. The score is the average SI-SNR value. $\mathrm{F_d}$ is the total mega floating-point operations for a single pass (MFLOPS) for the detector. Overall MFLOPS are $\mathrm{F_{PPR}}$, $\mathrm{F_{FPR}}$ and $\mathrm{F_{TPR}}$, which are due to PPR, FPR and TPR calculated by Equation 12. These three rates are according to 1% FNR on the validation set.

		No Detector	D1	D2	D3	D4	D5
Efficiency measures	$\mathrm{F_d}$	0	**7.26**	33.39	60.15	29.21	14.83
	PPR%	100	50.61	53.58	50.36	49.88	50.18
	$\mathrm{F_{PPR}}$	670	**346.33**	392.38	397.59	363.40	351.05
	FPR%	100	3.76	8.74	3.52	3.28	2.79
	$\mathrm{F_{FPR}}$	670	**32.47**	91.93	83.73	51.16	33.53
	TPR%	100	97.45	98.42	97.21	96.48	97.57
	$\mathrm{F_{TPR}}$	670	**660.18**	692.82	711.45	675.63	668.57
Performances	Score-overall	23.45	94.49	90.74	94.67	94.84	**95.22**
	Score-clean	26.98	169.23	161.68	169.61	169.98	**170.68**
	Score-noisy	19.90	19.75	19.81	19.73	19.70	**19.78**

CHAPTER 6
CONCLUSIONS AND FUTURE RESEARCH

Recent progress with deep networks has enabled superior speech background noise suppression performance. However, the computational complexity of these models is a limiting factor for many applications, as they can occupy computation resources which may be needed for other processes running in parallel, or they can consume the battery rapidly. Therefore, the efficiency of speech background noise removal is a crucial problem for many applications, where noise removal has to run in parallel with other applications in the device or has to run on mobile devices of which battery capacity is a concern. In this thesis, we have addressed by a detector-driven approach this efficiency issue. In contrast to prior studies, our method is not based on the design of a simpler architecture. Instead, the idea is to activate a given suppression model in the presence of background noise and de-activate it in the absence via a very lightweight noisy-speech detector. Therefore, this approach is beneficial if environmental noise does not constantly occur all the time.

Moreover, independent of other efficiency improvement techniques, the noise suppression network in our detector-driven scheme can be replaced by more efficient networks that might be developed in the future for further efficiency improvements. While the lightweight noisy-speech detector always runs without significant consumption of resources, it can immediately remove the noise by activating a more demanding noise removal model, when necessary, upon detection. This also offers a more practical experience as there would be no need for a manual switch to activate and de-activate a noise removal process.

We carried out our study in three stages. First, we implemented our detector-driven approach via CNNs since they offer highly efficient processing with good performance compared to other deep models like RNNs, as shown in recent studies (Luo & Mesgarani, 2019; Pandey & Wang, 2019a; Sonning et al., 2020; L. Zhang & Wang, 2020). Our background noise removal model is the popular Conv-TasNet model (Luo & Mesgarani, 2019), a modern revision of the completely time-domain network called TasNet by CNNs (Luo & Mesgarani, 2018). We empirically found an optimal Conv-TasNet model regarding the noise suppression performance and test-time efficiency. It obtains 19.9 dB SI-SNR on noisy clips of the VBD dataset (Valentini-Botinhao, 2017) with 640 MFLOPS.

Second, we designed a very simple CNN architecture to detect speech background noise of different types and SNR levels. We evaluated detection performances of different network sizes and resolutions. The test-time complexities vary from 7 MFLOPS to 60 MFLOPS, that is, approximately from 1% to 10% computation load of the most efficient Conv-TasNet. Hence, the detector computations can be assumed negligible. Though we observed some minor differences in the detection performances and FPR-FNR trade-offs, all of them were satisfactory for the task in general.

Third, we applied these detectors together with the optimal Conv-TasNet in our detector-driven scheme by estimating the detection thresholds according to 1% FNR on the validation set. Thanks to successful detection with minor computation overhead, which is about only 2% of the optimal Conv-TasNet by the optimal detector, we obtained about 350 MFLOPS on average over a balanced set of clean and noisy-speech clips. At first glance, this is almost a twofold increase in efficiency since our test dataset contains an equal number of clean and noisy clips. However, gain in efficiency is inversely proportional to background noise occurrence probability. Hence with less noise, we can save more computation resources. Since our detectors can successfully discriminate between noisy and clean-speech, efficiency gain, in the absence of noise, will be very high by almost always de-activating Conv-TasNet. On the other hand, in noisy cases, we will have only a negligible performance drop on average due to a very low noisy-speech miss-rate; because we observed about a 0.1 dB drop in SI-SNR with our best detector due to the miss of noisy clips with a small FNR. We showed that this performance drop happens only when the noise has very low power, rather than happening with strong noise. Hence its perceptual effect will not be serious. Thus, depending on the scene, detector-driven removal can provide dramatic economy in practice by preventing unnecessary processing in the absence of noise.

