UNIVERSITY OF SOUTHAMPTON
Faculty of Engineering and the Environment
Institute of Sound and Vibration

Cochlea Modelling and its Application to Speech Processing

by

Shuokai Pan

A thesis submitted for the degree of
Doctor of Philosophy

June 2018

UNIVERSITY OF SOUTHAMPTON

ABSTRACT

FACULTY OF ENGINEERING AND THE ENVIRONMENT
INSTITUTE OF SOUND AND VIBRATION

Doctor of Philosophy

COCHLEA MODELLING AND ITS APPLICATION TO SPEECH PROCESSING

by Shuokai Pan

Models of the cochlea provide a valuable tool for both better understanding its mechanics and also as an inspiration for many speech processing algorithms. Realistic modelling of the cochlea can be computationally demanding, however, which limits its applicability in signal processing applications. To mitigate this issue, an efficient numerical method has been proposed for performing time domain simulations, based on a nonlinear state space formulation [1]. This model has then been contrasted with another type of cochlear model, that is established from a cascade of digital filters. A comparison of the responses from these two models has been conducted, in terms of their realism in simulating the measured nonlinear cochlear response to single tones and pairs of tones. Guided by these results, the filter cascade model is chosen for subsequent signal processing applications because it is significantly more efficient than the state space model, while still producing realistic responses.

Using this nonlinear filter cascade model as a front-end, two speech processing tasks have been investigated: voice activity detection and supervised speech separation. Both tasks are tackled within a machine learning framework, in which a neural network is trained to reproduce target outputs. The results are compared with those using a number of other simpler auditory-inspired analysis methods. Simulation results show that although the nonlinear filter cascade model can be more effective in many testing scenarios, its relative advantage against other analysis methods is small. The incorporation of temporal context information and network structure engineering are found to be more important in improving the performance of these tasks. Once a suitable context expansion strategy has been selected, the difference between various front-end processing methods considered is marginal.

Contents

List of Figures

List of Tables

Declaration of Authorship

I, Shuokai Pan, declare that this thesis entitled *Cochlea Modelling and its Application to Speech Processing* and the work presented in it are my own and has been generated by me as the result of my own original research.

I confirm that:

1. This work was done wholly or mainly while in candidature for a research degree at this University;
2. Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated;
3. Where I have consulted the published work of others, this is always clearly attributed;
4. Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work;
5. I have acknowledged all main sources of help;
6. Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself;

Signed:

Date:

Acknowledgements

I would like to express my sincere gratitude to my supervisor Professor Stephen Elliott, without whom I would never be able to finish this challenging but rewarding jounery of a PhD. His wisdom and passion for scientific research support me immensely and will definitely continue to benefit me for the rest of my life. I would also like to thank Professor Paul White for answering many of my questions and providing additional resources that facilitate my research, and Dr Ed Craney from Cirrus Logic for visiting me biannually and guiding my research with industry expertise. Another thanks goes to Professor Stefan Bleeck who allows me to join their weekly discussion group hosted in the Hearing and Balance Centre at the ISVR, which greatly broadens my horizons in how auditory knowledge can be applied in various aspects of acoustic signal processing.

Many mang thanks go to my family: my parents for their continuous and selfless support, and my wife, Anna Deng, for accompanying me and always encouraging me to finish my study. This thesis is devoted to all of you.

I would like to thank all of the friends I have met at the ISVR, you guys have made my PhD a much more enjoyable experience.

Finally, I would like to thank sincerely,both EPSRC and Cirrus Logic International Semiconductor, for providing the financial aid that helps me finish my study.

Abbreviations

2TS	Two-tone Suppression
AGC	Automatic Gain Control
AMS	Amplitude Modulation Spectrum
AN	Auditory Nerve
AUC	Area Under the Curve
BM	Basilar Membrane
CASA	Computational Auditory Scene Analysis
CARFAC	Cascade of Asymmetric Resonators with Fast-Acting Compression
CF	Characteristic Frequency
cIRM	Complex Ideal Ratio Mask
CNN	Convolutional Neural Network
CNN-MR	Multi-resolution Convolutional Neural Network
CNN-SR	Single-resolution Convolutional Neural Network
CP	Characteristic frequency
DFT	Discrete Fourier Transform
DNN	Deep Feedforward Neural Network
FA	False Alarm
IBM	Ideal Binary Mask
IHC	Inner Hair Cells
IRM	Ideal Ratio Mask
LC	Local Criterion
LPC	Linear Predictive Coding
MFCC	Mel Frequency Cepstral Coefficient
MLP	Multilayer Perceptron

MMSE	Minimum Mean Square Error
MRCG	Multi-Resolution Cochleagram
MSE	Mean Square Error
MVN	Mean Variance Normalization
MWLCG	Multi-Window-Length Cochleagram
MWLFC	Multi-Window-Length Filter Cascade
MWLLM	Multi-Window-Length LogMel Spectrogram
OC	Organ of Corti
OHC	Outer Hair Cells
PESQ	Perceptual Evaluation of Speech Quality
PLP	Perceptual Linear Prediction
PNCC	Power-Normalized Cepstral Coefficients
ReLU	Rectified Linear Unit
ROC	Receiver Operating Characteristic Curve
ROG	Rate of Growth
ROS	Rate of Suppression
SM	Scala Media
SNR	Signal-to-Noise Ratio
SPL	Sound Pressure Level
ST	Scala Tympani
STFT	Short Time Fourier Transform
STOI	Short-Time Objective Intelligibility
SV	Scala Vestibuli
TL	Transmission-Line Model
TM	Tectorial Membrane
VAD	Voice Activity Detection

Chapter 1

Introduction

1.1 Motivation

Speech is a natural medium for human communications. Robust and efficient processing of speech has thus become a crucial component of many modern technologies, including telecommunication, automatic speech transcription and translation, speaker verification, and hearing aids. Driven by the huge demand of these technologies in different scenarios, many methods have been proposed for different applications. The continuous advancement of these algorithms has greatly improved their performance and robustness, and contributed to new generations of speech processing applications, such as the increasingly popular voice-controlled digital assistants, including Apple Siri, Microsoft Cortana, Google Home and Amazon Echo. These smart systems are able to listen and respond to our speech commands, for instance to retrieve answers from the internet for spoken queries and control smart home gadgets like light switches, thermostats and TVs, usually after the detection of some specific keyword phrase(s) such as "Hey Siri" and "Hi Alexa". Despite the great success of these applications, a common challenge that faces them is that, in the wide range of acoustic environments where they are employed, speech signals are often distorted by various interferences such as channel

effects, background noise, music and reverberation. These can significantly degrade their effectiveness and hence limit their widespread usage in practice, even though in well-controlled clean speech conditions, they often perform very well. However, human perception of speech is very robust against various forms of disturbances. Thus the motivation of this work is to explore the benefits of using auditory principles in developing environmentally robust speech processing algorithms that could meet the increasing demand of practical applications, such as voice controlled smart devices.

Because the complete auditory pathway is very complicated and contains a number of stages, many studies only include a small subset of auditory functionalities and combine them with traditional signal processing techniques, while those that integrate both low-level and more central high-level auditory principles are commonly known as computational auditory scene analysis or CASA [2]. In this work, we investigate the use of characteristics of the auditory peripheral, particularly the nonlinear cochlea mechanics, for speech processing, because (1) it is the first advanced signal processing stage in the auditory pathway and is much better understood than central auditory functions; (2) it is usually assumed that the cochlear contributions to auditory processing, such as dynamic range compression, frequency selectivity and sensitivity is so dominant that it largely accounts for the properties of the entire auditory system [3]; (3) many previous studies have indeed shown the advantages of incorporating knowledge of cochlear mechanisms [4, 5, 6, 7, 8].

The characteristics of the cochlea are integrated through a mathematical model, which is an important area of auditory research in itself. Because it not only provides a valuable tool for better understanding of its mechanisms, such as otoacoustic emissions, but also allows predictions of experiments that are difficult to perform due to technical limitations. It is worth noting that there are mainly two types of processing present in the cochlea: (1) the transformation of acoustic pressure variations into mechanical vibrations along the cochlear partition through fluid coupling, and (2) the transduction of this mechanical vibration to electrical impulses that are transmitted to the brain by the auditory nerves, thus making

the physical sound perceptible to human beings. These two processes are usually modelled separately and the complete cochlear functionality is simulated by connecting them together. In this work, we only focus on models of the mechanical aspect of the cochlea that are often simply referred to as cochlear models.

Since cochlear models have been proposed in a large number of ways, the first research direction of this work is the choice of a suitable model for subsequent speech processing. Specifically, two types of cochlear models are investigated. The first one is the transmission-line type of model that tries to realistically simulate the travelling wave inside the cochlea using differential equations. This model is based on that originally proposed by Neely and Kim [9], but is reformulated into the state space form by Elliott et al. [1] and retuned by Ku [10] and Young [11] to match the properties of a human cochlea. The second model consists of a cascade of auditory filters, called the cascade of asymmetric resonators with fast-acting compression or CARFAC [12], that is inspired by the cochlea travelling wave. Although more physiologically plausible than using parallel filterbank approaches, it still mainly aims to reproduce the overall experimental measurements of cochlear responses, without special considerations of the underlying biophysical mechanisms. The first model is more realistic, but it is also much more computationally demanding than the second one, especially for time domain simulations. This is also the primary reason why this type of model is seldom adopted in practical applications. Therefore, the first objective of this research direction is to develop an efficient numerical method for performing time domain simulations with the state-space model, in order to reduce the computational burden. This will benefit its applicability to both cochlear modelling and signal processing tasks. Next, a systematic comparison between the responses of these two models is performed in order to assess their capabilities in reproducing cochlear nonlinearities, because it is these nonlinearities that we are most interested in applying and exploring their potential benefits to speech processing, and hence serves as a verification step. Only single-tone and two-tone responses of these models are compared with experimental measurements, because these responses reveal key aspects of the nonlinear cochlear mechanics, that we assume are promising for speech processing

applications. These aspects include active amplification, dynamic compression, and mutual suppression, which is attractive for simulating masking effects.

The next research direction of this work is to investigate the potential benefits of integrating properties of the cochlea mechanics for speech processing applications. There is a long history of this type of work in the speech processing field, most notably with promising results in speech recognition [4], speech enhancement or separation [13, 5], speech detection [6] and audio coding [7, 8]. In this work, we consider two tasks: voice activity detection (VAD) and speech enhancement or separation. In many previous studies [5, 6] of these two areas, the input acoustic signals are firstly decomposed into sub-bands using a simple cochlea-like filterbank (mostly parallel filterbanks). Popular examples include the Mel-scale [14], the Bark-scale [15] and the gammatone [16] filterbanks. Although they provide representations of speech signals that are similar to those produced by a biological cochlea, they still only offer a very crude simulation of nonlinear cochlear mechanics. A more sophisticated cochlear model, however, has a more realistic representation of cochlear nonlinearities and is utilised in this work to investigate whether advanced modelling of cochlear mechanics can be more effective than other simpler cochlea-like models for the two speech processing tasks considered.

Since 2006, one class of machine learning called deep learning [17] has began to regain its popularity, and has now become the core technology behind a wide range of signal and information processing applications. It utilizes deep architectures that contain a hierarchy of nonlinear processing layers, which enable a machine to learn complex concepts in the real world environments from a nested set of simpler concepts [18]. This has been proven to be an extremely powerful learning paradigm, because algorithms based on it have been shown to significantly outperform traditional methods in a large number of machine learning problems, including image classification [19], speech recognition [20], voice activity detection [6] and speech separation [5]. For these reasons, the investigation of cochlear modelling for the two speech processing tasks mentioned above is carried out within the deep learning framework.

1.2 Thesis Contributions

The primary contributions of this work is summarised in the following,

- Development of an efficient numerical method, that substantially reduces the computational complexity of time domain simulations of a nonlinear, one-dimensional, state-space, transmission-line cochlear model.
- Establishment of the theoretical relationship between the transmission-line and filter cascade model of the cochlea and conducting a systematic comparison of their linear and nonlinear responses to single and two tones.
- Investigation of the advantages of using a nonlinear filter cascade cochlear model as a front-end processor in a neural network based voice activity detection task. Demonstration that the filter cascade model does not provide significant benefits, compared to other simpler auditory filterbank features, at least in the machine learning paradigm considered. However, the effective incorporation of context information is proved to be more important for performance improvement. Comparison with another two state-of-the-art VAD algorithms shows that proposed methods exhibit a significantly higher level of robustness against additive background noises.
- Investigation of the advantages of using the CARFAC filter cascade model for supervised speech separation. in most cases, filter cascade based systems show the highest level of performance compared to standard auditory filterbank based ones, but the advantage is rather marginal. The same set of temporal context expansion techniques firstly investigated in the VAD task has also been applied to supervised speech separation, but the relative advantage of these techniques is found to be rather different. The proposed methods also show substantially better performance than another three previous algorithms in terms of two objective metrics.

1.3 Publications

Some of the initial outcomes of this work have been published as a Journal paper and a conference proceeding:

- Shuokai Pan et al. "Efficient time-domain simulation of nonlinear, state-space, transmission-line models of the cochlea." In: *The Journal of the Acoustical Society of America* 137.6 (2015), pp. 3559-3562.
- Pan, Shuokai and Elliott, Stephen J and Vignali, Dario. "Comparison of the nonlinear responses of a transmission-line and a filter cascade model of the human cochlea." In: *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)* (2015), pp.1-5.

1.4 Thesis Outline

The remainder of this thesis is organized as follows. Chapter 2 presents the foundation knowledge and literature review for the work in this thesis. This includes an introduction to the physiology of the human auditory periphery, particularly the cochlea, and deep learning basics that are essential for subsequent speech processing tasks.

Chapter 3 presents a thorough investigation and comparison of two types of cochlear models, i.e., one transmission-line model and the CARFAC model, in terms of their realism in simulating nonlinear cochlear mechanics. The formal connections between the linear versions of these two models is firstly explained, followed by the comparison of the responses of the complete nonlinear models to simple stimuli including single tones and two tones.

Guided by the results from Chapter 3, the CARFAC model is chosen, instead of the transmission-line model, for neural network based voice activity detection study in Chapter 4. It is compared with a number of simpler auditory-inspired filterbank features and another two state-of-the-art algorithms in a wide range

of noisy test scenarios. Multiple types of context extension strategies have also been investigated and the most effective one for each feature type has also been identified.

Chapter 5 presents results on using the CARFAC model for supervised speech separation. A comparison of different combinations of feature type and network structure is carried out. Following the simulations presented in Chapter 4, the relative advantage of various context expansion schemes is determined as well, which is different from those observed in VAD task. A comparison of proposed methods with another supervised learning based method and two classical speech enhancement methods is also performed.

Chapter 6 concludes this thesis and proposes suggestions for future work.

Chapter 2

Background and Literature Review

In this chapter, we give a brief overview of the background to this thesis, including an introduction to the physiology and mechanics of the mammalian cochlea, and an overview of recent developments in deep learning. The following chapters build on this knowledge and elaborate on the problem of cochlear modelling and its application to speech processing tasks.

2.1 The Cochlea

The human auditory system is a marvellous achievement of biological evolution, capable of detecting and analysing sounds over a wide range of frequencies and intensities. Humans can hear sounds over the range of approximately ten octaves (20 Hz to 20 kHz) in frequency and 140 dB (0 to 140 dB) in sound pressure level (SPL) [21], while possessing great sensitivity and frequency selectivity. Extensive research has been carried out to understand and model various stages of this powerful but complicated biological system. The cochlea is the first major computational element of the human auditory pathway. It is a fluid-filled organ located in the inner ear, as shown in the illustration of the human ear in Fig. 2.1.

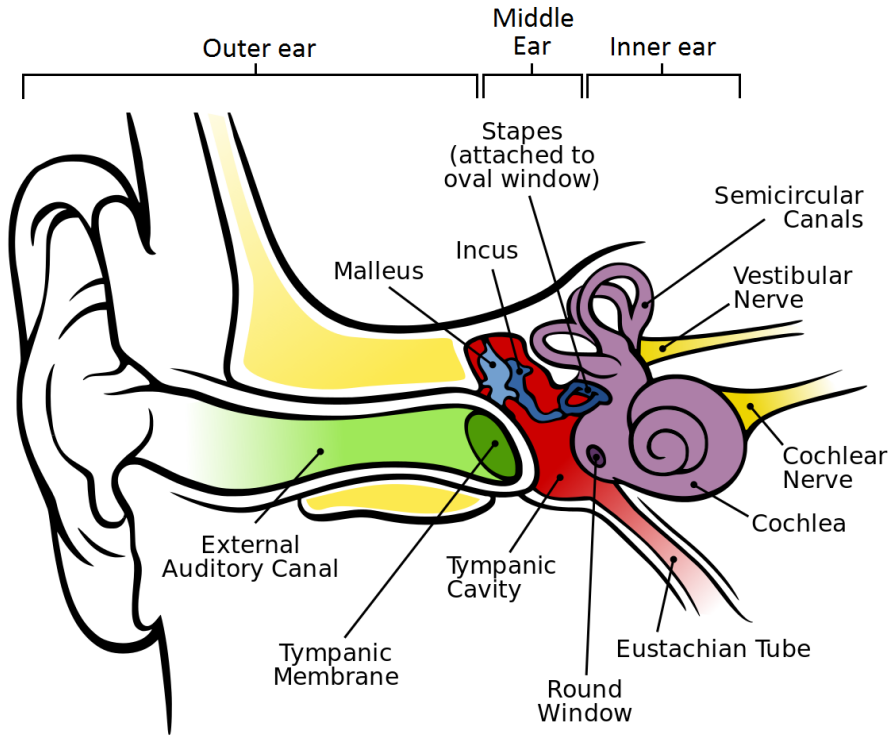


FIGURE 2.1: The anatomy of the human ear. The cochlea is located in the inner ear. Picture adapted from [22] with permission.

As can be seen in Fig. 2.1, the human ear, also called the auditory periphery, consists of three stages: the outer ear, the middle ear and the inner ear. The outer ear includes the pinna, the ear canal and the tympanic membrane or eardrum. It allows efficient capture of sound impinging on the head and its directional sensitivity provides a cue for sound localization. The middle ear is an air-filled space containing a set of three small bones, the Malleus, the Incus and the Stapes, collectively known as the ossicles, the first of which is connected to the ear drum and the last of which is connected to the cochlea. They act primarily as an impedance matching mechanism between the air in the outer ear (low impedance) and the fluid in the cochlea (high impedance), facilitating energy transfer into the cochlea. Overall, the effects of middle ear can be well modelled by a linear mass-spring-damper system for stimulus frequencies below 10 kHz [23] and stimulus levels below 140 dB SPL [24]. The cochlea serves as a bridge between the physical pressure variations and the perception of sound in our brains. It achieves this mainly through two steps: (1) the interaction between the cochlear fluid and partition maps the input stimuli along its longitudinal direction according to its spectral components. This

constitutes the so-called cochlear mechanics; (2) the conversion of the mechanical energy into electrical impulses, through specialized sensory cells, which are sent to the brain via the auditory nerves.

2.1.1 Cochlea Anatomy

The cochlea is one of the most delicate and complicated parts of the human body. It has a snail-shaped structure, with about 2.5 turns and a typical length of 35 mm. The cochlear duct is divided into three compartments: the scala vestibuli (SV), the scala media (SM) and the scala tympani (ST). The Reissner's membrane separates SV from SM, while the basilar membrane (BM) separates ST from SM, as shown in Fig. 2.2 (a). Since the Reissner's membrane is very thin and possesses a similar density to that of cochlear fluid, it is generally assumes to have no influence to cochlear mechanics [25], but only to separate the fluids in the SV and the SM. The SV and ST are filled with perilymph fluid and connected at the apex of the cochlea, called helicotrema, to allow fluid flow between them. The SM is completely separated from the other two chambers, and filled with endolymph fluid, that is generated in the stria vascularis. It houses an important sensory organ that is responsible for the cochlear transduction process, called the organ of Corti (OC), which is distributed along the length of the BM. The detailed structure of the OC is shown in Fig. 2.2 (b). As can be seen, it consists of a range of supporting cells and two important kinds of sensory cells, called the inner and outer hair cells, because of their relative positions. There are one row of about 3,500 inner hair cells (IHC) and three rows of about 12,000 outer hair cells (OHC) [26], and they possess hair-like structure at their apical end called stereocilia. The hair bundles of the OHCs are attached at their top to the lower surface of the tectorial membrane (TM). Such basic structure of the OC is similar for most mammals.

These sensory cells of the cochlea interact with the nervous system through the auditory nerve, which consists of approximately 30,000 neurons in human [27]. Most of these neurons are afferents, with their cell bodies in the spiral ganglion, that carry information from the hair cells to the brain. The remaining small

amount are the efferents, which are the axons of neurons in the brainstem that descend into the cochlea to allow central control of the cochlear transduction process. The afferents also include two types, called type I and type II. Type I neurons make up about 95% of the population, with one end innervates the IHCs directly and the other end projects into the core areas of the cochlear nucleus. The remaining around 5% afferents, called type II, innervate the OHCs instead. The efferent neurons, however, have their cell bodies in an auditory structure in the brainstem called the superior olivary complex and are thus called the olivocochlear bundle. There are two groups of olivocochlear bundle neurons as well. The so-called lateral efferents travel mainly to the ipsilateral cochlea and make synaptic connections on the dendrites of type I afferents under the IHCs, while the medial efferents, travel to both the ipsilateral and contralateral cochlae and make synapses on the OHCs directly. Furthermore, the density of efferent fibers is substantially greater for the OHCs than for the IHCs.

2.1.2 An Overview of Cochlear Mechanics

The footplate of the stapes is attached to a flexible membrane in the bony shell of the cochlea called the oval window, that leads to the SV. The inward and outward movements of the stapes as a result of incident sound waves create a pressure difference across the cochlear partition, which propagates down the length of the cochlea. Such a pressure wave deflects the BM, giving rise to the so-called travelling wave (TW) along it. The bending of the BM results in a shear motion between the BM and the TM, which further deflects the stereocilia of the sensory cells. Such deflections of the hair bundles lead to transduction currents inside the sensory cells, which for the OHCs, cause the expansion and contraction of the cell body, a phenomenon known as somatic motility or electromobility [30] and for the IHCs, give rise to neurotransmitter release at the IHC-auditory nerve synaptic interface which triggers the generation of neural impulses that are sent to the brain. The forces generated by the OHC mobility are believed to actively amplify the TW and sharpen its response along the BM for low level inputs. Such

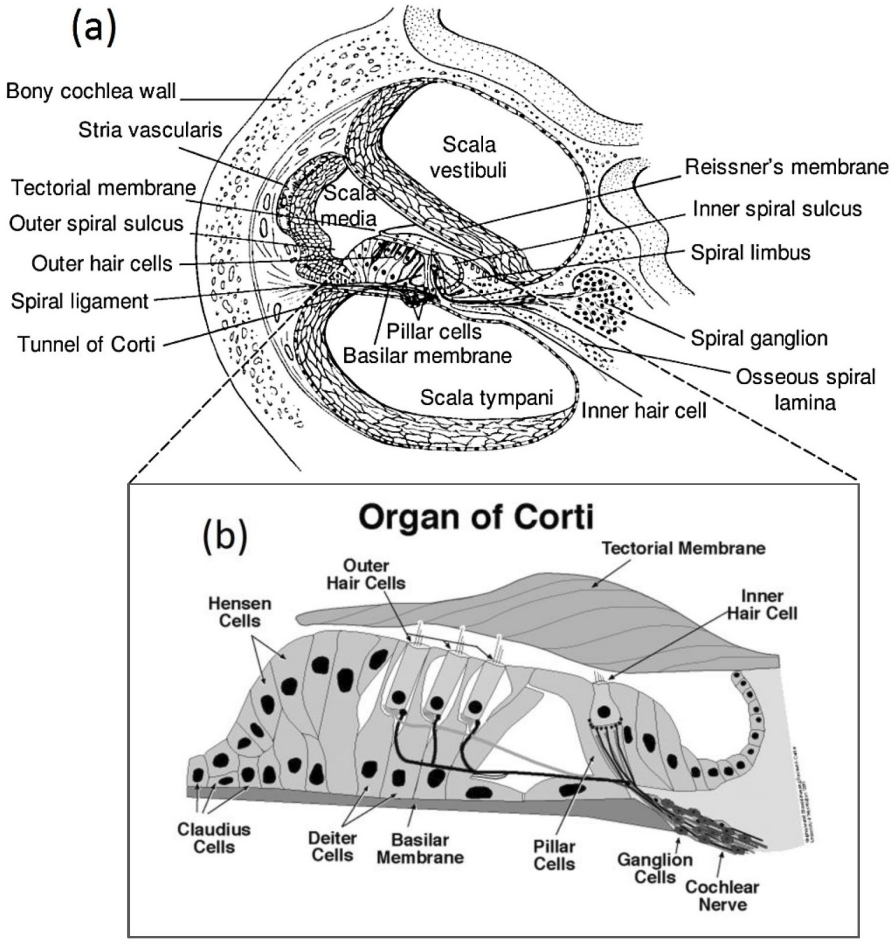


FIGURE 2.2: Cross section of the cochlear duct (a) and the Organ of Corti (b). Subplot (a) and (b) is obtained from [28] and [29] respectively with permission.

a biological feedback system is referred to as the cochlear amplifier, although this amplifier is not linear but compressive, as described in more detail below.

The properties of the BM are not uniform along its length, gradually decreasing in stiffness but increasing in width from its base (the oval window) to the apex. Such variation causes a natural tuning of the vibration of the BM. For a pure tone stimulus, the frequency which causes the maximum response at a particular position along the cochlear partition decreases monotonically from the base to the apex. This frequency is called the characteristic frequency (CF) of this specific place, and this place is in turn called the characteristic place (CP) of this frequency. As will be shown in later sections, the CF of a position along the cochlea tends to decrease with increasing stimulus level, CF is usually defined as the value obtained at very low input stimulus level, such auditory threshold. The frequencies obtained

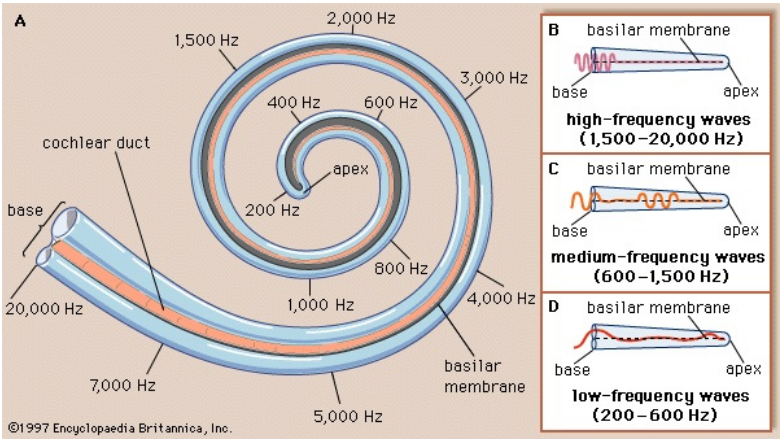


FIGURE 2.3: Tonotopical arrangement along the human cochlea. Picture taken from [32] with permission.

at higher input levels are often called best frequencies. In other words, each point of the BM is best tuned to a single frequency and only a specific group of hair cells around that point are activated for a pure tone stimulus, with high frequency inputs mainly activating the basal section while lower frequency stimuli mainly activating the more apical part. This tonotopical arrangement is illustrated in Fig. 2.3, where the approximate distribution of the CFs along the BM is drawn beside the spiral structure. Analytic expression approximating the relationship between the characteristic frequency and place along the BM has been established for several species in [31] and it is of the following form,

$$F = A(10^{ax} - k) \tag{2.1}$$

where x is the distance from the apex, A and k are constants that can be adjusted for different species to achieve a desirable frequency range. However, constant a is found to be roughly equal to 2.1 across many species investigated in [31], if the distance variable x is expressed as a proportion of total cochlear partition length (apex=0, stapes=1). Therefore, the cochlea is often treated as a spatial spectral analyzer that decomposes the input sound signal into its frequency components through this frequency-place mapping.

2.1.2.1 Single Tone Responses: Compressive Amplification

For a pure tone stimulus at the CF of the place under consideration along the cochlea, its response is greatly amplified at low levels and behaves roughly linearly for sound pressure levels of up to around 20 dB. However, at medium and high levels, the BM response grows compressively, with a growth rate typically of less than 0.5 dB/dB and can be as low as 0.2 dB/dB in very sensitive cochleae [33]. At even higher levels, the cochlea tends to become linear again in some experiments, but in other cases, compressive growth is maintained up to the highest intensities tested ($>$ 100 dB SPL) in very sensitive cochleae [33]. Fig. 2.4 shows an example of the growth of the BM response, or the so-called input-output (I/O) function, at the characteristic place of the single tone frequency tested for increasing stimulus level, illustrating these three different regions of response growth. It can also be seen that the input-output function becomes more linear and loses a large amount of gain when the cochlea is damaged, suggesting that this behaviour only exists in normal functioning cochlae. Such a phenomenon is commonly believed to originate from nonlinearities residing in the cochlear amplifier [34] that is probably mediated by the OHCs. From an engineering perspective, a normal cochlea realizes an automatic gain control, through which the gain of the cochlear amplifier becomes attenuated when the input intensity is increased. This compressive behaviour is also known as self-suppression [35] and is believed to be the main reason that allows the human auditory system to encode the remarkably large input dynamic range of 140dB SPL in the auditory nerve firing rates, which only have a dynamic range of 30-40 dB [36].

During the increase of stimulus intensity, the most responsive frequency and degree of tuning at the location under consideration decreases gradually. Fig. 2.5 (a) and (b) show two families of isointensity curves, measured at a BM site with a CF of 10 kHz in a normal chinchilla cochlea, representing the velocity and gain (velocity divided by stimulus sound pressure) of BM responses to single tone pips as a function of frequency and intensity (in dB SPL) respectively. It can be seen that the BM response to single tones has a nonlinear dependence on stimulus level close

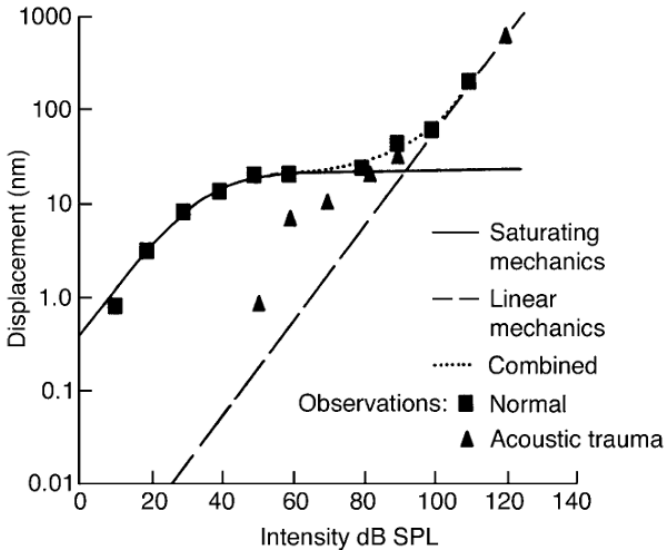


FIGURE 2.4: Input-output function of a BM site in response to characteristic frequency tones as a function of input SPL. Responses were measured before (squares) and after (triangles) trauma in a guinea pig cochlea. The solid, dashed and dotted lines represent the growth function that would be expected from a saturated mechanism, a linear mechanism or a combined mechanism respectively. Figure adapted from [37] with permission.

to the most responsive frequency, and a more linear dependence at higher and lower frequencies. The frequency response of the measured BM site gradually broadens as sound level increases, becoming much more flat at the highest stimulus level. In addition, the most responsive frequency shifts downwards by about half an octave after raising the sound level by 90 dB. For this reason, the CF of a specific location along the cochlea is defined at low stimulus levels, such as auditory threshold, when the cochlea is fully active, while the most responsive frequency is usually called the best frequency which can shift up and down as signal level changes.

There is also a clear difference between BM frequency response at the basal and apical sites, as shown in Fig. 2.6. The apical BM response is less sharply tuned and exhibits less gain from the cochlear amplifier compared to that of a more basal site. In addition, the frequency of maximum sensitivity is almost independent of stimulus level in the apical region but reduces with stimulus level in the basal region of the cochlea.

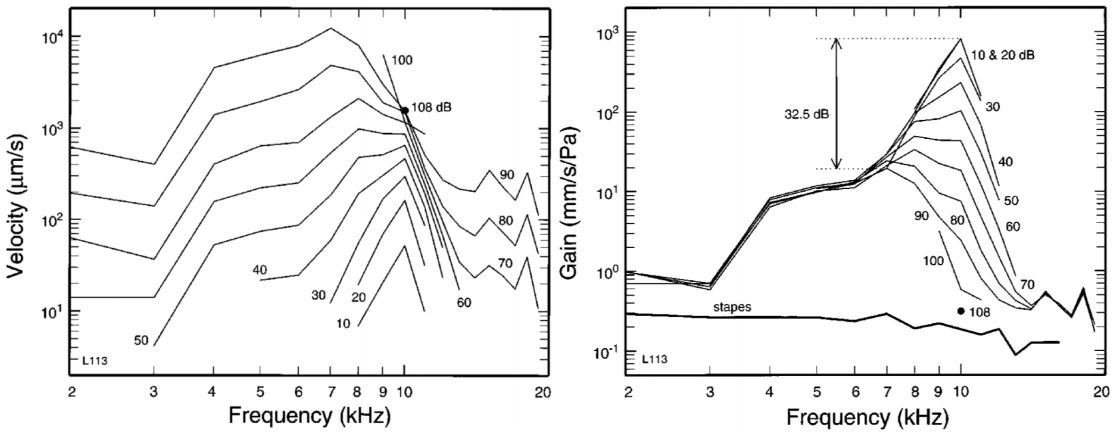


FIGURE 2.5: a) A family of isointensity curves representing the velocity of BM responses to single tone pips as a function of frequency and intensity (in dB SPL). b) A family of isointensity curves representing the gain or sensitivity (velocity divided by stimulus pressure) of the BM site to single tone pips as a function of frequency and intensity (in dB SPL), plotted using the same data as that shown in a). Figure adapted from [38] with permission.

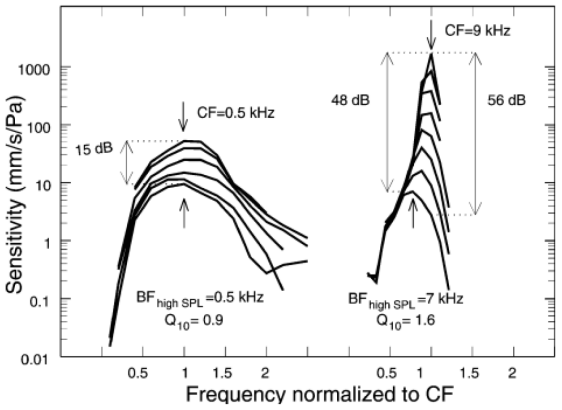


FIGURE 2.6: Frequency responses of the BM at a basal and apical location. Two families of sensitivity curves (ratio of BM velocity to stimulus pressure) were recorded at the 0.5 and 9 kHz characteristic places in the chinchilla cochlea at different stimulus frequencies and levels. The best frequencies and 10 dB quality factors for the highest stimulus level are also indicated for each location. Figure adapted from [33] with permission.

2.1.2.2 Two-tone Suppression

The nonlinearity of the cochlear amplifier can also generate various nonlinear interactions between different spectral components of more complex stimuli, such as pairs of tones. Two-tone suppression describes the phenomenon whereby the magnitude of the output signal in the live cochlea in response to a test tone is reduced in the presence of another tone. It was first observed at the level of the auditory nerve [39], where it was shown that the average discharge rate of an AN fiber to a CF tone could be reduced by the addition of a second tone of appropriate frequency and sound pressure level. Analogous suppression of the overall mechanical vibrations of the BM in response by a suppressor tone has been found in[40]. However, [41] and [42] did not find any overall suppression, but did observe reductions in the spectral component at the CF when a second tone was added. A suppressor tone with a frequency higher than that of the probe tone is termed a high-side suppressor, while the one with a lower frequency is called a low-side suppressor. Fig. 2.7 shows an example of measured two-tone suppression in a chinchilla cochlea for both (a) high-side suppressor tones and (b) low-side suppressor tones. Each curve represents the magnitude of the BM velocity versus the SPL of the probe tone as the level of the suppressor tone is varied. It can be seen that as the level of the suppressor tone is increased, the intensity of the probe tone must also increase to maintain the same BM velocity.

The most prominent difference between them is that the amount and growth rate of suppression of the probe tone with suppressor intensities is larger and steeper for low-side suppressors than for high-side suppressors, which is a direct consequence of the asymmetric nature of the travelling wave in the cochlear. A similar suppression effect has also been observed in the receptor potentials of the IHCs [43], psychophysics experiments with human subjects [44]. Many aspects of these phenomena are explainable by BM mechanics, except for example, the difference between BM and auditory nerve responses due to low-side suppressions [45].

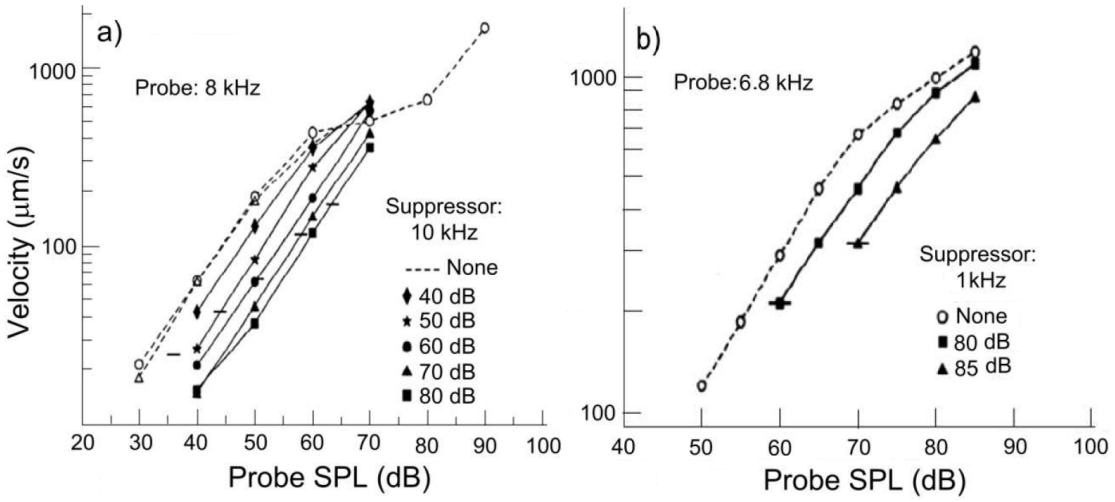


FIGURE 2.7: Two-tone suppression of the BM velocity responses measured in the chinchilla cochlea for both (a) higher-side suppressor tones and (b) lower-side suppressor tones. Figure adapted from [40] with permission.

2.2 Machine Learning and Deep Learning

Machine learning is the study of algorithms that are able to learn from data without being explicitly programmed. It is a valuable tool for solving many tasks that are often very difficult to be accomplished by fixed programs designed by humans. Some examples of the tasks that machine learning has proved to be powerful in are classification, regression, transcription, machine translation, denoising and probability distribution estimation. Generally speaking, machine learning algorithms can be divided into two categories: supervised and unsupervised, depending on the type of examples they are given during the learning process. In supervised learning, the machine has access to both a training dataset and a label or target associated with each example in the dataset. It must learn to produce the target as close as possible, given the corresponding example, and can thus be regarded as a function approximator. In fact, many practical problems can be formulated as function approximation, such as the speech processing tasks addressed in this thesis, i.e., voice activity detection (classification of speech from non-speech), and speech separation (classification or regression depending the targets used). Unsupervised learning on the other hand, refers to algorithms that are capable of learning useful properties of the structure of the training dataset without the guide of

targets. Examples of this category include dimensionality reduction, probability distribution estimation and clustering. Other learning paradigms that do not belong to any of these two classes also exist, such as semi-supervised learning [46], transfer learning [47] and reinforcement learning [48], although only supervised learning is used in this work.

In most cases, a machine learning method is composed of four fundamental elements: a dataset, a model, an objective function and an optimization procedure. The objective function measures the accuracy of the chosen model in predicting the desired targets, given the examples in the dataset, while the optimization procedure trains the model towards achieving optimal performance as measured by this objective function. It is common practice to formulate such optimization problems as minimizing the error between predicted and true targets in this field, so the objective function is also often called the cost or loss function. The choice of this cost function depends on the task that the machine learning algorithm is required to solve. For instance, in classification problems, the cross entropy error is usually chosen, while in regression problems, the mean square or mean absolute error is often adopted. In terms of optimization, the most famous and popular approach is the so-called method of steepest descent or gradient descent, which modifies the parameters of a model by repeatedly taking small steps in the direction opposite to the gradient of the error function with respect to model parameters. Suppose we have a training dataset consists of a total number of N examples, $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_i, ..., \mathbf{x}_{N-1}, \mathbf{x}_N\}$ and targets $\mathbf{T} = \{\mathbf{t}_1, \mathbf{t}_2, ..., \mathbf{t}_i, ..., \mathbf{t}_{N-1}, \mathbf{t}_N\}$, where $(\mathbf{x}_i, \mathbf{t}_i)$ is a single pair of example and target vectors with index i. If we represent the model as a function, $f(\mathbf{x}; \boldsymbol{\theta})$, with its parameters denoted as $\boldsymbol{\theta}$, and output denoted as $\mathbf{y}$, the method of gradient descent updates the model parameters with the following rule,

$$\boldsymbol{\theta(n)} = \boldsymbol{\theta(n-1)} - \epsilon \frac{\partial e(\mathbf{Y}, \mathbf{T})}{\partial \boldsymbol{\theta(n-1)}} \tag{2.2}$$

where $\mathbf{Y} = \{\mathbf{y}_1, \mathbf{y}_2, ..., \mathbf{y}_i, ..., \mathbf{y}_{N-1}, \mathbf{y}_N\}$ is the outputs of the model given the training examples, $\mathbf{X}$; $e(\mathbf{Y}, \mathbf{T}) = \sum_{i=1}^{i=N} e(\mathbf{y}_i, \mathbf{t}_i)$ is the error measuring the difference

between the model outputs and the true targets over all of the training examples; ϵ is a scalar learning rate parameter that determines the size of each update step; and $\boldsymbol{\theta(n)}$ represents the model parameters after the n^{th} update. A complete pass through all of the training examples with parameter update during the process is called one epoch of training in machine learning.