Moreover, we found that detectors prevent possible audible artifacts due to the processing of clean-speech. This is because Conv-TasNet slightly degrades the quality when applied to clean-speech. Therefore, our detector also provides some small improvement on the quality by not allowing Conv-TasNet to process already clean-speech.

The promising outcome of this thesis motivates future work. For instance, the same approach can be realized on small segments instead of whole clips by proper adjustment of the receptive fields of the models. This modification might bring some further efficiency improvements as well as be suitable for real-time processing.

Another direction of future work could be towards the selection of specialized Conv-TasNet models of varying complexities. For example, easy-to-remove noise types or SNR levels might be handled with simpler models, and more difficult ones might be handled with more complex models. Then, a multi-class classifier, in place of a detector as in this thesis, performs the selection depending on the input. Successful realization of such a fine-tuned scheme would enable further efficiency improvements.

REFERENCES

Almajai, I., & Milner, B. (2008). Using audio-visual features for robust voice activity detection in clean and noisy speech. *2008 16th European Signal Processing Conference*, 1–5.

Ba, J. L., Kiros, J. R., & Hinton, G. E. (2016). *Layer Normalization* (arXiv:1607.06450). arXiv. http://arxiv.org/abs/1607.06450

Boll, S. (1979). Suppression of acoustic noise in speech using spectral subtraction. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, *27*(2), 113–120. https://doi.org/10.1109/TASSP.1979.1163209

Braun, S., Gamper, H., Reddy, C., & Tashev, I. (2021). *Towards Efficient Models for Real-Time Deep Noise Suppression*. 656–660. https://doi.org/10.1109/ICASSP39728.2021.9413580

Chang, J.-H., Kim, N. S., & Mitra, S. K. (2006). Voice activity detection based on multiple statistical models. *IEEE Transactions on Signal Processing*, *54*(6), 1965–1976. https://doi.org/10.1109/TSP.2006.874403

Chang, S.-Y., Li, B., Simko, G., Sainath, T. N., Tripathi, A., van den Oord, A., & Vinyals, O. (2018). Temporal Modeling Using Dilated Convolution and Gating for Voice-Activity-Detection. *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 5549–5553. https://doi.org/10.1109/ICASSP.2018.8461921

Chao, R., Yu, C., Fu, S.-W., Lu, X., & Tsao, Y. (2022). *Perceptual Contrast Stretching on Target Feature for Speech Enhancement* (arXiv:2203.17152). arXiv. https://doi.org/10.48550/arXiv.2203.17152

Chollet, F. (2017). *Xception: Deep Learning with Depthwise Separable Convolutions*. 1800–1807. https://doi.org/10.1109/CVPR.2017.195

Colored, C. B. (2002). *A Multi-Band Spectral Subtraction Method For Enhancing Speech*.

Deng, C., Zhang, Y., Ma, S., Sha, Y., Song, H., & Li, X. (2020). Conv-TasSAN: Separative Adversarial Network Based on Conv-TasNet. *Interspeech 2020*, 2647–2651. https://doi.org/10.21437/Interspeech.2020-2371

El-Fattah, M., Dessouky, M. I., Diab, S., & Abd El-Samie, F. (2008). Speech Enhancement Using an Adaptive Wiener Filtering Approach. *Progress in Electromagnetics Research M*, *4*, 167–184. https://doi.org/10.2528/PIERM08061206

Engel, Y., Mannor, S., & Meir, R. (2004). The kernel recursive least-squares algorithm. *IEEE Transactions on Signal Processing*, *52*(8), 2275–2285. https://doi.org/10.1109/TSP.2004.830985

Fu, S.-W., Liao, C.-F., Tsao, Y., & Lin, S.-D. (2019). Metricgan: Generative adversarial networks based black-box metric scores optimization for speech enhancement. *International Conference on Machine Learning*, 2031–2041.