It is important to remember that machine learning differs from pure mathematical optimization in the sense that it must also perform well on new and previously unseen examples, i.e., test dataset. Such an ability is called generalization and the error measured on the test dataset is called generalization or test error. The central challenge of machine learning is to make both training and testing error small. Otherwise, a specific algorithm is said to be underfitting if it fails to obtain a sufficiently small error on the training set, or overfitting when the difference between training and testing error is too large. An excellent example illustrating these concepts is the polynomial curve fitting problem covered in detail in Chapter 1 of [49]. In summary, the capacity of a model should be sufficiently powerful to be able to tackle the true complexity of the task in order to prevent underfitting, but not overly powerful which could lead to overfitting. The techniques that are designed to reduce the test error but not the training error are collectively called regularization, and the development of these techniques has been one of the major research directions in machine learning.

There is a wide range of regularization techniques available for designing effective machine learning algorithms. Probably the most classical one is the addition of the parameter norm to the original cost function in order to penalize the size of model parameters. The most popular choice of such regularisation is the L^2 parameter norm penalty, which is the squared Euclidean distance between the point specified by the model parameters, $\boldsymbol{\theta}$, and the origin, and is commonly known as weight decay or leakage. But in this thesis another two regularisation strategies are adopted: early stopping and dropout training, which are discussed in more detail in section 2.2.3.

Deep learning is one type of the machine learning techniques discussed above. It provides a powerful framework for supervised learning by representing complicated concepts through a nested hierarchy of processing layers, with each subsequent layer extracting a progressively more abstract representation of the input data. By adding more layers and more units within a layer, a deep network can represent functions of increasing complexity, given sufficiently large datasets of labelled training examples. The initial phase of deep learning research aimed to find artificial models of biological learning process, i.e., how learning happens in the brain. This inspired a large portion of work in this field and contributed to many famous models that are critical for its recent resurgence. But modern deep learning should not be viewed as an attempt to simulate the brain, as it draws inspiration from many fields, especially probability, information theory, and numerical optimization. Moreover, the primary objective of modern deep learning is to build computer systems that are capable of successfully solving tasks that require intelligence, rather than attempting to understand the neuroscience of how the brain works. Deep learning has contributed to dramatic improvements in a wide range of speech processing tasks, such as noise reduction, speech recognition and natural language processing. This progress is made possible by a number of redeveloped or newly proposed deep architectures, of which the most widely adopted three types are deep feed-forward neural networks, or simply deep neural networks (DNN), convolutional neural networks (CNN) and recurrent neural networks. In the following, we present a brief description of DNN and CNN only because recurrent neural network is not utilized in this work. See [18] for a comprehensive review and discussion on these and deep learning in general.

2.2.1 Deep Feed-forward Neural Networks

Feed-forward neural networks (FNN) are very important in the field of machine learning and form the basis of many successful commercial applications. The most widely used form of FNN consists of a hierarchy of many layers, with the first one, middle one(s) and last one usually called input layer, hidden layer(s) and output

layer respectively. Each layer contains a number of (typically nonlinear) processing units, called artificial neurons, as shown in Fig. 2.8. In this work, the input layer only represents the input feature vector, while the actual signal processing layers are the hidden layer(s) and the output layer.

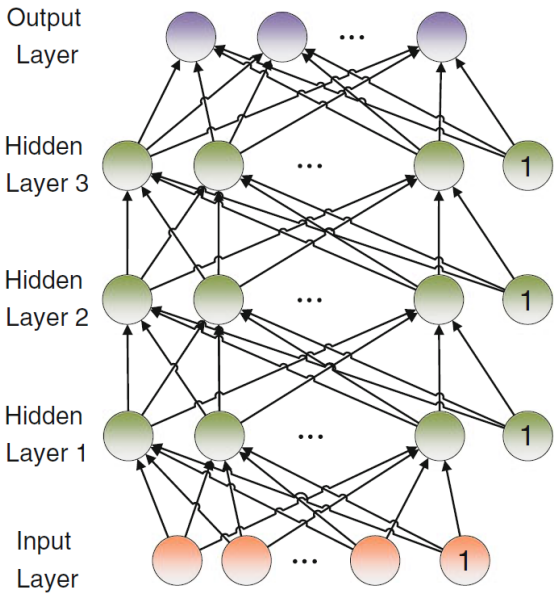


FIGURE 2.8: An example of deep feed-forward neural network with an input layer, three hidden layers, and an output layer. Picture reproduced from Fig. 4.1 of [20] with permission.

These models are described as neural networks because they are loosely inspired by neuroscience. Each unit in a given layer receives a weighted combination of activations from units in the previous layer and transforms the summed inputs to its own output through an activation function, $f(z)$, which is typically nonlinear and in biological terms, represents the average firing rate of a actual neuron, as given below,

$$\mathbf{z} = \mathbf{W}\mathbf{x}+\mathbf{b} \tag{2.3}$$

$$\mathbf{y} = f(\mathbf{z}) \tag{2.4}$$

where $\mathbf{x}$ and $\mathbf{y}$ are the input and output vector of a given layer respectively; each row of matrix $\mathbf{W}$ and each element of the column vector $\mathbf{b}$ contain the weights and bias connecting the "neurons" from the previous layer to a corresponding "neuron"

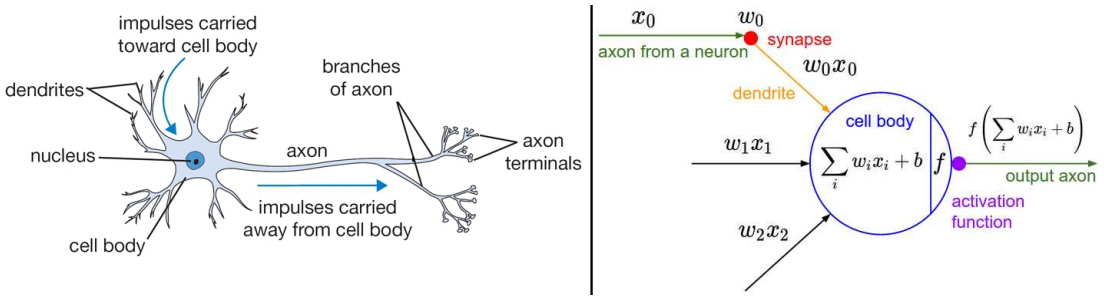


FIGURE 2.9: A comparison of a biological neuron and its mathematical representation in neural networks shown in Fig. 2.8. Picture reproduced from [19] with permission.

in the current layer; and $\mathbf{z}$ denotes the input to the activation function computed from an affine transformation of the input vector $\mathbf{x}$ using "neuron" connection weights and bias. Fig. 2.9 shows a graphical comparison of a biological neuron and its mathematical simplification that is widely used in machine and deep learning.

Typically, neural networks with at least two or more hidden layers are considered to be deep, and are often simply called DNNs, while those with only one hidden layer are traditionally referred to as multilayer perceptron (MLP). They are called feed-forward networks because information only flows in one direction, from the input layer, through intermediate layers, to the output layer. In other words, there are no feedback connections in the model. However, including feedback connections is possible and this extends them to another type of network structure called recurrent neural network. Historically, the two most common choices of activation function for the hidden units of neural networks are hyperbolic tangent, $\tanh(x)$, and logistic sigmoid, $\delta(x)$, functions, which are defined and related as given below,

$$f(z) = \tanh(z) = \frac{e^{2z} - 1}{e^{2z} + 1} \tag{2.5}$$

$$f(z) = \delta(z) = \frac{1}{1 + e^{-z}} \tag{2.6}$$

$$\tanh(z) = 2\delta(2z) - 1 \tag{2.7}$$

But there are well-known limitations with these functions, for instance, the learning of a neural network slows down because of weak gradients when the units are close to saturation in both very negative and very positive directions, a problem

known as the vanishing gradient [50]. To overcome this inherent weakness of saturating nonlinearities, rectified linear neurons [51, 52, 53] have been proposed to allow faster learning in deep networks with many layers, an example of which is called Rectified Linear Unit (ReLU) and is shown in Eq. 2.8. Various alternative activation functions are also available, such as the maxout function, the radial basis function and the softplus function.

$$f(z) = \max(z, 0) \tag{2.8}$$

For the output layer, both the choice of number of units and the activation function depend on the specific task at hand. For instance, in binary classification tasks, the standard configuration is to have one unit with the logistic sigmoid function. The output of this unit represents an estimate of the probability that the input data belongs to one class, while one minus its output represents the probability estimate of the other class. For multinomial classification problems, the convention is to use an output layer with the same number of units as the number of categories to be classified and estimates the conditional probability for each class with a softmax function, given below,

$$p(C_k|\mathbf{x}) = y_k = \frac{e^{z_k}}{\sum_{\tilde{k}=1}^{K} e^{z_{\tilde{k}}}} \tag{2.9}$$

where C_k represents the k^{th} class of a total number of K classes, $\mathbf{x}$ is an input vector, z_k and y_k are the input and output of the k^{th} activation function in the output layer.

2.2.2 Convolutional Neural Networks

Convolutional neural networks (CNNs) are a specialized class of neural networks that are inspired by mammalian visual neuroscience. They have been hugely successful in the computer vision community since the 1990s, long before arbitrary deep networks were considered viable, and have now been extended to the speech

processing domain. As its name suggests, the key element of CNN is the use of convolution operation with constrained connectivity patterns and parameter sharing. This deviation from the traditional fully-connected DNN is especially useful for high dimensionality inputs that exhibit some kind of topological structure, such as the ordering of pixels in an image or the spectral-temporal modulation of an audio signal. A typical CNN is composed of a convolutional layer, a pooling layer and a few fully-connected layers, as illustrated in Fig. 2.10, which is designed for speech processing tasks with 2-dimensional spectral-temporal features. Multiple convolutional and pooling layers can be stacked on top of each other before connecting to fully-connected layers. Such a combination of the convolutional and pooling layer is often called the feature extracting layer as it generates data-driven features that are directly learned from the training data. Subsequent fully-connected layers build upon these learned features, instead of conventional hand-engineered features, to do the final classification or recognition. Since both feature processing and pattern recognition are jointly optimized by the learning algorithm, CNNs generally outperform traditional techniques that tackle these two tasks separately.

The so-called convolutional layer includes a number of feature maps, each unit of which only receives inputs from a small local 2-dimensional (2D) patch or receptive field. This receptive field shifts in one or both dimensions until it sweeps through all vertical or horizontal channels within a context window to construct a complete feature map. The weights of all the neurons in one feature map are usually constrained to be the same, a scheme called weight sharing. This contributes to the shift invariance property of CNN and substantially reduces the number of unique parameters in the whole network. Under this scheme, the forward pass through this layer can be realized as a convolution operation between the neuron's weights and the input features within the context window, and hence the name convolutional layer. The set of parameters of each neuron is similarly often called a filter or a kernel. After the convolutional layer, nonlinear activation function as introduced 2.2.1 is usually added as well. The pooling layer is normally used to reduce the dimensionality of previous layer's outputs, which can further increases

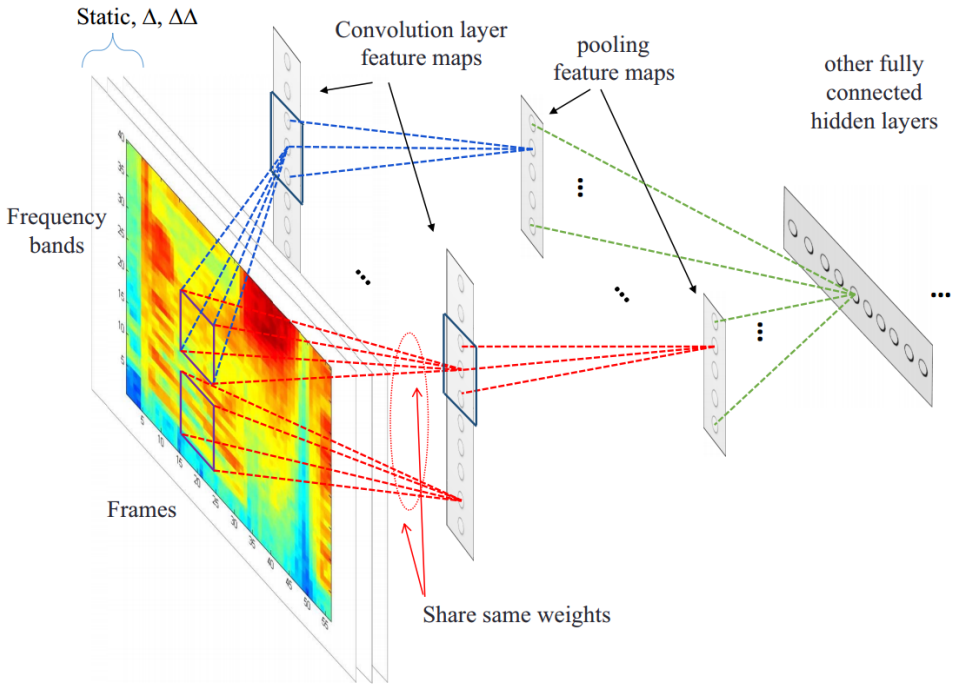


FIGURE 2.10: Illustration of a typical CNN, applied to a speech processing tasks, using a 2-dimensional time-frequency representation as input features. It consists of one convolutional layer, one pooling layer and a few fully-connected layers. Picture adapted from Fig. 3 of [54] with permission.

the robustness to distortions of subsequent layers. For example, the most widely adopted max-pooling layer involves picking the maximum value from a group of adjacent neuron's outputs.

2.2.3 Training of Neural Networks

As one class of machine learning techniques, training of neural network also follows the principles discussed in section 2.2. Optimization strategies are still mostly based on the first-order method of gradient descent, which needs the derivatives of the cost function with respect to the weights and bias of every layer in the network. These gradients are computed using a technique called back-propagation [55], or simply backprop, which is essentially a repeated application of the chain rule of calculus for computing the partial derivatives. It first computes the derivatives of the cost function with respect to the outputs of a network, and then the gradients

with respect to the weights and biases of the hidden part of the network layer-by-layer, in the reverse order, based on the chain rule, and hence the name back-propagation. Detailed derivation of this algorithm has been given in many places, hence is not repeated here. Please refer to Chapter 2 of [56], for instance, for such a treatment.

After obtaining the gradients, neural network parameters can be updated using the standard equation of gradient descent, as given in Eq. 2.2, although there are some important points to note for effective and fast training of deep networks. First, most optimization algorithms in deep learning only use a subset or mini-batch of all the training examples for each gradient computation and parameter update, which is generally called mini-batch or stochastic gradient descent. Secondly, the learning rate is usually annealed during training. Common choices of the decay strategy include step decay (reduce the learning rate by some factor every few epochs), exponential decay and $1/t$ decay (t is the iteration number) [19]. Furthermore, the standard gradient descent can make learning slow sometimes, for example in the case of loss function having high curvature and small gradients. The method of momentum [57] is an approach that is often adopted to accelerate the convergence speed when training deep networks, especially in such conditions. This method accumulates an exponentially decaying moving average of past gradients and continues to move the parameters in that direction. This effectively adds inertia to the search through the parameter space and modifies Eq. 2.2 to the following,

$$\boldsymbol{\theta(n)} = \boldsymbol{\theta(n-1)} + \Delta\boldsymbol{\theta(n)} \tag{2.10}$$

$$\Delta\boldsymbol{\theta(n)} = m\Delta\boldsymbol{\theta(n-1)} - \epsilon\frac{\partial e(\mathbf{Y},\mathbf{T})}{\partial\boldsymbol{\theta(n-1)}} \tag{2.11}$$

where $0 \leq m \leq 1$ is the momentum parameter; $\boldsymbol{\theta(n)}$ represents the weights and biases of a neural network after the n^{th} update, and $e(\mathbf{Y},\mathbf{T})$ is the error function as defined in Eq. 2.2. Similar to learning rate decay schedules, optimization can sometimes benefit a little from momentum schedules, where the momentum is increased in later stages of learning. A typical setting is to start with a momentum

at about 0.5 in a few initial epochs and increase it to 0.99 or so over multiple later epochs [19]. A number of more sophisticated gradient descent based algorithms have also been developed, including Nesterov momentum and adaptive learning rate algorithms such as AdaGrad [58], RMSProp [59], AdaDelta [60] and Adam [61]. These adaptive learning rate algorithms are different from simple learning rate decay methods as presented above because the learning rate for each network weight and bias is continuously adjusted after each parameter update during training.

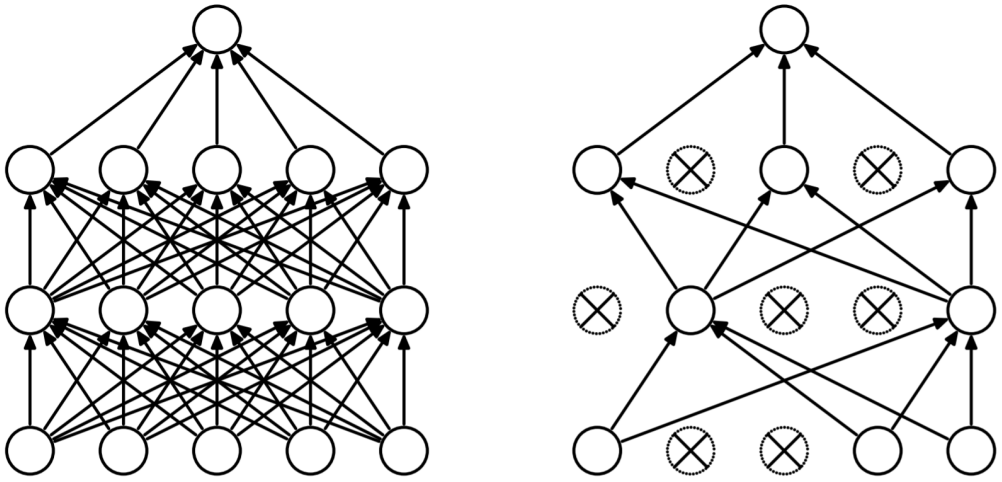
It is worth noting that the core algorithms for training deep networks have been developed since 1980s [55], but they have not been widely used until recently because deep networks are traditionally believed to be very difficult to train. Indeed, gradient information tends to vanish with saturating nonlinear units as it is propagated backwards through several layers, making it difficult to train the lower layers of a deep network. In the initial period of deep learning renaissance since 2006, unsupervised greedy layer-wise pretraining [62] was considered to be the key to largely avoid bad local minima of the cost function and the vanishing gradient problem, so that deep models can be trained effectively. But with much larger datasets, larger models, more advanced training and regularization strategies and more powerful computer hardware and software infrastructure, training deep networks has become much more viable, which in turn improves their performance dramatically. Furthermore, the introduction of the linear rectification activation function significantly reduces the problem of vanishing gradient. All of these make pre-training unnecessary in most cases.

2.2.3.1 Regularization

As in general machine learning, regularisation is an important element of the training procedure, because it improves the generalization ability of a neural network by preventing overfitting. In this thesis, we use two of these methods, called early stopping and dropout training. Early stopping is probably the most commonly used regularization technique in deep learning, because of its effectiveness and

simplicity. To use this method, a small part of the training set (usually 10-20%) is reserved as a validation set. At the end of each training epoch, the error function is evaluated on this separate validation set with the newly trained parameters. This validation error is recorded and compared with those obtained at the end of previous training epochs. If the current set of weights and biases yields the lowest validation error, it is stored as the current optimal set of model parameters, until they are replaced by newly trained values obtained from a later training epoch that produces a new lowest validation error. When the validation error fails to decrease for a consecutive certain number of epochs, 10 for instance, the training process is stopped, hence the name early stopping. Using this method, it is typically observed that the error decreases for both training and validation sets initially, but after a certain number of epochs, the validation error stalls and (or) begins to increase, while the training error continues to decrease. This means that the network is beginning to overfit the training dataset, and training should thus be stopped at the point before it happens.

Dropout is another technique that prevents overfitting by approximately combining many different neural network architectures together efficiently. The term "dropout" refers to temporarily removing units from the network, along with all its incoming and outgoing connections, as shown in Fig. 2.11. The decision of which units to drop is random, and a common practice is that each unit is retained with a fixed probability, p, independent of other units [63]. This effectively samples a sub-network from the original one, consisting of all the units that survive the dropout operation (Fig. 2.11 (b)). For each iteration of training, a new sub-network is sampled and trained. However, during testing, the complete network without dropout is used, except that the outgoing weights of the units in this network are scaled down by a factor of the retain probability, p. This is to ensure that the expected output of any hidden unit is the same during training and testing [63].



(a) Standard Neural Net (b) After applying dropout.

FIGURE 2.11: The application of dropout to a neural network. (a) A standard neural network with 2 hidden layers. (b) The same network after applying dropout. Units labelled with cross have been dropped temporarily. Picture taken from [63] with permission.

2.3 Summary

In this chapter, the background to two major parts of this thesis is discussed, i.e., the mammalian cochlea and machine learning. Basics of cochlear anatomy are given, followed by some discussions of nonlinear cochlear mechanics in response to single-tones and two-tones. These are some of the key characteristics that a reasonable model should aim to reproduce, and will be used as a reference to compare the properties of two cochlear models in the next chapter. An introduction to machine learning, with a focus on deep neural networks, is presented. Standard network structures and training strategies that will be used in Chapters 4 and 5 have also been discussed.

Chapter 3

Comparison of a Transmission -line and a Filter Cascade Model of the Human Cochlea

The human auditory system is remarkably effective and robust under a wide variety of disturbances. One way of understanding its physiological mechanisms and the resulting psychophysics is to build computational models of it. Such models have provided inspirations for effective and robust algorithms in many areas of acoustic signal processing, such as voice activity detection, speech enhancement, and speech recognition, narrowing the performance gap between the human auditory system and machine hearing systems. Most of these modelling efforts start from the cochlea, the first significant computational component of the auditory system. In this chapter, we investigate two types of cochlear models, one transmission-line model and one filter cascade model. The theoretical relationship between their linear responses is first established. The nonlinear dynamics of both models are then thoroughly tested and compared to experimental measurements of biological cochlea responses, specifically in terms of self-suppression and two-tone suppression (2TS) on the level of BM. This comparison serves as an important indication of the validity of a model, especially when used as a front-end module for simulating the entire auditory system. Because modelling errors at

this step will inevitably propagate to later stages, and may also limit its capability in signal processing applications. Single tone responses have been extensively tested in a number of TL models [64, 65, 66] and phenomenological models [67, 68, 69]. Two-tone suppression on the level BM motion, however, have been much less comprehensively investigated in these models. In fact, most of them only show 2TS for a limited number of pairs of probe and suppressor tones [70, 71, 72], although neural two-tone suppression has been modelled in more detail in [73, 69]. Both self-suppression and 2TS are important in many areas of auditory signal processing, such as the peripheral encoding of speech and music stimuli and the determination of perceptual masking effects, such as upward spread of masking.

3.1 Introduction

Cochlear models have been formulated in numerous ways (see a comprehensive review in [74]) and can be broadly divided into two classes, as discussed by Duifhuis [75]. The first class, which includes the so-called transmission-line models [76, 77], attempts to explicitly simulate the interaction between the fluid dynamics and basilar membrane mechanics within the inner ear using partial differential equations. They can be computationally demanding, especially when implemented in the time domain. The second class, often referred to as phenomenological or signal processing models, is only concerned with reproducing the observed outputs of the auditory system for given inputs, for example using a parallel bank of bandpass filters, such as the gammatone [16], gammachirp [78], band-pass nonlinear [67, 71] and dual-resonance nonlinear [68] filters. The underlying biophysical processes are not explicitly considered in these models, but they can be much more efficient to implement. The filter cascade model [79, 80, 81, 12] sits between these two classes, since it is both computationally efficient and, in effect, simulates the forward cochlear travelling waves. To account for cochlear nonlinearity, a coupled automatic gain control (AGC) network can be integrated into the model simulating the fast compressive responses and inter-channel interactions, giving rise to a dynamic version of the model, called the cascade of asymmetric resonators with

fast acting compression or the CAR-FAC model [12]. This enjoys both the physiological plausibility of TL models and the computational efficiency of filter-based ones, achieving a good compromise between model capacity and implementation complexity.

One weakness of the CAR-FAC model is that it does not reproduce the backward travelling waves in the cochlea, which are believed to generate otoacoustic emissions. The TL model is capable of reproducing the backward travelling waves and hence otoacoustic emissions, particularly when small levels of random irregularities are introduced into its parameters [82]. A price that must be paid for this feature is that the TL model can become unstable. Whereas this can be an advantage in the simulation of spontaneous otoacoustic emissions [83], this becomes a problem when it limits the maximum level of active gain that can be obtained for single tone excitations, as described below. Since the comparison between the CAR-FAC and TL models is motivated by their use as front-end processors for signal processing, otoacoustic emissions are not considered here and focus is put on their abilities to model the normal responses of the mammalian auditory system at the peripheral level.

The parameters of the filter cascades, such as the all-pole and pole-zero filter cascades, have been fitted with human notched-noise masking data to predict tone-in-masker detection thresholds based on the EQ-NL theorem by de Boer [84]. It was shown to outperform various versions of the gammachirp filter with fewer parameters [80]. By moving its poles and zeros in the Laplace domain with input level, it can also be adjusted to match the level-independent zero-crossing times in the impulse responses [81], as observed from physiological measurements. Using these fitted parameterizations and nonlinear adaptations, improved performance has been demonstrated when using the CAR-FAC as a front-end processor, compared with other standard techniques like MFCC, in a number of machine hearing applications, e.g. content-based music retrieval [85] and large-scale cover song recognition [86].

3.2 Model Descriptions

3.2.1 The Transmission-line Model

The transmission-line model used here is one-dimensional, active and nonlinear, state-space model developed by Elliott et al. [1], that uses the micromechanical model proposed by the Neely and Kim [9], but with parameters fitted by Young [87]. It simplifies the biological cochlea into a symmetric fluid-filled rectangular box, separated by a flexible partition representing the combined effect of the scala media, the OC and the BM, as shown in Fig. 3.1.

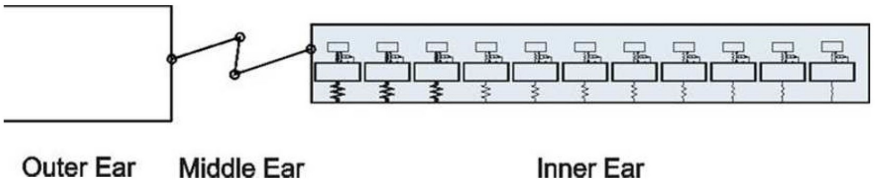


FIGURE 3.1: The box model of the human cochlea. Adapted from Fig. 2 of [88] with permission.

The Neely and Kim's micromechanical model represents an individual radial slice of the cochlea partition by a two degree-of-freedom lumped-parameter oscillators, as shown in Fig. 3.3. The two masses, M_1 and M_2, can be interpreted as the masses of the BM and the TM respectively and they are coupled by the impedance of the OHC. To simulate the cochlear amplifier, the OHC impedance is hypothesized to contain two parts: a passive part, $Z_3 = K_3/s + C_3$ and an active part, Z_4, where s is Laplace variable. This enables the model to replicate measured frequency selectivity and sensitivity of biological cochlae. A total number of 500 discrete elements are used to represent the full length of the cochlea, which is set to 35 mm. This is considered to be a sufficient number of elements in order to obtain accurate numerical solutions of the transmission-line model, as shown in [10] and [87]. The first 499 of them are Neely and Kim's micromechanical elements that simulate the cochlear partition, while the last element is a mass-damper system simulating the boundary condition at the helicotrema, as described in the appendix of [89]. All of these elements are locally reacting and coupled solely by the inertia of the cochlear fluid.

A dimensionless factor, γ, regulates the strength of the active feedback force, so that when $\gamma = 1$, the TL model is deemed as fully active while the model becomes fully passive with $\gamma = 0$. The saturation of the cochlear amplifier is simulated with a first-order Boltzmann function placed before the active impedance in the feedback loop of each oscillator, as illustrated in Fig. 3.3. Such an arrangement and choice of nonlinear function is to better represent the harmonic responses and distortion products generation, while maintaining reasonable responses to pure tones [90]. Furthermore, the input-output characteristic of the Boltzmann function has been shown to be a good match to that of the OHCs [91, 92, 93]. The factor γ is then computed, at each element and time step, as the absolute value of the ratio of the output and input of the Boltzmann function [83], so that it decreases with increasing stimulus level.

$$\gamma(n,t) = \left| \frac{f(x_d(n,t))}{x_d(n,t)} \right| \tag{3.1}$$

$$f(x_d(n,t)) = \alpha \left(\frac{1}{1+\beta e^{(-x_d(n,t)/\eta)}} - \frac{1}{1+\beta} \right) \tag{3.2}$$

where n is element index, t is time step index, $x_d(n,t)$ is the relative displacement between the two masses, $f(x_d(n,t))$ is the Boltzmann function, whose parameter α sets the saturation point, η affects the slope of the output, and β affects the asymmetry of the function. In order to set the slope of the function to unity for small input displacements, η is set to $\frac{\alpha\beta}{(1+\beta)^2}$. Fig. 3.2 shows some examples of Boltzmann function, with varying values of α and β.

A two-port network model of the ear canal and middle ear, based on the model in [94] and implemented as in [10], is also included. Model responses are computed in the time domain using a fast implementation of the modified state-space method [95]. More details of the state-space formulation and the introduced modifications can be found in appendix B.1, or chapter 3 of [10] and therefore are not repeated here.

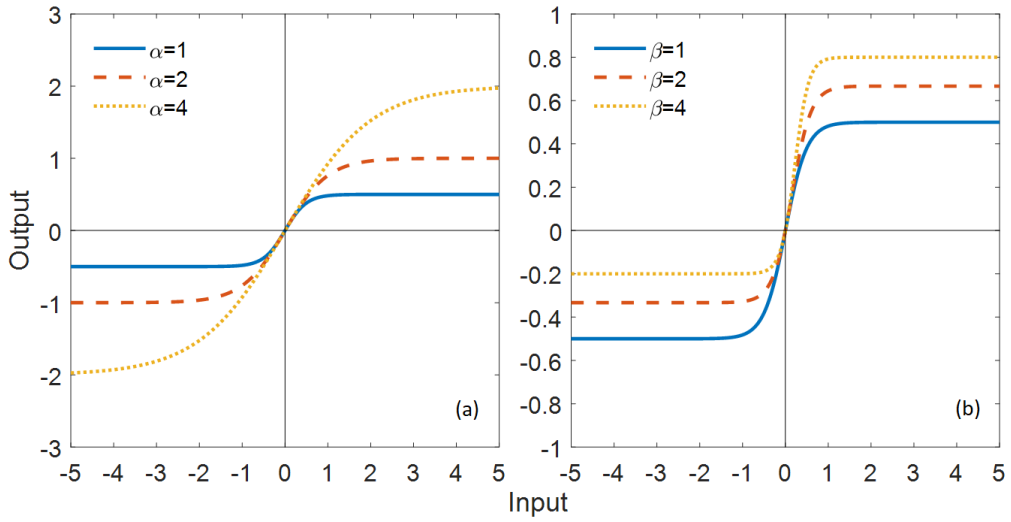


FIGURE 3.2: Some example Boltzmann functions with varying values of α and β. (a) Variation of the Boltzmann function with parameter $\aleph$, while β is set to 1. (b) Variation of the Boltzmann function with parameter β, while α is set to 1.

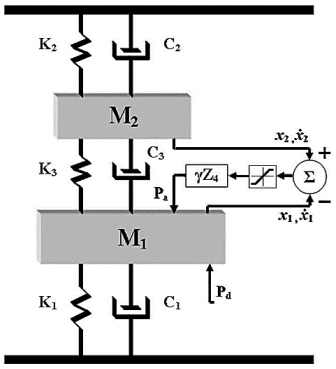


FIGURE 3.3: Block diagram of the micromechanical model of Neely and Kim [9] with a saturating nonlinearity in the active force. Adapted from Fig. 11 of [1] with permission.

3.2.2 The CAR-FAC Model

The CAR-FAC model [12] is based on using a cascade of local filters to simulate the local behavior of the forward travelling waves and the overall structure of the CAR-FAC model is shown in Fig. 3.4. A formal derivation of the underlying structure of the CAR-FAC model from the above TL model is presented in section 3.3.1. The cascade of asymmetric resonators models the forward hydromechanical wave propagation on the BM. The combination of the OHC module

and the automatic gain control (AGC) network is used to implement the fast acting compression, while the IHC model compute the mechanical-neural conversion. The OHC module represents the instantaneous, local feedback path that adapts the active undamping at various input SPLs, while the AGC module simulates the slower feedback that modulates the activity of the OHC module from more central stage of the auditory system. The outputs from asymmetric resonators represent the BM motion (with arbitrary scale), and those of the IHC module represent an estimate of the average instantaneous firing rates on the auditory nerve fibres. However, in this chapter only the outputs from the asymmetric resonators are considered, as the TL model does not have a mechanical-neural transduction module.

Each individual filter contains a complex-conjugate pole pair and a complex-conjugate zero pair at a somewhat higher frequency in the Laplace plane. Other pole-zero placements are also possible, such as the three-pole and two-zero filter section proposed by Kates [96]. The transfer function of each filter stage is thus a rational function of the Laplace transform variable s, of second order in both numerator and denominator [81]:

$$H(s) = \frac{s^2/\omega_z^2 + 2\zeta_z s/\omega_z + 1}{s^2/\omega_p^2 + 2\zeta_p s/\omega_p + 1} \tag{3.3}$$

where ω_p and ω_z are the natural frequencies and ζ_p and ζ_z are the damping ratios of the poles and zeros, respectively. The resulting frequency response has a unity gain at DC, a peak in the gain around the pole frequency, simulating the active amplification of the travelling wave and a dip around the zero frequency, corresponding to the rapid decay of energy after the characteristic place (CP), as shown in (C) of Fig. 3.5. The value of the peak gain is controlled by the resonator pole damping factor, with lower damping leading to higher gain. None of the filter sections are highly resonant, but the overall system achieves a large gain and sharp tuning from the contribution of many cascaded stages in an effect known as "pseudo-resonance" [97]. In other words, a change of only a few percent of the gain of each filter can alter the gain of the cumulative transfer function between

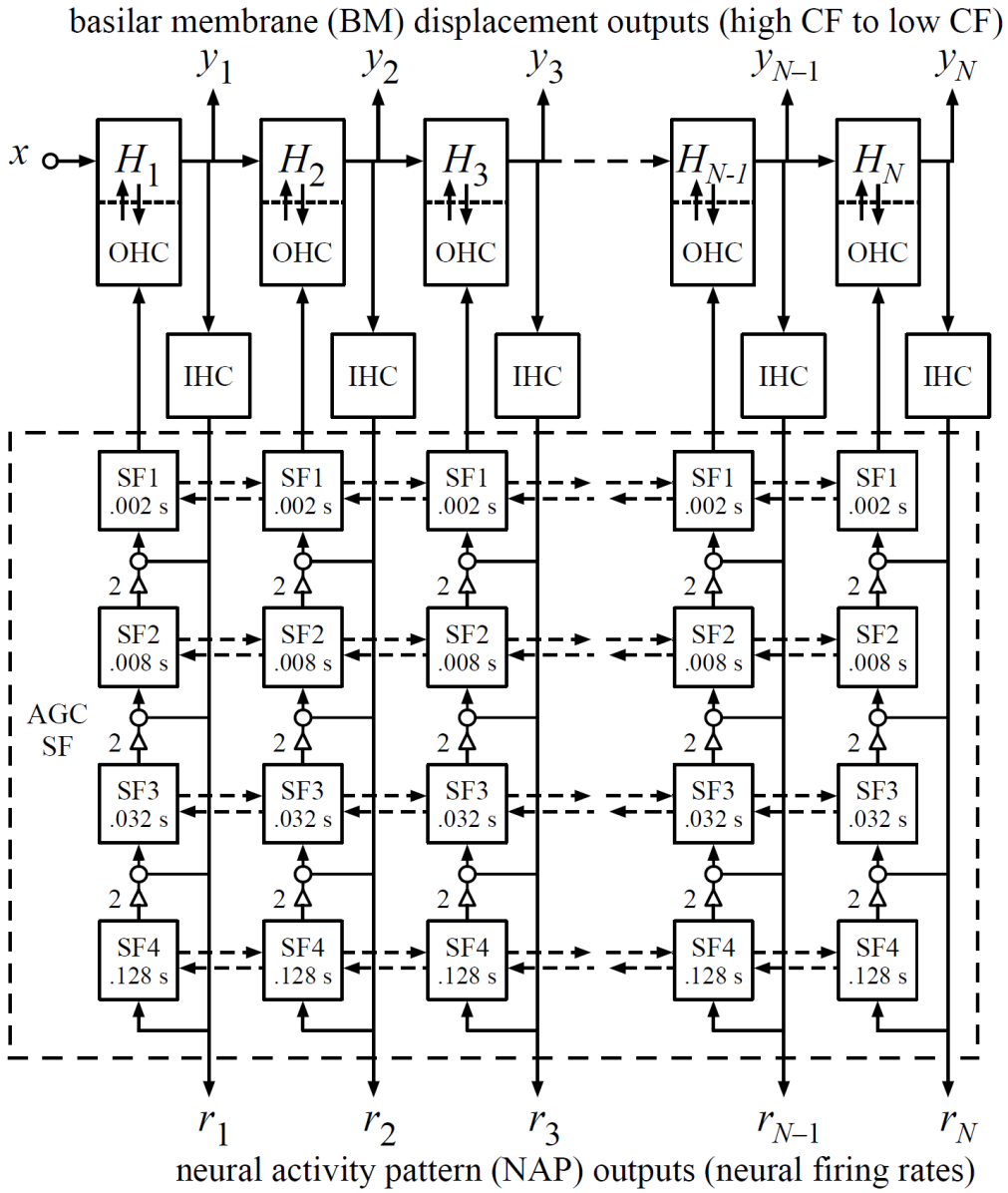


FIGURE 3.4: Architecture of the CAR-FAC model of the human cochlea. H_1 to H_N represent the transfer functions of the cascade of asymmetric resonators, modelling the BM motion with outputs y_1 to y_N. Damping parameters of these resonators are controlled by the OHC model and the coupled AGC smoothing filters (AGC SF), shown in dashed rectangular box, to implement fast-acting and level-dependent compression. The AGC module of each channel consists of a parallel-of-cascade of four first-order low-pass filters with increasing time constant. Smoothing filters with the same time constant are also laterally connected (dashed arrows), which allows a diffusion-like coupling across both space and time. The IHC model outputs, r_1 to r_N, present an estimate of average instantaneous firing rates on the auditory nerve fibres. Picture obtained by combining Fig. 15.2 and Fig. 19.5 of [12] with permission.

the stapes and cochlear partition by several orders of magnitude, achieving a wide range of dynamic gain control while not dramatically changing each serial filter's bandwidth and temporal resolution. The pole frequencies in the original implementation [98] are specified using the equivalent rectangular bandwidth scale with 50% bandwidth overlap between neighbouring sections, resulting in a total of 85 channels from 20 to 20,000 Hz. In this work, to better match cochlear mechanics, pole frequencies are determined as the CFs of the central locations of 85 equally divided elements along the length of the human cochlea, using the Greenwood map [31]

$$F = 165.4(10^{0.06(35-x)} - 0.88) \tag{3.4}$$

where F is the characteristic frequency of a position in the cochlea that is x mm from the base. This change simply introduces a shift in the peak frequency of the cumulative frequency response at each tap and does not influence the key properties of the CAR-FAC model. Finally, the continuous CAR is transformed into digital filters through pole-zero mapping using $z = \exp(s/f_s)$ [81], where f_s is the model sampling frequency.

To account for the observed cochlear nonlinearity, the damping ratios of pole and zero pairs of each section are dynamically adjusted in the CAR-FAC model in proportion to filter outputs, to turn up or down the gain of each serial filter, using an OHC module, in cooperation with a dedicated IHC module which is an adaptation of that proposed by Allen [99], and a AGC network. The AGC module of each section is a set of four first-order low-pass filters with increasing time constant organized in a parallel-of-cascades structure (see Fig. 3.4), which provides temporal smoothing to the outputs of the IHC module. The state of each low-pass filters is combined with those of its nearest basal and apical neighbors with the same time constant through weighted averaging. Such spatial smoothing and coupling allows the activity in one channel to reduce the gains of nearby channels, which helps to keep the gains of nearby channels in a similar range, which in turn helps to preserve local spectra contrasts. This spatially distributed feedback gain control is also beneficial for simulating lateral inhibition [100] and two-tone suppression. A comprehensive description of the CAR-FAC model and

its software implementation can be found in [12] and [98], respectively. In the following sections, all of the simulations with the CAR-FAC model are performed using the default parameter values specified in [98], except that the model sampling rate is increased from 22.05 kHz to 100 kHz in order to be the same as that of the TL model.