Fu, S.-W., Yu, C., Hsieh, T.-A., Plantinga, P., Ravanelli, M., Lu, X., & Tsao, Y. (2021). MetricGAN+: An Improved Version of MetricGAN for Speech Enhancement. *Interspeech 2021*, 201–205. https://doi.org/10.21437/Interspeech.2021-599

Greenwood, M., & Kinghorn, A. B. (1999). *SUVING: AUTOMATIC SILENCE /UNVOICED/VOICED CLASSIFICATION OF SPEECH*.

Heitkaemper, J., Jakobeit, D., Boeddeker, C., Drude, L., & Haeb-Umbach, R. (2020). Demystifying TasNet: A dissecting approach. *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 6359–6363.

Hu, Y., Liu, Y., Lv, S., Xing, M., Zhang, S., Fu, Y., Wu, J., Zhang, B., & Xie, L. (2020). DCCRN: Deep Complex Convolution Recurrent Network for Phase-Aware Speech Enhancement. *Interspeech 2020*, 2472–2476. https://doi.org/10.21437/Interspeech.2020-2537

Hu, Y., & Loizou, P. C. (2002). A subspace approach for enhancing speech corrupted by colored noise. *IEEE International Conference on Acoustics Speech and Signal Processing*, I-573-I–576. https://doi.org/10.1109/ICASSP.2002.5743782

Hummersone, C., Stokes, T., & Brookes, T. (2014). *On the Ideal Ratio Mask as the Goal of Computational Auditory Scene Analysis* (pp. 349–368). https://doi.org/10.1007/978-3-642-55016-4_12

Hwang, I., Park, H.-M., & Chang, J.-H. (2015). Ensemble of Deep Neural Networks Using Acoustic Environment Classification for Statistical Model-Based Voice ActivityDetection. *Computer Speech & Language*, *38*. https://doi.org/10.1016/j.csl.2015.11.003

Kadıoğlu, B., Horgan, M., Liu, X., Pons, J., Darcy, D., & Kumar, V. (2020). An Empirical Study of Conv-Tasnet. *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 7264–7268. https://doi.org/10.1109/ICASSP40776.2020.9054721

Koizumi, Y., Karita, S., Wisdom, S., Erdogan, H., Hershey, J. R., Jones, L., & Bacchiani, M. (2021). DF-Conformer: Integrated architecture of Conv-TasNet and Conformer using linear complexity self-attention for speech enhancement. *2021 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, 161–165.

Koyama, Y., Vuong, T., Uhlich, S., & Raj, B. (2020). *Exploring the Best Loss Function for DNN-Based Low-latency Speech Enhancement with Temporal Convolutional Networks* (arXiv:2005.11611). arXiv. http://arxiv.org/abs/2005.11611

Le Roux, J., Wisdom, S., Erdogan, H., & Hershey, J. R. (2019). SDR–half-baked or well done? *ICASSP 2019-2019 IEEE International Conference on Acoustics,* *Speech and Signal Processing (ICASSP)*, 626–630.

lu, X., Tsao, Y., Matsuda, S., & Hori, C. (2013). Speech enhancement based on deep denoising Auto-Encoder. *Proc. Interspeech*, 436–440.

Luo, Y., & Mesgarani, N. (2018). TaSNet: Time-Domain Audio Separation Network for Real-Time, Single-Channel Speech Separation. *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 696–700. https://doi.org/10.1109/ICASSP.2018.8462116

Luo, Y., & Mesgarani, N. (2019). Conv-TasNet: Surpassing Ideal Time–Frequency Magnitude Masking for Speech Separation. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, *27*(8), 1256–1266. https://doi.org/10.1109/taslp.2019.2915167

M., T., Adeel, A., & Hussain, A. (2018). A Survey on Techniques for Enhancing Speech. *International Journal of Computer Applications*, *179*, 1–14. https://doi.org/10.5120/ijca2018916290

Ma, C., Li, D., & Jia, X. (2020). Optimal scale-invariant signal-to-noise ratio and curriculum learning for monaural multi-speaker speech separation in noisy environment. *New Zealand*, 5.