3.3 Model Comparisons

3.3.1 Derivation of the Filter Cascade Model from the TL Model

Following the long-wave approximation, the one-dimensional wave equation for the 1D box model shown in Fig. 3.1 can be derived using the principles of conservation of mass and momentum [77] as,

$$\frac{\partial^2 p(x,t)}{\partial x^2} = \frac{-2\rho}{h}\frac{\partial v(x,t)}{\partial t} \tag{3.5}$$

where x is the longitudinal directional along the cochlea, $p(x,t)$ is the pressure difference across cochlear partition, $v(x,t)$ is the vertical velocity of the cochlear partition, ρ is the fluid density and h is the effective height of the cochlear chamber. A finite difference approximation of this equation is used in the numerical implementation of the TL model, but its responses can also be derived analytically [77]. If the BM dynamics are assumed to be linear and characterized by the locally-acting impedance, Z_{cp}, Eq. (3.5) can be rewritten in the frequency domain as,

$$\frac{\partial^2 p(x,\omega)}{\partial x^2} + k^2(x,\omega)p(x,\omega) = 0 \tag{3.6}$$

where $p(x,\omega)$ is the pressure difference in the frequency domain, $k(x,\omega)$ is the wavenumber of cochlear travelling wave, which is related to the cochlear partition

impedance as the following,

$$k^2(x,\omega) = \frac{-2i\omega\rho}{hZ_{cp}(x,\omega)} \tag{3.7}$$

If the model parameters vary slowly along the longitudinal direction, such that the relative change of the wavenumber over one wavelength is not too large [77], the WKB approximate solution [101] to Eq. (3.6) for the forward traveling waves can be written as,

$$p(x,\omega) = p_0\ k^{-1/2}(x,\omega)\ e^{-i\Phi(x,\omega)} \tag{3.8}$$

where p_0 is an amplitude factor determined by the excitation and boundary condition and $\Phi(x,\omega)$ is the phase integral, expressing the cumulative phase at position x,

$$\Phi(x,\omega) = \int_0^x k(x',\omega)\,\mathrm{d}x' \tag{3.9}$$

The vertical velocity of the cochlear partition can then be expressed as,

$$v(x,\omega) = \frac{-p(x,\omega)}{Z_{cp}(x,\omega)} = v_0\ k^{3/2}(x,\omega)\ e^{-i\Phi(x,\omega)} \tag{3.10}$$

where v_0 is related to p_0 by $v_0 = ip_0H/2\omega\rho$. The frequency response of the BM motion for a forward-travelling wave in a longitudinal segment of the cochlea, extending from x_0 to $x_0 + \Delta$, can thus be written as,

$$H_\Delta(x_0,\omega) = \frac{v(x_0+\Delta,\omega)}{v(x_0,\omega)} = \left(\frac{k(x_0+\Delta,\omega)}{k(x_0,\omega)}\right)^{3/2} e^{-i\int_{x_0}^{x_0+\Delta}k(x',\omega)\,\mathrm{d}x'} \tag{3.11}$$

The responses at a sequence of positions along the cochlear partition can then be regarded as the outputs of a cascade of filters with individual frequency response corresponding to that given by (3.11). In [79], however, only the exponential of the phase integral in (3.11) was considered, and the wavenumber ratio (WNR) factor, defined below, was ignored,

$$WNR = \left(\frac{k(x_0+\Delta,\omega)}{k(x_0,\omega)}\right)^{3/2} \tag{3.12}$$

Since the complex wavenumber changes along the length of the cochlea, this term contributes to variations in both magnitude and phase. The influence of neglecting this WNR term is investigated by computing two sets of magnitude and phase responses from the linear and fully active TL model, one with and the other without the WNR, as shown in panels (A) and (B) in Fig. 3.5 as thin and thick solid lines respectively. The 35 mm length of the cochlear partition has been uniformly divided into 85 sections, which is the number of channels in the CAR-FAC. The wavenumber ratio term in Eq. 3.11 can be readily computed from the impedance values at the starting and ending locations of each section. In order to obtain the phase integral in Eq. 3.11, each section is further divided into 10 subsections, so that the phase integral is computed numerically using the trapezoidal rule. This operation effectively uses 850 discrete elements to represent the 35 mm length of the cochlear partition. In single-tone and two-tone simulations, however, only 500 elements are used. Every fifth of the frequency responses of the first 76 sections are shown in Fig. 3.5 for better visualization. In the TL model, it can be seen that, the peak gains at basal sections are higher than those at more apical ones. But if the WNR is neglected, the gain in the pass-band of each section is significantly reduced, especially at basal sections. For comparison, panels (C) and (D) in Fig. 3.5 show the frequency responses of the cascaded filters of the CAR-FAC model, which are also configured as fully active and with the maximum gain. Unlike the TL model, each serial filter in the CAR-FAC model has a smoother high-frequency roll-off and roughly an equal amount of peak gain, which is about 7 dB. This peak gain is similar to that of the individual segment of the TL model with the WNR term, except at a few basal sections. Thus the WNR term has dramatic effects on the behavior of the TL model, greatly increasing its responses at low input levels. The influences of this WNR term can be largely compensated for by decreasing the damping, or equivalently turning up the gain, of each section in the CAR-FAC, as is already done in the default software implementation. Thus although the CAR-FAC filters were originally motivated only by the phase term in Eq. 3.11, it is clear that the wavenumber ratio term is also necessary so that the cascade filters show a more accurate representation of the cochlear responses,

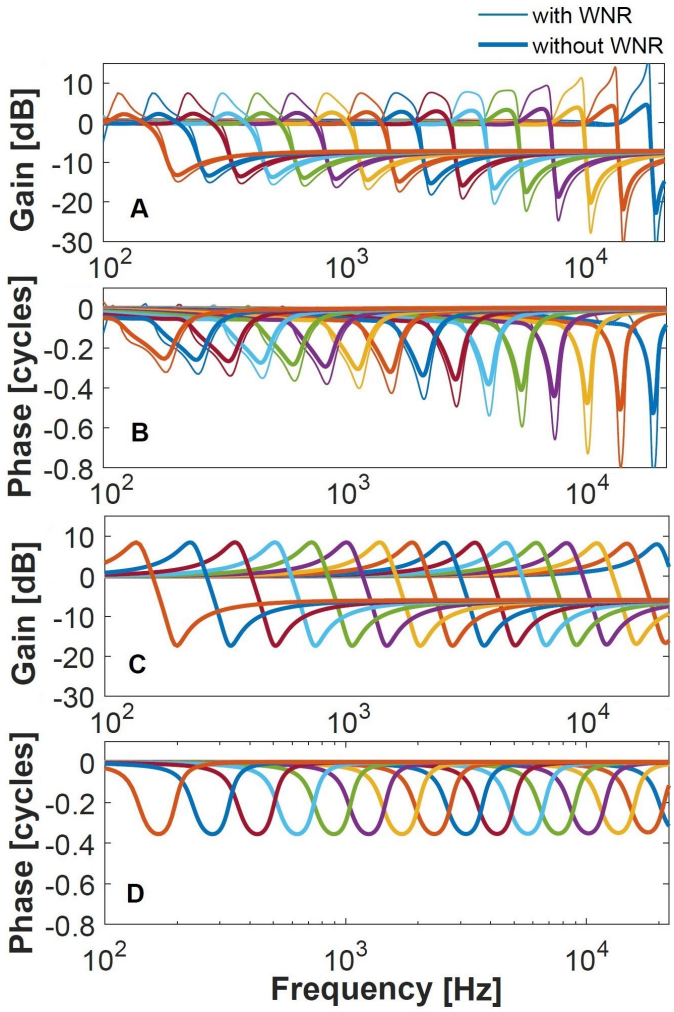


FIGURE 3.5: Frequency responses of the individual segments of the TL model (A, B), computed with (thin) and without (thick) the WNR term and those of the serial filters of the CAR-FAC model (C, D). Only every fifth of the first 76 output channels are shown for clarity.

mainly in terms of providing sufficient active gain.

3.3.2 Cumulative Frequency Responses and Active Gain

The cumulative frequency responses at a specific location or channel of the TL (with WNR) and CAR-FAC models are shown in panels (A, B) and (C, D), respectively in Fig. 3.6, when both models are linear and are either entirely passive or entirely active. The frequency responses of every fifth of the first 76 channels are shown for better visualization. Fully active model responses ($\gamma = 1$ in the TL model and minimum damping in the CAR-FAC model) are shown as thin lines

while those for the fully passive model ($\gamma = 0$ in the TL model and maximum damping in the CAR-FAC model) as thick lines. It can be seen that the sharpness of tuning along the longitudinal dimension in the TL model gradually decreases from the base to the apex, providing a better simulation of physiological results in [33] than the CAR-FAC model which shows almost constant sharpness of tuning.

The active gain provided in the cochlear travelling waves at different locations along the length of both models is shown in Fig. 3.7, computed as the difference in dB in the cumulative magnitude responses at the corresponding CF between the fully active and fully passive model settings. Channels of the CAR-FAC model are converted to the equivalent positions along the cochlea according to their CFs using the Greenwood map for humans, given in Eq. (3.4). At basal locations in both models, the cumulative transfer functions have lower gains than at more apical positions, since there is only a small number of filters to cascade. As the travelling waves reach the middle region of the cochlea, the peak gains at CF are greatly increased and the active gains reach their maximum values, due to the contribution of larger number of cascaded sections. In the apical area, the damping factors in the CAR-FAC model are gradually increased, compared to those at more basal places, to reduce the effect of the cochlear amplifier in this region. Thus both models are in reasonable agreement with the wide range of experimental observations (20 dB to 70 dB) from different animals [33], in that the active gain is greater at the base than at the apex. Although the detailed distributions of the active gains along the cochlea are different between the two models, there is currently not enough experimental evidence to justify which one is more appropriate than the other.

3.3.3 Single-tone compression

In simulations of the nonlinear responses to single tones, the 4 kHz characteristic place of the cochlea was taken for both models as an example for comparison of their responses. All input stimuli had a duration of one second, with a 10 ms half-Hanning window onset ramp. The spectral component at the stimulus frequency

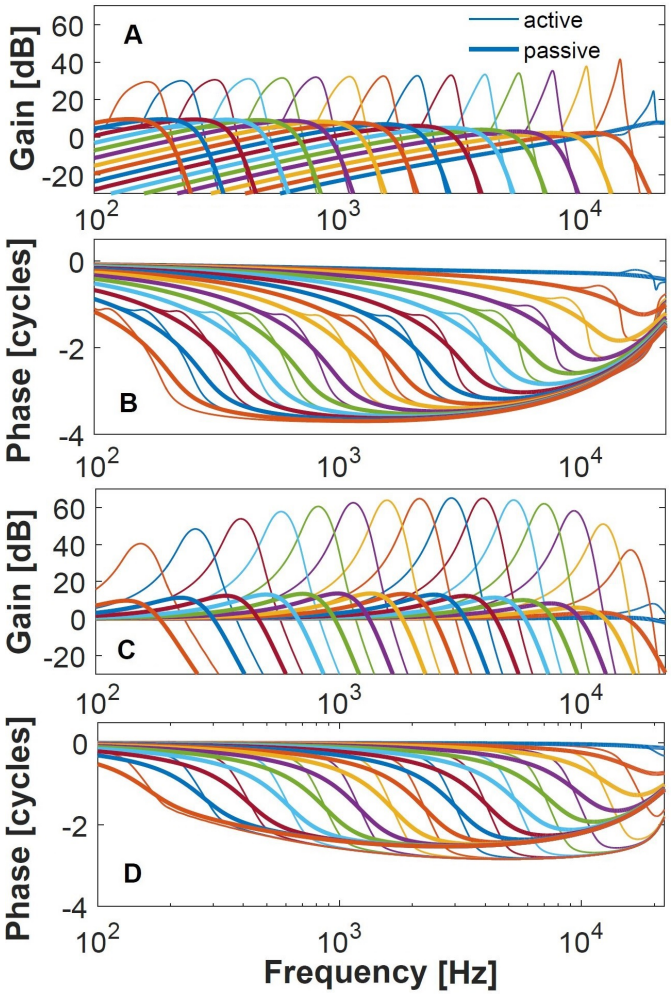


FIGURE 3.6: Overall frequency responses at various positions for the TL model (A, B) and CAR-FAC model (C, D) when configured as fully active (thin lines) and fully passive (thick lines). Only every fifth of the first 76 output channels are shown for clarity.

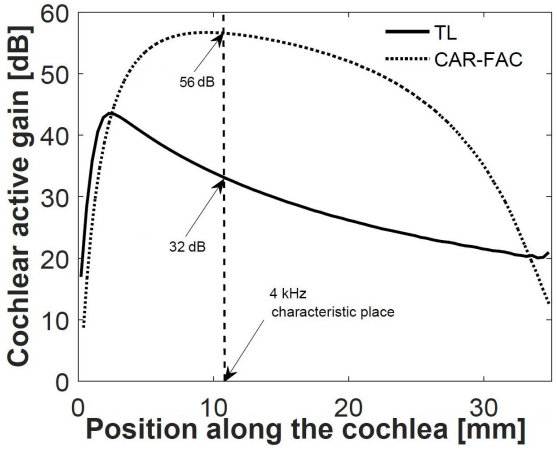


FIGURE 3.7: Active gain provided to the travelling waves at CF in the TL and CAR-FAC models for different positions along the length of both cochlea models.

was obtained by a Fourier analysis of the last $0.5s$ of the time domain solution. Fig. 3.8 shows the compressive growth of the BM motion to sinusoidal tones with different frequencies and intensities in the TL and CAR-FAC [98] models. The responses of the TL model are shown in the first column (A, B, C, D) and those of the CAR-FAC model in the second column (E, F, G, H).

Panels (A) and (E) show the BM input/output (I/O) functions, in the TL and CAR-FAC respectively, for some pure tones with frequencies ranging from 0.5 kHz to 7 kHz, and with levels ranging from 0 to 120 dB SPL in steps of 5 dB. The frequencies of these pure tones can be found in the legend of panel (G). The solid black line in these plots represents a linear growth function. For low-level stimuli at the characteristic frequency, 4 kHz, the response was amplified by $\sim$30 dB in the TL and $\sim$50 dB in the CAR-FAC as indicated in Fig. 3.7. It was then highly compressed until $\sim$100 dB SPL, after which both models behaved linearly. For stimuli that are higher and lower in frequency than 4 kHz, the TL model responded almost linearly over the entire range of levels tested, except for the ones that are very near to the CF (e.g. 3-4.5 kHz). In the CAR-FAC model, although low frequency stimuli showed a similar linear response pattern, the high frequency responses also grew compressively from 0 dB to 100 dB SPL, even at frequencies that are almost an octave above the CF, as shown more clearly in panels (G, H).

Panels (D) and (H) show the BM sensitivity curves as a function of stimulus frequency with increasing input SPLs, which is represented by increasing line thickness, with the thinnest line corresponding to 0 dB, thickest line to 100 dB and the step between two neighbouring lines is 10 dB SPL. BM sensitivities were obtained by normalizing the BM responses by the corresponding stapes velocity and sound pressure in the TL and CAR-FAC model respectively. As can be seen, BM sensitivity around the CF was reduced by $\sim$30 dB and more than 50 dB in the TL and CAR-FAC model respectively, as sound pressure level was increased from 0 dB to 100 dB, broadening the BM tuning in both models. However, compared to the CAR-FAC, the TL model is more sharply tuned and has a higher rate of high frequency roll-off. The sensitivity for stimuli having frequencies lower and higher than the CF remain almost constant in the TL model regardless of the stimuli

SPLs, while this is only the case for frequencies that are well below the CF in the CAR-FAC model. Finally, in the filter cascade model, the most responsive stimulus frequency, which is 4 kHz at low levels, smoothly shifts to 3.4 kHz at 100 dB SPL, equivalent to a decrease of ∼0.23 octave relative to the CF. But in the TL model, this shift does not appear until ∼60 dB SPL, and the shift at 100 dB SPL amounts to ∼0.68 octave lower than the CF.

To quantify the extent of compression to single tone stimuli in both models, the rate-of-growth (ROG) functions, computed in dB/dB as the slope of the I/O functions shown in panel (A, E), are shown in two different manners in panels (C, G) and (D, H), respectively. Panels (C) and (G) mainly illustrate the variation of ROG with probe level. In the TL model, CF response started linearly at the lowest SPL and became compressive at ∼20 dB. Further increasing the CF level gradually decreased the ROG to the minimum of ∼0.4 dB/dB at ∼50 dB, after which the ROG function gradually returned back to being linear. Responses to all the other example stimuli grew roughly linearly over the range of 120 dB SPL. In the CAR-FAC model, however, responses to stimuli at CF and higher frequencies were compressive even at at the lowest SPL tested, 0 dB. As stimulus level was raised, the growth rates also decreased, stablizing at about 0.3 dB/dB in the range 30 dB to 60 dB SPL. At even higher input levels, the response growth to CF tones gradually became linear, while the responses to higher frequency pure tones became expansive. Panels (D) and (H) show more clearly the dependence of ROG on stimulus frequency with increasing stimulus level. As already shown above, the TL model shows compressive growth of responses to pure tones that are within a small region around the CF. In the CAR-FAC model, stimuli with frequencies that are well above and below the CF are also highly compressed, especially ones with higher frequency than the CF. In fact, compression was almost to the same extent of the CF tones at low to mid SPLs. Although this is in disagreement with the physiological measurements [33], the gains to these high frequency tones are significantly lower than CF tones, and therefore is not expected to make a large impact to the overall response.

Physiological measurements of mammalian cochleae from different studies and across various species show a high degree of variation in terms of compression range and rate. Some experiment [102] measured compressive behavior at levels as low as 0 dB SPL in the chinchilla, while others [103, 104] did not observe compression until the levels were above 50 dB SPL in the guinea pig. The growth rate of the BM response in the compressive region also varies from ∼0.2 to ∼0.5 dB/dB [33]. Thus, the responses of the TL and CAR-FAC models can be regarded as quantitatively consistent with the wide range of experimental data in terms of compression start SPL and compression rate. For the reduction in peak frequency of the normalized BM transfer function with increasing input intensity, physiological measurements show a decrease of 0.36-0.51 octave relative to CF [38]. Thus both models only agree qualitatively with these measurements, as the TL exhibited a larger shift, while the CAR-FAC model showed a smaller shift in peak frequency with stimulus level. Furthermore, compressive responses have been observed for single tone stimuli with frequencies that are around 30% higher and lower than the CF in the chinchilla cochlea [38]. Both models are also not very accurate in reproducing this, since the effective bandwidth of its "cochlear amplifier" is too limited in the TL model and too broad in the CAR-FAC model, especially for high-side (relative to the CF) pure tones.

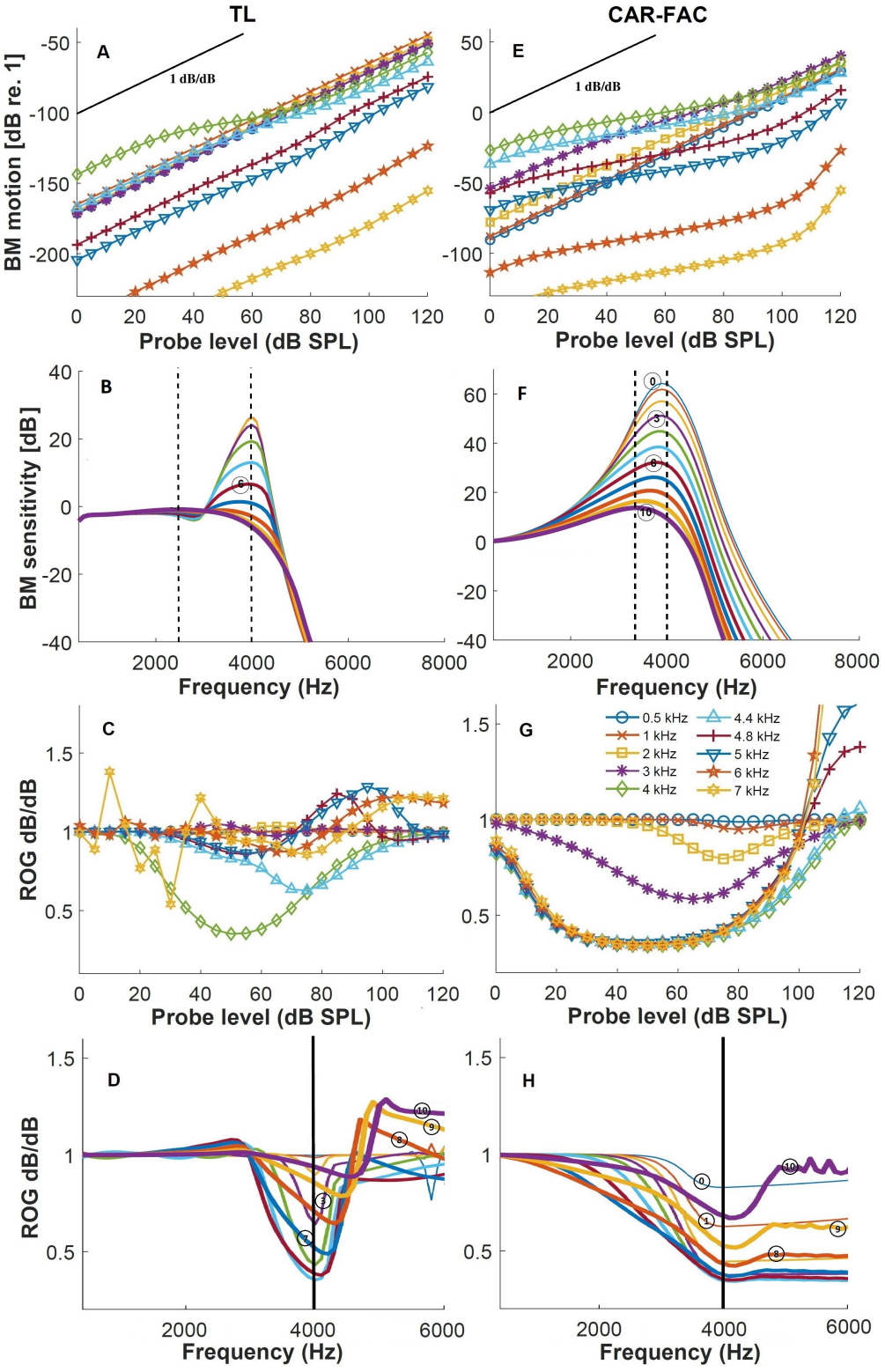

TL
CAR-FAC
A
E
1 dB/dB
1 dB/dB
BM motion [dB re. 1]
Probe level (dB SPL)
Probe level (dB SPL)
B
F
BM sensitivity [dB]
Frequency (Hz)
Frequency (Hz)
C
G
0.5 kHz
1 kHz
2 kHz
3 kHz
4 kHz
4.4 kHz
4.8 kHz
5 kHz
6 kHz
7 kHz
ROG dB/dB
Probe level (dB SPL)
Probe level (dB SPL)
D
H
ROG dB/dB
Frequency (Hz)
Frequency (Hz)

FIGURE 3.8: Single-tone compression simulations at the 4 kHz characteristic place in the TL (A, B, C, D) and CAR-FAC (E, F, G, H) model. (A, E) BM motion I/O functions at a number of example frequencies, including the CF, 4 frequencies that are lower and 3 that are higher than the CF. The straight line indicates linear growth rate of 1 dB/dB for easy reference. (B, F) Sensitivity functions obtained by normalizing the BM responses in (A, E) by the corresponding stapes velocity in the TL model and stimulus sound pressure level in the CAR-FAC model. Stimulus intensity varied from 0 dB to 100 dB in steps of 10 dB and is indicated by the thickness of the lines with the thinnest line corresponding to 0 dB and the thickest line to 100 dB. Numbers inside the circle symbol represent probe SPL divided by 10. The vertical lines indicate the place of the CF and the most sensitive frequency at the highest probe level, 100 dB SPL. (C, G) Rate-of-growth (ROG) values are shown as a function of stimulus level for the same example frequencies as in (A) or (E). (D, H) Rate-of-growth (ROG) values are shown as a function of stimulus frequency for some example SPLs. Line style and marker meaning in (D, H) follow that in (B, F).

3.3.4 Two-tone suppression

The effects of two-tone suppression from a mechanical perspective on the BM have been demonstrated in a number of studies [40, 43, 105, 41, 42, 106, 45, 107], although the measurements are not always consistent with each other. Ruggero *et al* [40] measured suppression of the overall velocity response of the BM with both high-side and low-side suppressors, while later experiments [105, 41, 42] only observed reductions in the overall response for high-side suppressors. Low-side suppressors can reduce the spectral component at the probe frequency, but never the overall response amplitude. In addition to this "tonic" suppression (reduction in the average response at the probe tone frequency), they also cause "phasic" suppression where the probe tone response is suppressed in a manner that varies with the phase of the low-side suppressor [41, 42, 45]. In this work, we adopt the definition of two-tone suppression used in [105, 41, 42, 106, 45, 107]; as a reduction in the magnitude of the spectral component at the probe frequency during the simultaneous presentation of a suppressor tone. We considered the suppression of a 4 kHz tone at its characteristic place along the cochlea. Suppressor tones were used with frequencies from 100 Hz to 10,000 Hz, in steps of 100 Hz (except

at 4 kHz), and levels from 10 dB to 90 dB SPL, in steps of 5 dB. Each input stimulus had a length of one second, but only the last 0.5 seconds of the time-domain model outputs were used to extract the spectral component at the probe frequency. Finally, only tonic suppression was investigated and phasic suppression within these two models is left for future work.

The level dependent suppression of a 4 kHz probe tone, presented at 30 dB SPL, is investigated for different suppressor frequencies and Fig. 3.9 shows a subset of results in panels (A) and (C) for the TL and CAR-FAC models respectively. The degree of suppression was quantified as the ratio, in dB, of the magnitude responses at the probe frequency in the presence and absence of the suppressor, so that 0 dB means no suppression and suppression is present when this ratio is negative. Both low- and high-side suppressors are capable of reducing the probe tone response, but the CAR-FAC generates greater suppression at almost all levels of each suppressor and the suppression effect appears at lower suppressor levels. The maximum amount of suppression is ∼30 dB in the TL model and ∼35 dB in the CAR-FAC model, which would probably continue to increase with even higher suppressor levels. Both models are capable of reproducing the general trend that low-side suppressors produce higher amounts and rates of suppression than high-side suppressors. However, in the TL model, the influence of high-side suppressors drops off considerably quicker than that in the CAR-FAC model, with increasing separation of the probe and suppressor frequencies.

Two-tone neural suppression in the auditory nerve has been extensively studied [39]. One typical method to determine the amount of suppression is to calculate the right-ward shift of AN rate-intensity functions at different suppressor intensities. This paradigm has also often been adopted in BM two-tone suppression studies [106, 45]. Panels (B) and (D) of Fig. 3.9 show the input/output level curves for a 4 kHz probe tone at its characteristic place in the absence and presence of a high-side (5 kHz) suppressor tone at 5 different SPLs. The horizontal shift in the BM I/O functions is seen in both models, resulting in an increasingly reduced compressive region of the probe tone growth curve at low-to-medium level as the intensity of the suppressor tone is raised. For high-level probe tones, however, the

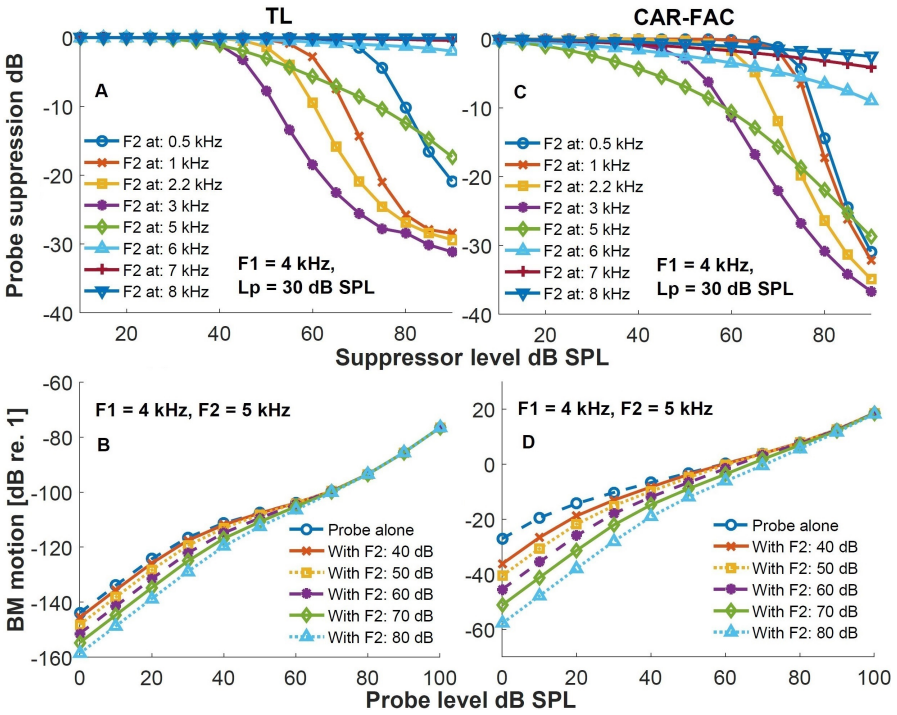


FIGURE 3.9: Level dependence of two-tone suppression to a 4 kHz probe tone (F1) at its characteristic place on suppressor tone frequency using the TL (A) and CAR-FAC (C) model. The probe tone sound pressure level is 30 dB. Input/output functions for a 4 kHz probe tone (F1) at its characteristic place in the absence and presence of a high-side, 5 kHz, suppressor tone (F2) at 5 different sound pressure levels as indicated in the legend, in the TL (B) and CAR-FAC (D) model.

I/O function is almost linear and hardly affected by the suppressor. These results are in good qualitative agreement with those measured experimentally, e.g. Fig. 2A of [41].

To quantify the differences between these two models, three perspectives of two-tone suppression were computed from the complete simulation dataset and are presented in columns 1, 2, and 3 in Fig. 3.10 for the CAR-FAC and in Fig. 3.11 for the TL model, respectively. The CF probe was presented at three different levels, 30, 50 and 70 dB SPL, and the corresponding simulation results are shown in rows 1, 2, and 3. Figures were plotted following the style adopted in Fig. 4 of [106], which is also reproduced here in Fig 3.12 for easy comparison.

The first column (A, B, C) shows the iso-level suppression curves with suppressor level increasing from 10 dB to 90 dB SPL, in increments of 5 dB. The intensity of

the suppressor is also coded by line thickness, with the thinnest corresponding to 10 dB SPL and the thickest to 90 dB SPL. In both models, the frequency extent of suppression is greatly expanded for all low-side suppressors with increasing suppressor level, while for high-side suppressors, their effects quickly decrease with increasing frequency separation with the probe tone regardless of their levels, which is in reasonable agreement with experimental measurements [106, 45]. The amount of suppression exhibited in the CAR-FAC model is larger than that shown in the TL model mainly because of the difference in active gain between the two models, as discussed in section 3.4. Both models introduce lower level of suppression than those measured in recent experiments in very sensitive cochleae [45], where suppression amplitudes of more than 40-50 dB are observed for some very low frequency suppressors (see Fig. 3(a) and 4(a) of [45]). However, the CAR-FAC model produces responses that are very similar to the measurements of an earlier experiment [106] (see panels (A, B, C) in Fig. 3.12). Additionally, in the TL model, very low frequency suppressors (100, 200, 300 Hz) almost have no effect in reducing the probe tone responses and the influence of high-side suppressors is negligible when their frequencies are higher than around 6 kHz, equivalent to $\sim$0.59 octave above the probe. This is less realistic compared to the CAR-FAC model as physiological experiments [42] have shown that low frequency suppressors are nearly as equally effective and high-side suppression is still observable when the suppressor frequency is nearly an octave above the probe frequency [106]. The reduction in two-tone suppression at 2 kHz in the CAR-FAC and at 0.8 kHz, 2 kHz in the TL model when the probe tone is at 30 dB SPL is due to harmonic distortions generated at the probe frequency, i.e. 4 kHz, by the suppressor tone.

The second column (D, E, F) shows the suppressor threshold levels for 1, 10 and 20 dB reduction of the probe response, when the probe is at 30, 50 and 70 dB SPL. The CAR-FAC model is closer to experimental measurements than the TL model for low-side suppression because the suppression threshold is nearly constant in the low frequency region [106]. However, the TL model predicts better the 1 dB threshold curve in the region around the probe frequency when the probe is at the lowest level, 30 dB SPL. As the level of the probe tone increases, the shapes

of the threshold curves tend to lose their 'tips' (see Fig. 3.12 reproduced from [106]), but those produced by the two models remain roughly unchanged at all three probe levels. Furthermore, both models show suppression thresholds that are higher than biological observations in nearly all combinations of probe and suppressor tones simulated [106].

The third column (G, H, I) shows the rate of suppression (ROS) values as a function of suppressor frequency, computed as the slope of the probe reduction curves shown Fig. 3.9 (A) and (C). Only every 10 dB increment in suppressor level is plotted for better visualization. The influence of harmonic distortions is also visible from these plots, reducing the values of ROS and even causing an increase of the probe spectral component when the suppressor is at very high levels, such as 80 dB and 90 dB SPL. The maximum ROS results across all suppressor levels are plotted in Fig. 3.13 for both models. Since experimental measurements, i.e. Fig. 9 of [45], have shown that the maximum ROS can be as high as $\sim$2 dB/dB for low-side suppression, the CAR-FAC model can be considered as more realistic than the TL model in this aspect.

3.4 Discussion

It should be emphasized that the simulation results presented in this work are based on the sets of model parameters described in section 3.2, and with different sets of model parameters, the results would change to some extent. The aim of this initial study is to objectively evaluate the performance, with the current parameter settings, of two types of cochlear models in simulating experimentally observed single-tone and two-tone responses. It would be interesting to further explore whether the limitations in their responses are due to the particular choices of model parameters, or due to more fundamental issues associated with the structures of the two models, which is beyond the scope of the current investigation.

One obvious drawback of the TL model is that it cannot produce the same level of active gain in the travelling waves as often measured in sensitive cochleas (see

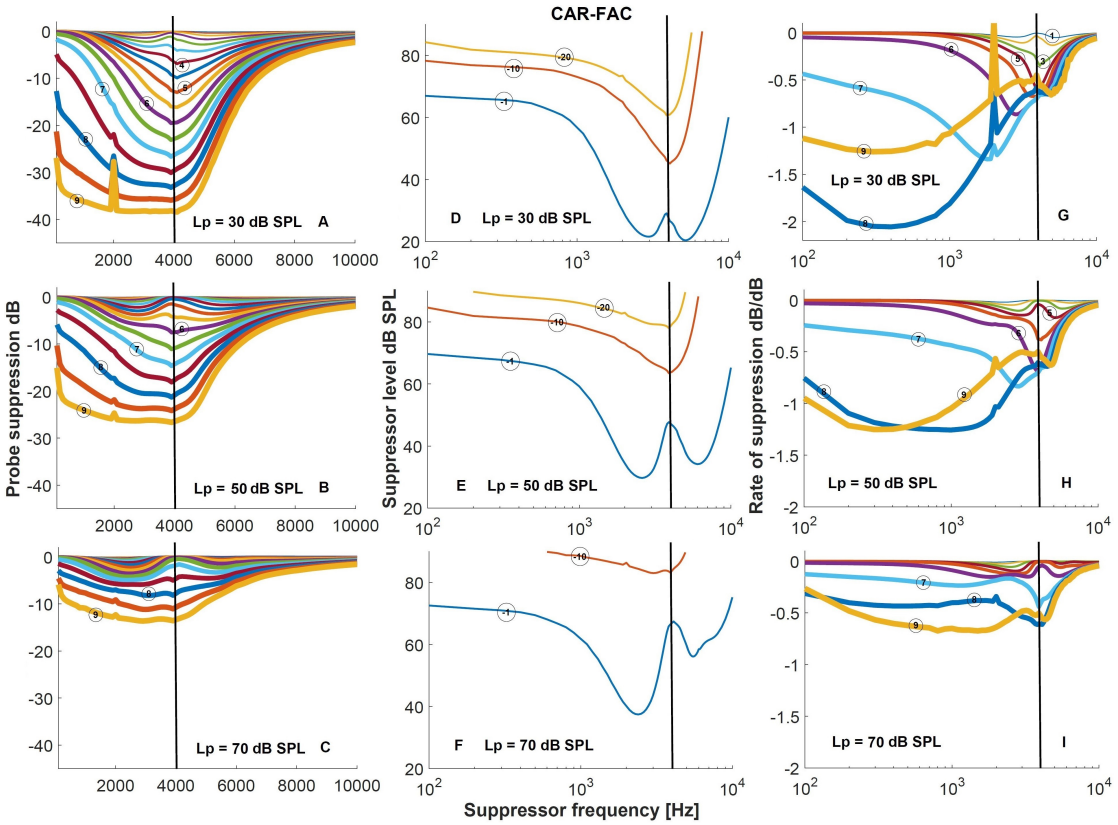


FIGURE 3.10: Two-tone suppression simulations using the CAR-FAC model. Results are displayed from three perspectives. Column 1 (A, B, C): iso-level suppression curves for suppressors ranging from 10 to 90 dB SPL in steps of 5 dB, as indicated by the line thickness, while the probe is at 30, 50 and 70 dB SPL in rows 1, 2 and 3 respectively. 10 dB suppressor levels are also marked with numbered circles: e.g., 30 dB SPL=10 times ③. Column 2 (D, E, F): suppressor levels necessary to reduce the probe amplitude by 1, 10 and 20 dB, as indicated by the encircled numbers on each line. Column 3 (G, H, I): Rate of suppression (ROS) as a function of suppressor frequency, computed as the slope of the probe reduction curves shown in Fig. 3.9 (C). Only every 10 dB increment in suppressor level is plotted for better visualization. 10 dB suppressor levels are also marked.

Fig. 3.7). One method of increasing the active gain in the TL model is to increase the dimensionless gain factor, γ, in the micro-mechanics. But this cannot be used here, since even a slight increase in γ will causes the TL model to become unstable. A similar limit on the active gain due to instability is also seen in other TL models with different micro-mechanical implementations [83, 64, 65, 9]. In this respect, the tuning of the CAR-FAC model seems to be more flexible, as system stability is not a concern as long as poles of the individual filters are constrained to have negative real parts.

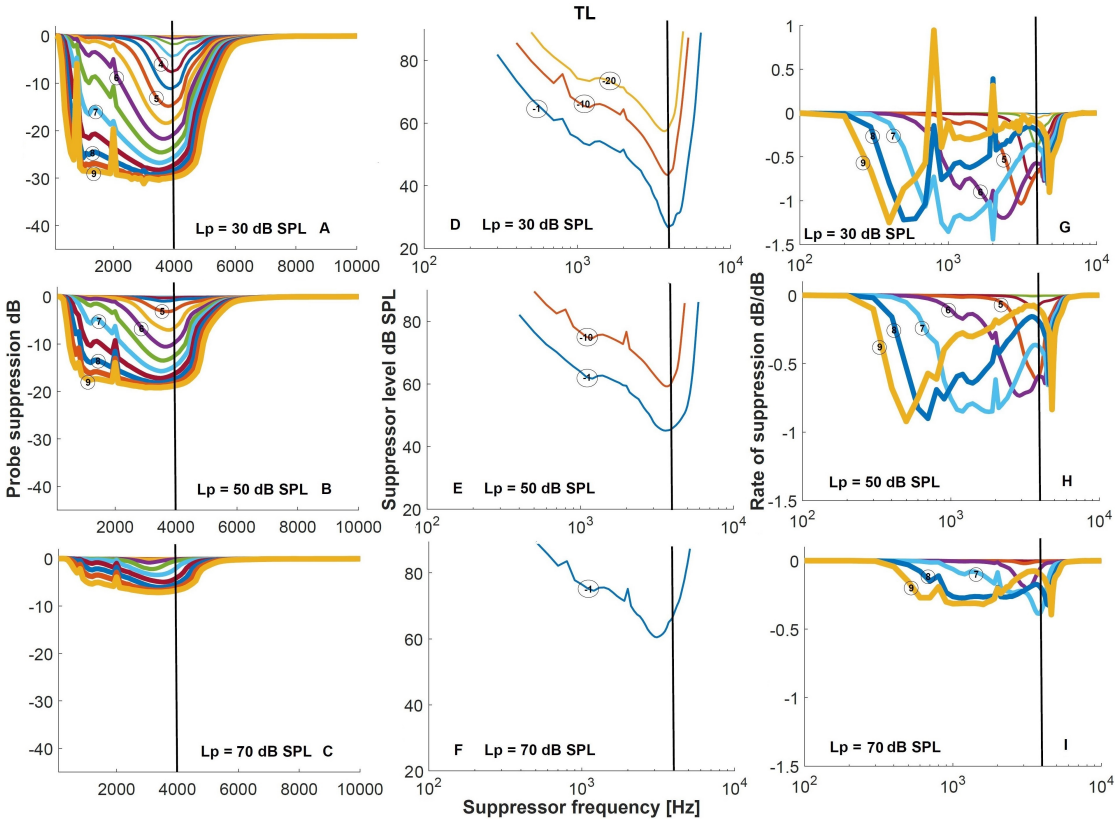


FIGURE 3.11: Two-tone suppression simulations using the TL model. Results are displayed from three perspectives. Column 1 (A, B, C): iso-level suppression curves for suppressors ranging from 10 to 90 dB SPL in steps of 5 dB, while the probe is at 30, 50 and 70 dB SPL in rows 1, 2 and 3 respectively. Column 2 (D, E, F): suppressor levels necessary to reduce the probe amplitude by 1, 10 and 20 dB. Column 3 (G, H, I): Rate of suppression (ROS) as a function of suppressor frequency, computed as the slope of the probe reduction curves shown in Fig. 3.9 (a). Only every 10 dB increment in suppressor level is plotted for better visualization. Line styles follow those in Fig. 3.10.