Noise Cancelling App & Echo Reduction Software | Krisp. (n.d.). Retrieved July 24, 2022, from https://www.krisp.ai/

NVIDIA: World Leader in Artificial Intelligence Computing. (n.d.). NVIDIA. Retrieved July 24, 2022, from https://www.nvidia.com/en-us/

Pandey, A., & Wang, D. (2019a). TCNN: Temporal Convolutional Neural Network for Real-time Speech Enhancement in the Time Domain. *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 6875–6879. https://doi.org/10.1109/ICASSP.2019.8683634

Pandey, A., & Wang, D. (2019b). A New Framework for CNN-Based Speech Enhancement in the Time Domain. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, *27*(7), 1179–1188. https://doi.org/10.1109/TASLP.2019.2913512

Park, S. R., & Lee, J. W. (2017). A Fully Convolutional Neural Network for Speech Enhancement. *Proc. Interspeech 2017*, 1993–1997. https://doi.org/10.21437/Interspeech.2017-1465

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

Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., Lin, Z., Desmaison, A., Antiga, L., & Lerer, A. (2017). Automatic differentiation in PyTorch. *NIPS-W*.

Pearce, D., & Hirsch, H.-G. (2000). The AURORA experimental framework for the performance evaluations of speech recognition systems under noisy condition.

Proc. ICSLP, *4*, 29–32.

Rakesh, P., & Kumar, T. K. (2015). A Novel RLS Based Adaptive Filtering Method for Speech Enhancement. *International Journal of Electronics and Communication Engineering*, *9*(2), 6.

Ramirez, J., Segura, J. C., Benitez, C., Garcia, L., & Rubio, A. (2005). Statistical voice activity detection using a multiple observation likelihood ratio test. *IEEE Signal Processing Letters*, *12*(10), 689–692. https://doi.org/10.1109/LSP.2005.855551

Ramírez, J., Segura, J. C., Benítez, C., de la Torre, Á., & Rubio, A. (2004). Efficient voice activity detection algorithms using long-term speech information. *Speech Communication*, *42*(3), 271–287. https://doi.org/10.1016/j.specom.2003.10.002

Rethage, D., Pons, J., & Serra, X. (2018). A Wavenet for Speech Denoising. *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 5069–5073. https://doi.org/10.1109/ICASSP.2018.8462417

Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., & Chen, L.-C. (2018). MobileNetV2: Inverted Residuals and Linear Bottlenecks. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 4510–4520. https://doi.org/10.1109/CVPR.2018.00474

Sehgal, A., & Kehtarnavaz, N. (2018). A Convolutional Neural Network Smartphone App for Real-Time Voice Activity Detection. *IEEE Access*, *6*, 9017–9026. https://doi.org/10.1109/ACCESS.2018.2800728

Sohn, J., Kim, N. S., & Sung, W. (1999). A statistical model-based voice activity detection. *IEEE Signal Processing Letters*, *6*(1), 1–3. https://doi.org/10.1109/97.736233

Sonning, S., Schuldt, C., Erdogan, H., & Wisdom, S. (2020). Performance Study of a Convolutional Time-Domain Audio Separation Network for Real-Time Speech Denoising. *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 831–835. https://doi.org/10.1109/ICASSP40776.2020.9053846

Stearns, S. D. (1985). *Fundamentals of Adaptive Signal Processing*.

Subakan, C., Ravanelli, M., Cornell, S., Bronzi, M., & Zhong, J. (2021). Attention Is All You Need In Speech Separation. *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 21–25. https://doi.org/10.1109/ICASSP39728.2021.9413901

Subakan, C., Ravanelli, M., Cornell, S., Grondin, F., & Bronzi, M. (2022). *On Using Transformers for Speech-Separation* (arXiv:2202.02884). arXiv. http://arxiv.org/abs/2202.02884

Sun, L., Du, J., Dai, L.-R., & Lee, C.-H. (2017). Multiple-target deep learning for LSTM-RNN based speech enhancement. *2017 Hands-Free Speech*

Communications and Microphone Arrays (HSCMA), 136–140. https://doi.org/10.1109/HSCMA.2017.7895577

Suter, A. H. (1991). Noise and its effects. *Administrative Conference of the United States*, 1–47.