The limited active gain in the TL model partly explains the lower amount of two-tone suppression and lower suppression rates shown in Fig. 3.11, as the maximal amount of suppression is given by the active gain provided to a probe tone at its CP. For instance, at the 4 kHz CP, the active gain to a 4 kHz probe tone at 30 dB SPL is around 30 dB as illustrated in Fig. 3.8 and Fig. 3.9. The amount of suppression of this probe tone at the highest suppressor level is also around 30 dB, as shown in Fig. 3.11. Although the CAR-FAC model can produce much higher active gain (∼56 dB) at the 4 kHz CP, it does not introduce much larger suppression of the probe tone than the TL model, which is a lower amount than that measured experimentally in very sensitive cochlea [45]. This is because the

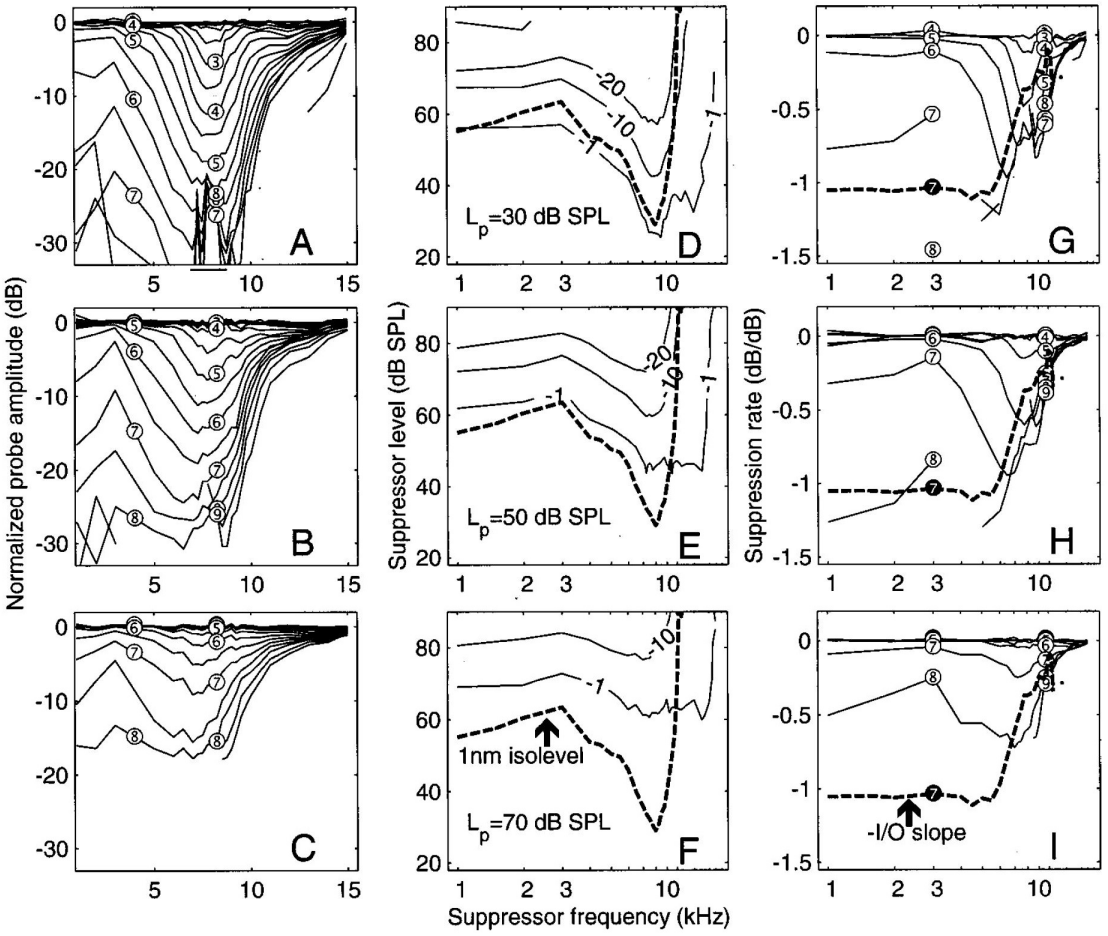


FIGURE 3.12: Two-tone suppression measurements from the basal region of the chinchilla cochlea. Results are displayed from three perspectives. Column 1 (A, B, C): iso-level suppression curves for suppressors ranging from 10 to 90 dB SPL in steps of 5 dB, while the probe is at 8 kHz and 30, 50 and 70 dB SPL in rows 1, 2 and 3 respectively. 10 dB levels are indicated with numbered symbol: e.g., 30 dB SPL=10 times ③. Column 2 (D, E, F): suppressor levels necessary to reduce the probe amplitude by 1, 10 and 20 dB (labelled as solid lines). The 1-nm iso-amplitude curve for a single tone is repeated in each panel (dashed line). Column 3 (G, H, I): Rate of suppression (ROS) as a function of suppressor frequency, computed as the slope of the probe reduction curves. Only every 10 dB increment in suppressor level is plotted for better visualization (solid lines and numbered symbols). The negative of the slope of single-tone I/O curves at 70 dB SPL is superimposed in each panel (dashed line), where-1 dB/dB is linear and 0 dB/dB is complete compression. Taken from Figure 4 of [106] with permission.

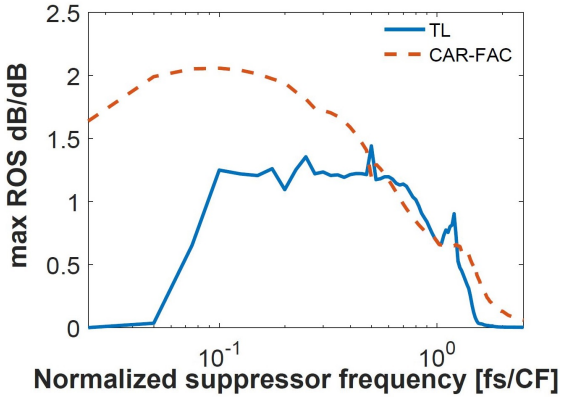


FIGURE 3.13: Maximum rate of suppression as a function of suppressor frequency for a 4 kHz, 30 dB SPL, probe tone at its characteristic place in the TL and CAR-FAC model. Note that the sign of ROS values is inverted compared to those in Fig. 3.10, Fig. 3.11 and Fig. 3.12 to facilitate comparison with experimental measurements shown in Fig. 9 of [45].

sensitivity of each channel of the CAR-FAC model starts to decrease at the lowest level tested, 0 dB SPL, such that when the probe tone is at 30 dB SPL, the active gain is reduced to ∼40 dB (Fig. 3.9 (F)), which is roughly consistent with the amount of two-tone suppression observed in Fig. 3.11 (B). In other words, to further increase the amount of two-tone suppression in the CAR-FAC model, the minimum damping could be reduced, and/or the maximum damping could be increased or a higher threshold could be set so that channel sensitivities do not decrease until the probe levels are higher than 0 dB SPL.

3.5 Summary

This chapter compares the nonlinear responses to single-tone and two-tone stimuli of a TL model and a filter cascade model, CAR-FAC, of the mechanics of the human cochlea. The formal connections between these two models is also explained. The computed outputs of these two models have also been compared with experimental measurements on laboratory animals, which can be quite variable. Both models can account for the general features of these measurements, but not every aspect of each datasets.

In single-tone simulations, both models quantitatively agree with the wide range of experimental data in terms of starting SPL for compression to occur and compression rate. However, they are less accurate in simulating the reduction in peak frequency of the normalized BM transfer function with increasing input intensity and the effective bandwidth of their "cochlear amplifiers". In two-tone suppression simulations, the CARFAC model shows reasonable amount and rate of suppression, whereas the TL model exhibits lower amount of suppression and lower values of ROS than those measured physiologically in very sensitive cochlae [45], mainly because of limited amount of active gain, as discussed in section 3.4. The TL model also shows very limited two-tone suppression for very low- and very high-side suppressors. Finally, the suppression threshold curves predicted by both models are not compatible with biological data, although the CAR-FAC model produces closer results for low-side suppression. Thus, despite of its limitations, the CAR-FAC model is found to reproduce more closely the nonlinear phenomena observed in the auditory periphery, at least for single- and two-tone stimuli.

Chapter 4

Cochlea Modelling as a Front-end for Neural Network based Voice Activity Detection

In chapter 3, two types of cochlear model, a transmission-line model and a filter cascade model, CAR-FAC, were compared in terms of their realism in simulating some observed cochlear nonlinearities. The CAR-FAC model was found to yield responses closer to cochlear measurements at least for single- and two-tone stimuli, while being much more computationally efficient. In this chapter, we explore the potential benefits of using this filter cascade model as a front-end processor for one speech processing application, voice activity detection (VAD), to investigate whether the modelling of cochlear nonlinearity may lead to improvements in detection accuracy, in comparison to other simpler auditory filterbank features. Specifically, we integrate the CAR-FAC cochlear model with deep neural network based classifiers, since, as reviewed in chapter 2, they have superior performance compared with traditional shallow classifiers such as Gaussian Mixture Models and Support Vector Machines in a range of pattern classification tasks.

4.1 Introduction

Voice activity detection (VAD) refers to an algorithm that determines the presence or absence of speech, usually in a distorted signal (e.g., noisy, reverberant or both). It plays an important role in a number of modern speech signal processing applications. For instance, it improves the efficiency of speech communication system by detecting and transmitting speech-only segments or discontinuous transmission, since about 60% of the acoustic signal produced during a normal mobile communication contains just silence or noise [108]. It can also help many speech enhancement systems, which rely on a VAD to decide speech absent intervals, within which noise characteristic estimates are updated. In speech recognition systems, VAD can be beneficial in reducing the power consumption since they are normally computationally intensive and speech signals are only present for a fraction of time. In high noise scenarios, where a speech enhancement module is included for preprocessing before recognition, a VAD could also play an essential role, as discussed above. Furthermore, non-speech frame-dropping (based on the VAD decision) from the input is also a frequently used technique in speech recognition to reduce the number of insertion errors caused by the noise.

Motivated by the wide range of applications, VAD is subject to continuous research and numerous approaches have been proposed. Typically, a VAD module includes the following two main stages, as shown in Fig. 4.1: (i) feature extraction from the recorded noisy signals in order to achieve a representation that distinguishes speech from non-speech; (ii) a decision module that assigns a certain segment of observation to either a speech class or a non-speech class, based on the feature vector calculated from (i). Optionally, a third so-called hangover scheme can also be added, which is used to smooth the decisions obtained from step (ii) and to account for weak speech segments at the beginning and end of a speech active region. One of the major challenges to the VAD problem is that many of the proposed methods work satisfactorily with clean speech signals but fail dramatically when the level of the background noise or reverberation increases. Moreover, speech characteristics vary with time and can be affected by a series

of unpredictable factors, including speaker's gender, age, temper and vocal effort. Thus the development of a robust combination of feature vector and decision rule has been the primary focus of almost all the more recently proposed methods for speech detection in highly complex acoustic environments.

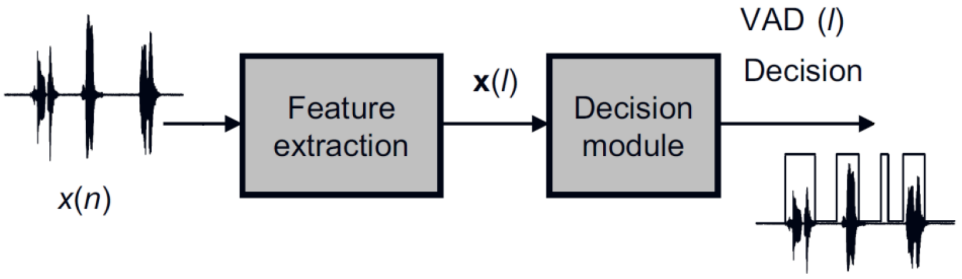


FIGURE 4.1: Block diagram of a typical VAD algorithm. Adapted from [108] with permission.

In practical operations, the recorded signal is usually divided into short frames of 10-30 ms with a frame shift of around 10 ms. For each frame, the goal of VAD module is to determine whether it contains speech or whether it is purely noise, i.e. the following two hypotheses,

$$\begin{aligned} H_1 &: \text{speech present}, \quad \mathbf{x}(i) = \mathbf{s}(i) + \mathbf{n}(i) \\ H_0 &: \text{speech absent}, \quad \mathbf{x}(i) = \mathbf{n}(i) \end{aligned} \tag{4.1}$$

where $\mathbf{x}(i)$ denotes a segment of observation at frame i, $\mathbf{s}$ and $\mathbf{n}$ denote speech and noise signals respectively and they are assumed to be additive, H_0 represents the hypothesis of noise-only frame and H_1 the speech present frame. The ideal decision for each of these two hypotheses at frame i is usually written as in Eq. 4.2, which formalizes VAD as a binary classification problem. Various algorithms then integrate different features with a number of classification paradigms to solve this problem. In the next section we give a brief overview of popular techniques in the field of VAD.

$$decision(i) = \begin{cases} 1, & \text{when } H_1 \text{ is accepted} \\ 0, & \text{when } H_0 \text{ is accepted} \end{cases} \tag{4.2}$$

4.2 An Overview of Existing VAD Methods

4.2.1 Feature Extraction

As the first step in most voice activity detection systems, numerous kinds of features have been developed. To achieve good performance in diverse acoustic environments, this representation should be able to capture the discriminative properties of the sound, and be resistant to distortions caused by various noise sources. Additionally, noise reduction, feature compensation and normalization techniques can be incorporated to reduce the mismatch between training and testing feature vectors, although in this work, we only consider feature extraction on its own. Early research investigated simple acoustic features such as short-term energy and zero-crossing rate [109], which work satisfactorily in high signal-to-noise ratio (SNR) environments, but fails significantly when the SNR decreases. During the following decades, feature extraction method has become increasingly sophisticated, enabling them to achieve reasonable detection accuracy in more challenging acoustic conditions. One exception is the rise of deep learning based methods, however, which prefer more raw feature representations like spectrograms or even the raw waveform to more advanced hand-engineered features. There are many ways to categorize these kinds of features, but since the aim of this work is to investigate the use of auditory modelling in machine learning based speech processing, we divide them into the following two classes depending on if any auditory principle is integrated in the feature extraction process: (a) Acoustic features and (b) Auditory-inspired features. We then briefly review some popular examples of both classes.

4.2.1.1 Acoustics features

In this work, acoustic features refer to those features that are based on mathematical transformations of the speech signals, while not explicitly relying on any human hearing principle.

Harmonicity Following the source-filter model of voice production, human speech can be regarded as the result of glottal airflows (also called the excitation or source signal) that are spectrally shaped by the vocal tract that consists of the oral, nasal, and pharyngeal resonant cavities [110]. Human speakers can produce a wide range of sounds by manipulating the vibration pattern of the vocal folds, or the configuration of the vocal tract above the larynx [111]. For voiced phonemes, periodic vibration of the vocal cords modulates the airflow from the lungs to produce a harmonically rich sound with a distinct pitch, typically between 50 and 400 Hz. This harmonic structure is one of the primary characteristics of voiced speech and thus a reliable indicator of speech presence. Various voicing or harmonicity based features have been developed, and a common approach is to use the time-domain autocorrelation function (ACF). For instance, the Maximum Autocorrelation Peak [112] finds the magnitude of the maximum peak of the ACF within the range of lags that correspond to the valid range of pitches of human voices. The Autocorrelation Peak Count [113], on the other hand, counts the number of peaks found in a certain range of lags.

Since we can treat voiced speech as a glottal pulse train filtered by the vocal tract, several techniques can be used to first deconvolve the influence of the vocal tract filter before characterizing the degree of harmonicity. This can be achieved by inverse Linear Predictive Coding (LPC) filtering and the Maximum LPC Residual Autocorrelation Peak measure simply finds the peak value of the autocorrelation function of the LPC residual signal. Apart from LPC, cepstral analysis, as given in Eq. 4.3, has also been used to separate the contribution of source and vocal tract for a speech segment. Specifically, the low order cepstral coefficients characterize the vocal tract filter, whereas the high order coefficients capture the excitation signal and normally show a strong peak for voiced speech within the valid pitch range. Hence the Maximum Cepstral Coefficient or Cepstral Peak [114] has also been used as a voicing measure for VAD.

$$ceps = DCT(\log(|FFT((x))|^2)) \tag{4.3}$$

One obvious limitation of harmonicity based features is that they cannot be expected to detect unvoiced speech phonemes, such as some fricatives, which lack the harmonic structure. Moreover, other repetitive noises such as motor noise or music components might be misinterpreted as speech by these measures [115].

Spectral Shape Another major characteristic of human speech is the vocal tract filter that allows human speakers to form different phonemes. The resonance frequencies of the varying vocal tract cavities, during speech production process, emphasize certain sections of the spectrum, resulting in a specific shape of the spectral envelope or formant structure for different types of phonemes. As introduced in the previous section, linear prediction and cepstral analysis are two popular ways of extracting the spectral envelope of a signal. Assuming that the effects of human vocal tract can be approximated by an infinite impulse response (IIR) filter, linear prediction technique aims to compute these filter coefficients (also often called LPC coefficients), which can then be used for speech detection. An alternative representation of the LPC coefficients is the line spectral frequencies (LSFs), which can be interpolated while retaining stability of the corresponding IIR filter. For this reason, LSFs are utilized in the standardized VAD procedure, ITU-T G.729 Annex B [116], which is one of the most popular baseline methods used for comparison in the literature. In terms of cepstral analysis, low order cepstral coefficients can also be used. However, instead of using the discrete cosine transform (DCT) coefficients directly, the most popular way to compute the cepstral coefficients is to transform the linear frequency scale to a perceptually motivated Mel-scale [14] and compute the Mel spectrum before performing logarithm compression and DCT analysis, leading to the so-call Mel frequency cepstral coefficients, or MFCC. More details of MFCC feature extraction is given in the next section.

4.2.1.2 Auditory-inspired Features

There is a long history in incorporating various auditory principles for effective representation of speech signals for a wide range of applications, including VAD. In this section, we briefly review some popular auditory inspired features that are originally developed mainly for speech recognition and separation, but have also been applied to speech detection problem and shown promising results. These features are compared with the filter cascade spectrogram in the next section to explore the potential advantages of filter cascade based cochlear modelling for VAD tasks.

Mel Spectrogram and Cepstral Coefficients Probably the most popular auditory-based feature is the Log-Mel spectrogram (LogMel) and Mel Frequency Cepstral Coefficients (MFCC), which were first introduced in the 1980s [117]. It is similar to standard cepstral analysis, except that it transforms the linear frequency axis of power spectrum estimate into the perceptually motivated mel-scale [14]. Spectral components at frequency bins that are within the passband of a bank of triangular-shaped filters (or mel-filterbank), the centre frequencies of which are equally spaced along the mel-scale are accumulated to simulate the reduced frequency resolution at higher frequencies in human auditory system. Taking the natural logarithm of the mel-transformed power spectrum results in the so-called Log-Mel spectrogram, and further taking DCT of this representation yields the MFCC.

Power-normalized Spectrogram and Cepstral Coefficients Power normalized spectrogram (PNspec) and its cepstral coefficients (PNCC) [118] represent one of the major extensions to the original Log-Mel spectrum and MFCC. Fig. 4.2 compares the structure of PNCC processing to that of MFCC feature extraction algorithm as presented above. As can be seen, the major innovations of PNspec and PNCC processing include a series of modules designed to compensate adverse

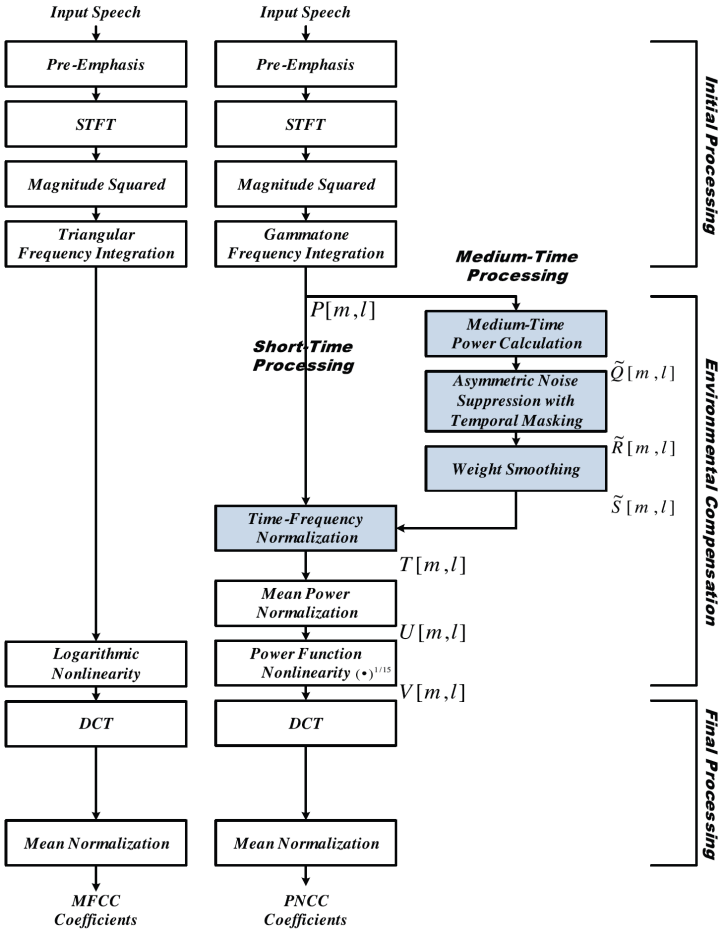


FIGURE 4.2: Comparison of the computation structure of the MFCC, and PNCC algorithms. Picture adapted from [118] with permission.

acoustic environments based on medium-duration temporal analysis and the redesigned nonlinear rate-intensity function. The initial processing stages of PNCC are quite similar to the corresponding stages of LogMel and MFCC, except that the frequency integration is performed using gammatone filters. This is followed by the asymmetric noise suppression and temporal masking modules that can achieve noise reduction as well as some degree of robustness to reverberation. The final stages of processing use a carefully-chosen power-law nonlinearity with an exponent of 1/15, to approximate the nonlinear relation between signal intensity and auditory-nerve onset firing rate, which provides superior robustness by suppressing small signals and their variability.