Tanyer, S. G., & Ozer, H. (2000). Voice activity detection in nonstationary noise. *IEEE Transactions on Speech and Audio Processing*, *8*(4), 478–482. https://doi.org/10.1109/89.848229

Thiemann, J., Ito, N., & Vincent, E. (2013a). The Diverse Environments Multi-channel Acoustic Noise Database (DEMAND): A database of multichannel environmental noise recordings. *Proceedings of Meetings on Acoustics*, *19*(1), 035081. https://doi.org/10.1121/1.4799597

Thiemann, J., Ito, N., & Vincent, E. (2013b). *DEMAND: A collection of multi-channel recordings of acoustic noise in diverse environments* [Data set]. Zenodo. https://doi.org/10.5281/zenodo.1227121

Ting, P.-J., Ruan, S.-J., & Li, L. P.-H. (2021). Environmental Noise Classification with Inception-Dense Blocks for Hearing Aids. *Sensors*, *21*(16), 5406. https://doi.org/10.3390/s21165406

Tucker, R. (1992). Voice activity detection using a periodicity measure. *IEE Proceedings I (Communications, Speech and Vision)*, *139*(4), 377–380. https://doi.org/10.1049/ip-i-2.1992.0052

Udrea, R. M., & Ciochina, S. (2003). Speech enhancement using spectral over-subtraction and residual noise reduction. *International Symposium on Signals, Circuits and Systems, 2003. SCS 2003*, *1*, 165–168 vol.1. https://doi.org/10.1109/SCS.2003.1226974

Valentini-Botinhao, C. (2017). *Noisy speech database for training speech enhancement algorithms and TTS models*.

Veaux, C., Yamagishi, J., & King, S. (2013). *The voice bank corpus: Design, collection and data analysis of a large regional accent speech database*. 1–4. https://doi.org/10.1109/ICSDA.2013.6709856

Weninger, F., Erdogan, H., Watanabe, S., Vincent, E., Le Roux, J., Hershey, J. R., & Schuller, B. (2015). Speech Enhancement with LSTM Recurrent Neural Networks and its Application to Noise-Robust ASR. In E. Vincent, A. Yeredor, Z. Koldovský, & P. Tichavský (Eds.), *Latent Variable Analysis and Signal Separation* (pp. 91–99). Springer International Publishing. https://doi.org/10.1007/978-3-319-22482-4_11

Wiener, N., Wiener, N., Mathematician, C., Wiener, N., Wiener, N., & Mathématicien, C. (1949). *Extrapolation, interpolation, and smoothing of stationary time series: With engineering applications* (Vol. 113). MIT press Cambridge, MA.

Xu, Y., Du, J., Dai, L.-R., & Lee, C.-H. (2014). An Experimental Study on Speech Enhancement Based on Deep Neural Networks. *IEEE Signal Processing*

Letters, *21*(1), 65–68. https://doi.org/10.1109/LSP.2013.2291240

Xu, Y., Du, J., Dai, L.-R., & Lee, C.-H. (2015). A Regression Approach to Speech Enhancement Based on Deep Neural Networks. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, *23*(1), 7–19. https://doi.org/10.1109/TASLP.2014.2364452

Yang, G.-P., Tuan, C.-I., Lee, H.-Y., & Lee, L. (2019). Improved Speech Separation with Time-and-Frequency Cross-Domain Joint Embedding and Clustering. *Interspeech 2019*, 1363–1367. https://doi.org/10.21437/Interspeech.2019-2181

Yang, L.-P., & Fu, Q.-J. (2005). Spectral subtraction-based speech enhancement for cochlear implant patients in background noise. *The Journal of the Acoustical Society of America*, *117*(3), 1001–1004. https://doi.org/10.1121/1.1852873

Yu, C., Fu, S.-W., Hsieh, T.-A., Tsao, Y., & Ravanelli, M. (2021). OSSEM: one-shot speaker adaptive speech enhancement using meta learning. *ArXiv, abs/2111.05703*.

Zhang, L., & Wang, M. (2020). Multi-Scale TCN: Exploring Better Temporal DNN Model for Causal Speech Enhancement. *Interspeech*, 2672–2676.

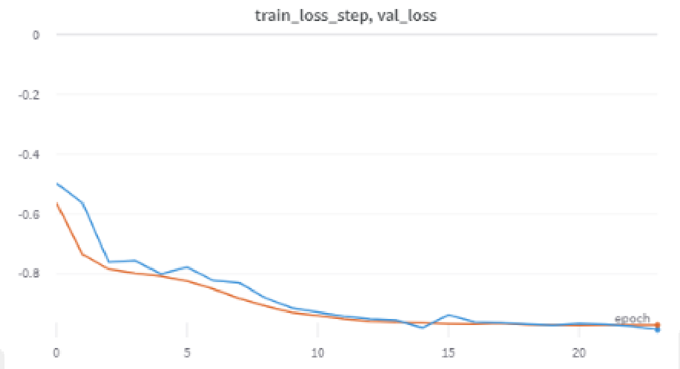
Zhang, Y.-D., Pan, C., Sun, J., & Tang, C. (2018). Multiple sclerosis identification by convolutional neural network with dropout and parametric ReLU. *Journal of Computational Science*, *28*, 1–10. https://doi.org/10.1016/j.jocs.2018.07.003

Zhao, H., Zarar, S., Tashev, I., & Lee, C.-H. (2018). Convolutional-Recurrent Neural Networks for Speech Enhancement. *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2401–2405. https://doi.org/10.1109/ICASSP.2018.8462155

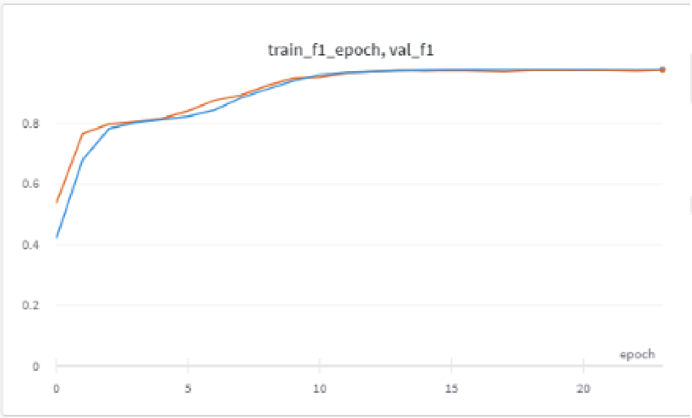
APPENDIX

Noisy-speech detector Network Evaluation

Model 1



blue: train_loss_step, orange: val_loss



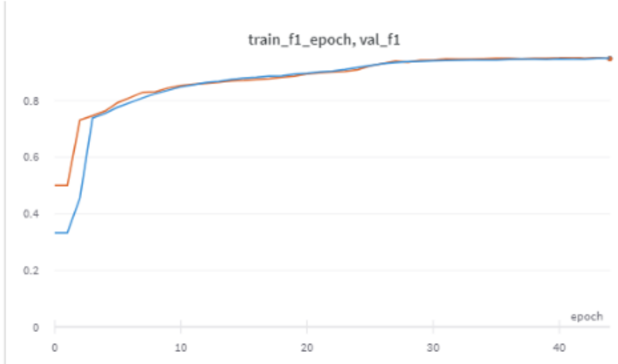
blue: train_f1_epoch, orange: val_f1

Confusion Matrices at FNR 1 % (threshold = 0.991)

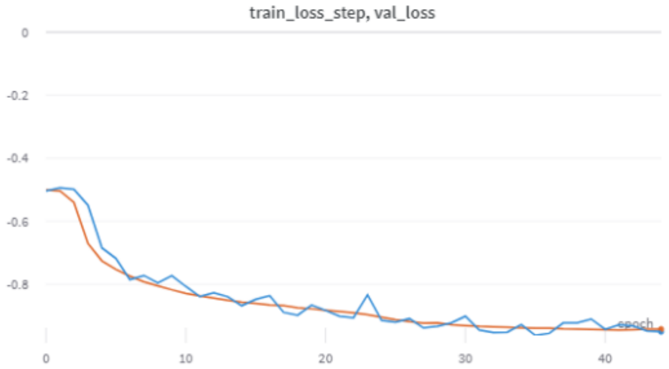
P: noisy, N: not noisy, rows actual, columns predict

Actual	Predicted->	P	N
P		803	21
N		31	793

Model 2



blue: train_f1_epoch, orange: val_f1

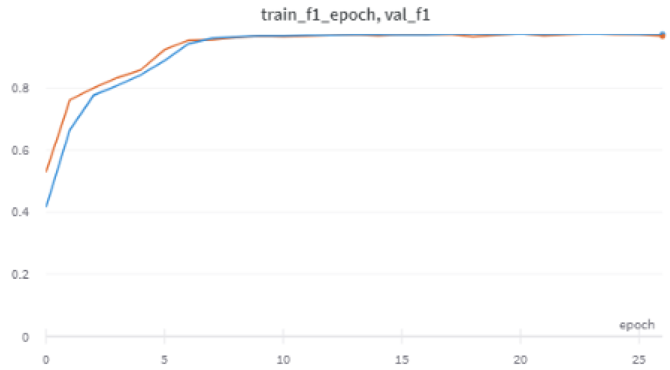


blue: train_loss_step, orange: val_loss

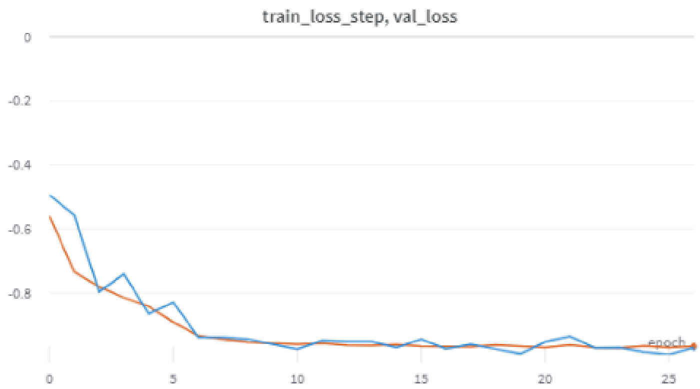
Confusion Matrices at FNR 1 % (threshold = 0.988)

#	P	N
P	811	13
N	72	752

Model 3



blue: train_f1_epoch, orange: val_f1

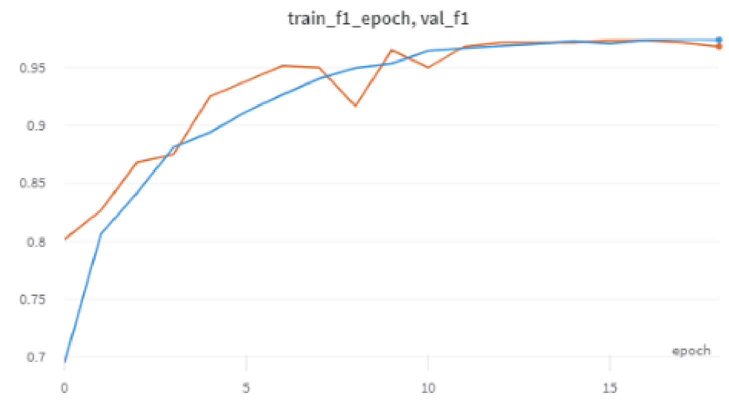


blue: train_loss_step, orange: val_loss

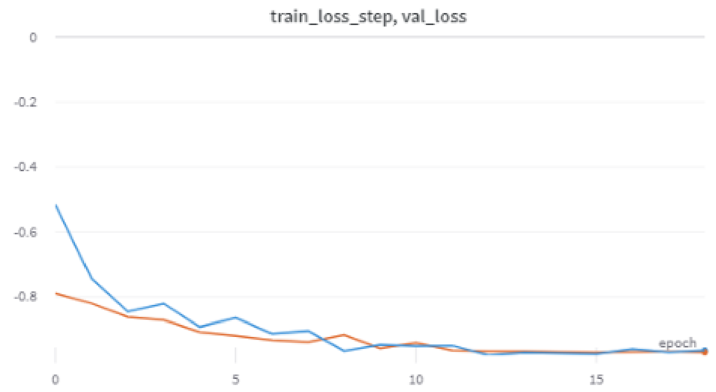
Confusion Matrices at FNR 1 % (threshold = 0.987)

#	P	N
P	801	23
N	29	795

Model 4



blue: train_f1_epoch, orange: val_f1

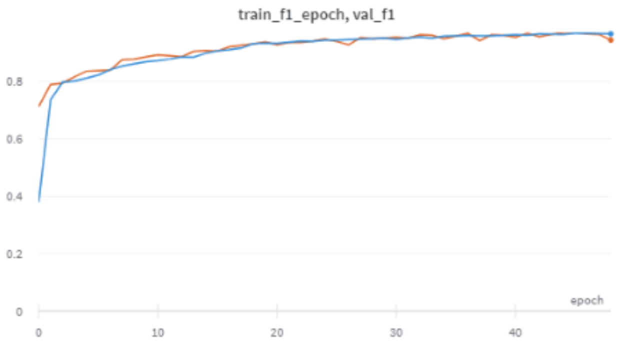


blue: train_loss_step, orange: val_loss

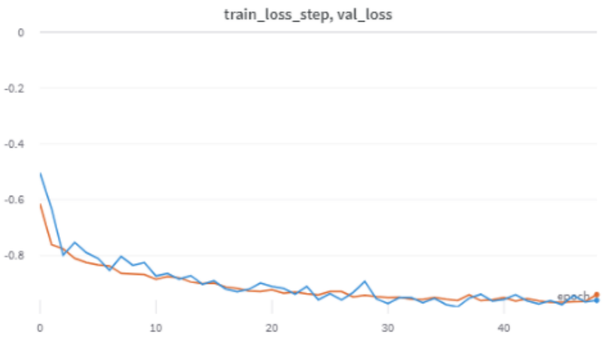
Confusion Matrices at FNR 1 % (threshold = 0.993)

#	P	N
P	795	29
N	27	797

Model 5



blue: train_f1_epoch, orange: val_f1

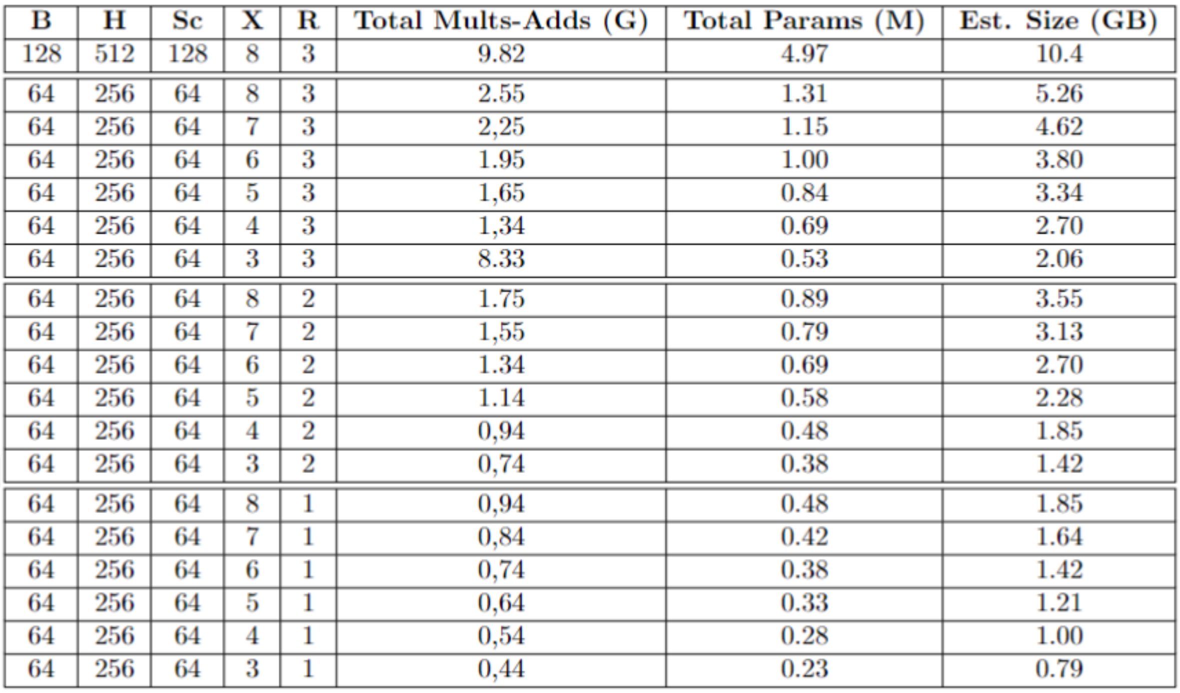


blue: train_loss_step, orange: val_loss

Confusion Matrices at FNR 1 % (threshold = 0.944)

#	P	N
P	804	20
N	23	801

Conv - TasNet



B	H	Sc	X	R	Total Mults-Adds (G)	Total Params (M)	Est. Size (GB)
128	512	128	8	3	9.82	4.97	10.4
64	256	64	8	3	2.55	1.31	5.26
64	256	64	7	3	2,25	1.15	4.62
64	256	64	6	3	1.95	1.00	3.80
64	256	64	5	3	1,65	0.84	3.34
64	256	64	4	3	1,34	0.69	2.70
64	256	64	3	3	8.33	0.53	2.06
64	256	64	8	2	1.75	0.89	3.55
64	256	64	7	2	1,55	0.79	3.13
64	256	64	6	2	1.34	0.69	2.70
64	256	64	5	2	1.14	0.58	2.28
64	256	64	4	2	0,94	0.48	1.85
64	256	64	3	2	0,74	0.38	1.42
64	256	64	8	1	0,94	0.48	1.85
64	256	64	7	1	0,84	0.42	1.64
64	256	64	6	1	0,74	0.38	1.42
64	256	64	5	1	0,64	0.33	1.21
64	256	64	4	1	0,54	0.28	1.00
64	256	64	3	1	0,44	0.23	0.79