Multi-resolution Cochleagram (MRCG) Multi-resolution cochleagram is another auditory-inspired feature that is originally proposed for classification-based speech separation [119] and was later applied to the VAD problem. The key idea of MRCG is to incorporate both local and global information through multi-resolution extraction. The local information is produced by extracting cochleagram features with a small frame length or a small smoothing window (i.e., high resolutions), while the global information is extracted with a large frame length or a large smoothing window (i.e., low resolutions). As illustrated in Fig. 4.3(a), MRCG is a concatenation of 4 cochleagram features with different smoothing window sizes and frame lengths. The first and fourth cochleagram features are generated from two N-channel gammatone filterbanks [2] with frame lengths set to 20 ms and 200 ms respectively. The second and third cochleagram features are calculated by smoothing each time-frequency unit of the first cochleagram feature with two square windows that are centred at one given time-frequency unit and have the sizes of 11×11 and 23×23. At the boundaries, where there are not enough channels and/or frames to cover the full range of the square windows, the overflowed parts of the windows are truncated correspondingly.

It is worth noting that the number of filters, N, in the gammatone filterbank can be varied in different implementations, and it was set to 64 and 8 in two VAD studies [120] and [6] respectively by the same authors. In this work, we adopt a realization that utilizes 16 gammatone filters at each resolution, which is different from the previous studies, the reason of which is given in section 4.3. The calculation of a single-resolution eight-dimensional cochleagram feature in Fig. 4.3(a) is detailed in Fig. 4.3(b). The input noisy speech is firstly filtered by an 8-channel Gammatone filterbank, then the energy of each time-frequency unit is calculated by $\sum_{k=1}^{K} s_{c,k}^2$, where $s_{c,k}$ represents the kth sample of a given filtered frame in the cth channel and K is frame length in samples. Finally the energy in each frame is rescaled by the logarithm function with base of 10.

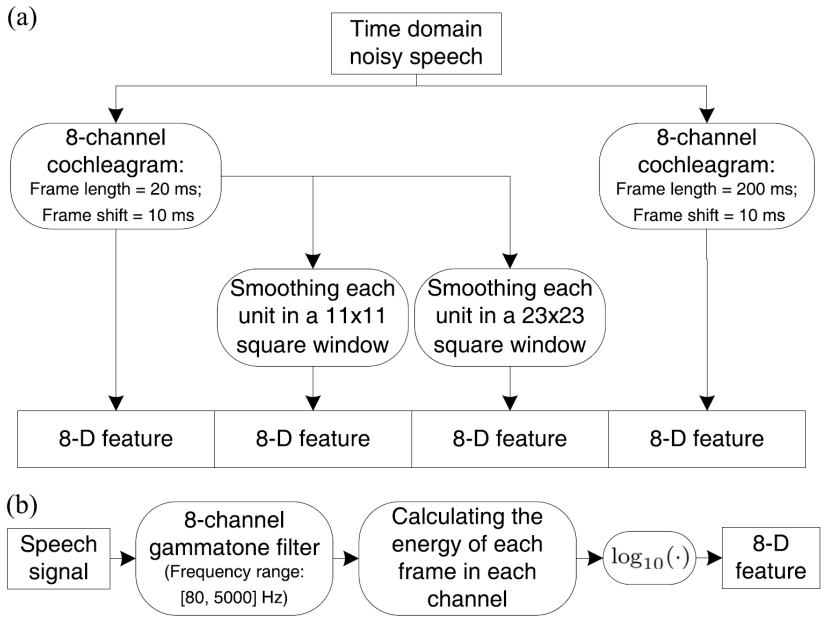


FIGURE 4.3: The MRCG feature. (a) Diagram of the process of extracting a 32-dimensional MRCG feature. (b) Calculation of one single-resolution 8-dimensional cochleagram features in detail. Adapted from Fig. 3 of [6] with permission.

4.2.2 Decision Rules

After feature extraction, the next module for most VAD algorithms is a set of decision rules or classification schemes that partitions the feature space into two non-overlapping regions, each of which belongs to one of the two speech activity hypotheses. Broadly speaking, these classification methods can be grouped into three categories, following Chapter 2.2 of [121]:

- **1.** Thresholding.
- **2.** Statistical modelling based approaches.
- **3.** Machine learning based approaches.

Thresholding is the simplest form of the decision rule, which just uses a line or a set of lines to divide the feature space. For instance, if the extracted feature at the i^{th} frame is a scalar x_i, a single scalar threshold, η, can be used. To tackle frequently changing noise condition, adaptive thresholding techniques have also been developed [122, 123]. However, thresholding based methods essentially

assume that features extracted from speech and nonspeech are linearly separable, which normally does not hold in complex acoustic environments. Thus we do not consider it any further in this work, but briefly review the statistically modelling and machine learning based approaches.

4.2.2.1 Statistical Modelling Approaches

For the two-hypothesis testing problem in VAD, the optimal decision rule that minimizes the error probability is the Bayes classifier, i.e., for each frame, i, the class selected is the one with the largest posterior probability $\max_j P(H_j|\mathbf{x}), j = 0, 1$, or more formally,

$$P(H_1|\mathbf{x}) \underset{\mathrm{H_0}}{\overset{\mathrm{H_1}}{\gtrless}} P(H_0|\mathbf{x}) \tag{4.4}$$

In the statistical modelling based framework, Eq. 4.4 is transformed to a generalized likelihood ratio test using the Bayes rule,

$$\frac{P(\mathbf{x}|H_1)}{P(\mathbf{x}|H_0)} \underset{\mathrm{H_0}}{\overset{\mathrm{H_1}}{\gtrless}} \frac{P(H_0)}{P(H_1)} \tag{4.5}$$

Or equivalently,

$$\frac{P(\mathbf{x}|H_1)}{P(\mathbf{x}|H_0)} \underset{\mathrm{H_0}}{\overset{\mathrm{H_1}}{\gtrless}} \eta \tag{4.6}$$

where η is a tunable decision threshold. If speech, $\mathbf{s}$, and background noise, $\mathbf{n}$, are assumed to independent Gaussian random processes, their discrete time Fourier transform (DFT) coefficients at frame i, $\mathbf{S}(i) = [S(i,1), S(i,2), ..., S(k), ..., S(K-1), S(K)]$ and $\mathbf{N}(i) = [N(i,1), N(i,2), ..., N(k), ..., N(K-1), N(K)]$, are also Gaussian and asymptotically independent, where k denotes the spectral bin index and K is the total number of spectral bins. The probability density functions of the observed DFT coefficients, $\mathbf{X}(i) = [X(i,1), X(i,2), ..., X(i,k), ..., X(i,K-1), X(i,K)]$, conditioned on the two speech activity hypotheses can thus be written

as,

$$p(\mathbf{X}(i)|H_0) = \prod_{k=1}^{k=K} \frac{1}{\pi\lambda_{n,k}} \exp\left\{-\frac{|X(i,k)|^2}{\lambda_{n,k}}\right\} \tag{4.7}$$

$$p(\mathbf{X}(i)|H_1) = \prod_{k=1}^{k=K} \frac{1}{\pi(\lambda_{s,k} + \lambda_{n,k})} \exp\left\{-\frac{|X(i,k)|^2}{(\lambda_{s,k} + \lambda_{n,k})}\right\} \tag{4.8}$$

where $\lambda_{s,k}$ and $\lambda_{n,k}$ denote variances of speech and noise at the k^{th} frequency bin respectively. The likelihood ratio at the k^{th} spectral bin is then defined as,

$$\Lambda_k = \frac{p(X(i,k)|H_1)}{p(X(i,k)|H_0)} = \frac{1}{1+\xi_k} \exp\left\{\frac{\gamma_k \xi_k}{1+\xi_k}\right\} \tag{4.9}$$

where $\xi_k = \frac{\lambda_{s,k}}{\lambda_{n,k}}$ and $\gamma_k = \frac{|X(i,k)|^2}{\lambda_{n,k}}$ is referred to as *a priori* and *a posteriori* signal to noise ratio respectively and are firstly introduced in statistical-model based speech enhancement [124]. Finally, the likelihood ratio for the i^{th} frame is commonly computed from the geometric mean of the likelihood ratios for all the frequency bins, and is then put in the logarithmic domain,

$$\log\Lambda = \frac{1}{K}\sum_{k=1}^{k=K} \log\Lambda_k \underset{\mathrm{H_0}}{\overset{\mathrm{H_1}}{\gtrless}} \eta \tag{4.10}$$

Here η is still the threshold parameter for choosing one of the speech activity hypotheses. Solving the VAD problem now becomes solving the estimation of two SNR measures, ξ_k and γ_k. Estimation of noise variance, $\hat{\lambda}_{n,k}$ and hence γ_k, can be obtained by many noise tracking algorithms, such as the minimum statistics based methods [125, 126] and MMSE-based approaches [127, 128]. For the *a priori* SNR estimation, the decision-directed approach [124], that is originally proposed for speech enhancement, is often adopted,

$$\xi(i,k) = \alpha\frac{|\hat{S}(i-1,k)|^2}{\hat{\lambda}_n(i-1,k)} + (1-\alpha)P[\gamma(i,k)-1]) \tag{4.11}$$

where $|\hat{S}(i-1,k)|^2$ is the speech spectral amplitude estimate of the previous frame obtained using the MMSE estimator [124]; α is a smoothing parameter that is usually assigned a value between 0.95 and 0.99; P is an operator that is defined

by,

$$P[x] = \begin{cases} x, & \text{if } x \geq 0 \\ 0, & \text{otherwise} \end{cases} \tag{4.12}$$

Despite its simplicity and popularity, the Gaussian distribution is not an accurate model for the distributions of speech, noise and their combination in many scenarios, mainly due to speaker and acoustic environment variations. To improve the quality of statistical modelling, a number of alternative distribution functions have also been adopted, including the Laplacian [129], Gamma [129] and Generalized Gaussian distribution [130]. Instead of using just one statistical model, authors in [131] utilize a set of preferred distributions and design a switching mechanism that automatically chooses the most suitable one in different situations based on an online Komogorov-Smirmov test [132] between the observed speech samples and all of the available distribution functions. Inspired by the same idea, Petsatodis *etal.* [133] proposed a convex combination of multiple distributions to model the speech signal in any given frame. The relative contribution of each of these models is determined by a set of weights and the final model showed an improved modelling capability.

4.2.2.2 Machine Learning Approaches

As reviewed in Chapter 2, machine learning techniques can automatically learn to solve certain tasks from provided training data without being explicitly programmed. In terms of VAD, this is a classical binary pattern classification problem, for which a wide range of powerful models have been proposed. Early research in machine learning based VAD mostly utilize shallow classifiers, such as Gaussian mixture model, support vector machine, and MLP. More recently, inspired by its huge success in various speech processing tasks such as speech recognition and natural language processing, deep learning based methods, especially neural networks, have become much more popular and have shown good performance in extremely difficult scenarios. Thus we review these neural network models for supervised VAD in more details in the following parts.

In contrast to statistical modelling approaches that rely on explicit assumptions about speech and noise distributions, neural network based VAD systems directly estimate the posterior probabilities of speech and non-speech classes using several layers of nonlinear transformations of the extracted features. They can easily fuse the advantages of multiple feature types that capture distinct properties of speech and non-speech and have the ability to discover the underlying regularity within these features for the speech detection task using their hidden layers. In the output layer, either a single sigmoid unit or two softmax units can be used. In this work, the softmax output layer is adopted to jointly estimate speech presence and absence probabilities. And the final VAD decision is obtained by comparing their difference with a threshold parameter η, as shown in following function,

$$P(H_1|\mathbf{x}) - P(H_0|\mathbf{x}) \underset{\mathrm{H_0}}{\overset{\mathrm{H_1}}{\gtrless}} \eta \tag{4.13}$$

Driven by the initial interests in unsupervised layer-wise pre-training, Zhang and Wu [134] proposed to apply the deep belief network with sigmoid hidden units to VAD using a total number of ten classical feature types, such as MFCC, perceptual linear prediction, amplitude modulation spectrum and pitch based features. In most of the noisy scenarios tested, this system was shown to outperform 11 reference VAD methods, including traditional algorithms such as G729B VAD [116] and ETSI advanced front-end methods [135], statistical model based approaches [136, 137, 138], Support Vector Machine based supervised methods [139] and an unsupervised machine learning based VAD [140]. Zhang and Wang [6] further developed a deep architecture that utilises a multi-resolution stacking of an ensemble of boosted DNNs. When trained with MRCG features, the proposed method produced considerable performance improvements compared to standard DNN based and classical VAD algorithms.

Recently, one major research effort in machine learning based VAD is the DARPA Robust Automatic Transcription of Speech program (RATS), the goal of which is to develop techniques for speech activity detection, language identification, speaker identification and keyword search in multiple languages on degraded audio signals

transmitted over communication channels (labelled as channel A to H) that are extremely noisy or highly distorted [141]. The VAD module is of great importance to this program because it serves as the first step to select speech only regions that can be sent downstream to the other components, such as speaker identification and keyword search, for further processing. A wide variety of features were employed for this VAD module, including most of those introduced in section 4.2.1. In terms of acoustic modelling, initial development in phase one utilized shallow models including Gaussian Mixture Model and MLP, and the segmentation of audio signals is obtained either by a log likelihood ratio test or by treating it as a simple speech recognition problem with a three "word" vocabulary (speech, non-speech and non-transmission), in which an HMM based decoding is performed. In the second and third phases of this program, deep neural network based acoustic models were adopted. For instance, in phase three, the IBM VAD system [142] used a DNN model to combine multiple advanced feature extraction schemes by concatenating their feature vectors, and a CNN model that operated on the more raw topographical features such as Log-Mel and Gammatone filterbank energy features to allow automatic feature engineering. A hybrid model was then created by fusing the DNN and CNN model outputs using several shared fully connected hidden layers and a single output layer, as shown in Fig. 4.4. After jointly training this CNN-DNN architecture, considerable performance improvements were achieved compared to the systems in all previous phases. In spite of these impressive improvements, most of the models proposed in this program are channel-dependent, i.e., there is a separate model for each of the eight channel degraded signal and the one selected is determined by a channel selection module. Hence it is difficult to know exactly how well these systems can generalize in unseen channel conditions and noise scenarios.

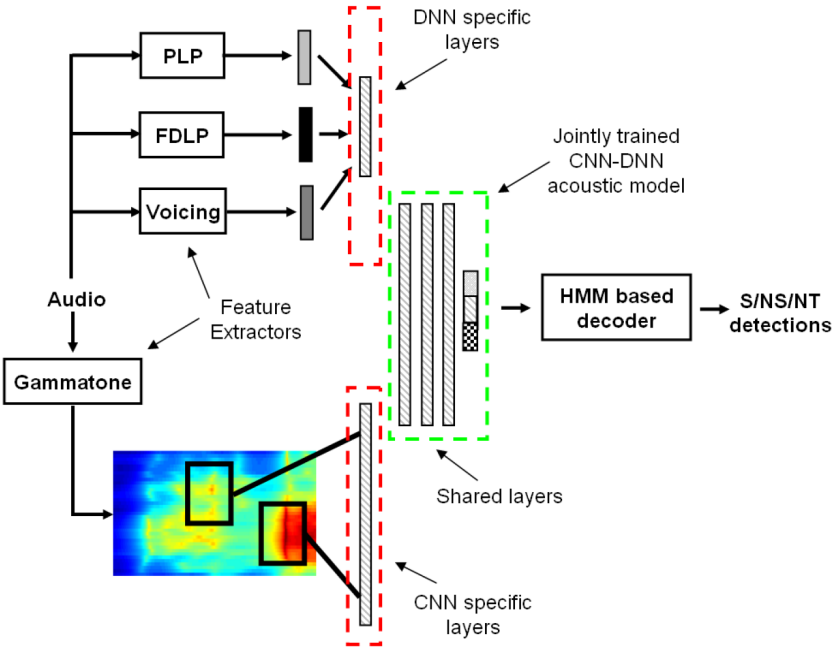


FIGURE 4.4: Block diagram of the IBM VAD system developed for the DARPA Robust Automatic Transcription of Speech (RATS) program. This is a hybrid model including a DNN and a CNN sub-module and is jointly trained on multiple types of features. PLP: Perceptual Linear Prediction; FDLP: Frequency Domain Linear Prediction. Figure is adapted from Fig. 2 of [142] with permission.

4.3 Filter Cascade Spectrogram for Neural Network based VAD

In this section, we investigate the use of the filter cascade model of human cochlea, as introduced in previous chapter, for neural network based VAD to explore the benefits of modelling cochlear nonlinearity in improving detection accuracy in challenging acoustic environments. To evaluate its advantages and disadvantages, the performance of the filter cascade spectrogram feature is compared with five other popular time-frequency representations, including short-time Fourier transform (STFT) based Log-Power spectrogram, Log-Mel-filterbank, Gammatone spectrogram or Cochleagram, Power-normalized spectrogram and MRCG. Cepstral coefficients are not considered in this work because it has been demonstrated in many deep learning based speech recognition studies and also in our preliminary simulations that they are inferior to their corresponding spectrogram features when

TABLE 4.1: List of different feature types used in this study and their dimensions at each time frame.

Feature type	Description	Dimension
Spec	STFT based Log-Power spectrogram	$257 \times 3 = 771$
LogMel	Log-Mel filterbank spectrogram	$64 \times 3 = 192$
GTspec	Gammatone filterbank spectrogram or Cochleagram	$64 \times 3 = 192$
PNspec	Power Normalized spectrogram	$40 \times 3 = 120$
MRCG	Multi-resolution Cochleagram	$64 \times 3 = 192$
FCspec	CARFAC filter cascade spectrogram	$66 \times 3 = 198$
MRLM	Multi-resolution Log-Mel filterbank spectrogram	$64 \times 3 = 192$
MRFC	Multi-resolution CARFAC filter cascade spectrogram	$66 \times 3 \times 4 = 792$
MWLLM	Multi-window-length Log-Mel filterbank spectrogram	$64 \times 3 = 192$
MWLCG	Multi-window-length Cochleagram	$64 \times 3 = 192$
MWLFC	Multi-window-length CARFAC filter cascade spectrogram	$66 \times 3 \times 4 = 792$

neural network based acoustic modelling is adopted. As will be presented in section 4.7, we also apply the same multi-resolution strategy as employed in MRCG and multi-window-length extension to the above spectrogram features, which are capable of producing further improvements to VAD system. The full list of different feature types used in this work and their dimensions at each time frame are shown in Table 4.1.

All input clean speech signals are sampled at 16 kHz and rescaled to 60 dB SPL before mixing with various types of noises at a number of SNR levels. The reason for such rescaling is to allow the filter cascade model to exhibit significant nonlinearity to input signals. Different spectrogram features are computed using an analysis window of 25 ms (or 400 samples) with a window shift of 10 ms (or 160 samples). Filter centre frequencies of various auditory filterbanks are chosen to be in the range of 50 Hz to 8000 Hz, equally spaced in-between, using

the corresponding auditory frequency scale. For the filter cascade model, this results in a total number of 66 serial sections following the default settings given in [98]. It is worth noting that it is not very straightforward to change the channel number in the filter cascade model. This is because its internal mechanisms for simulating cochlear nonlinearity such as the amount of active gain, depend on the predefined channel density, so we just follow the default settings for convenience. For Log-Mel and Gammatone spectrograms, a 64-channel Mel filterbank and a 64-channel Gammatone filterbank is created respectively, with centre frequencies equally spaced along the Mel scale and ERB-rate scale. This channel number is selected to make them similar in size to the filter cascade model, so that the real inherent difference between various time-frequency representations for the task of VAD can be revealed. For Power-normalized spectrogram, the original authors in [118] published an implementation with 40 channels, with its parameters and internal processing empirically optimized for a speech recognition task . Although this channel number can be increased to 64, our initial simulations show that this can make VAD performance worse, probably because of parameter and filterbank size mismatch, as also noted by the authors [118]. So we only use the original 40-channel implementation in all following experiments. For MRCG feature, we use 16 Gammatone filters for each resolution, so that the dimension of static feature vector is (16) 64. Note that this is in contrast with the application of MRCG in VAD task in [120] and [6], where 64 and 8 filters were used for each resolution respectively. The main reason for choosing such channel density is to reduce filterbank size difference between various auditory models. To compute Log-Power spectrogram, a 512-point FFT analysis is performed for each frame, from which only the first half is selected to form a static feature vector with dimension of 257. Finally, static feature vector of all feature types are expanded with their delta and double-delta dynamic elements within a five-frame window, as the following,

$$\Delta x_{t,c} = \frac{x_{t+1,c} - x_{t-1,c} + 2(x_{t+2,c} - x_{t-2,c})}{10} \tag{4.14}$$

where $x_{t,c}$ is the feature component at t^{th} time frame and c^{th} channel. The double-Delta feature is calculated by applying Eq. 4.14 to the Delta feature $\Delta x_{t,c}$.

To visualize the differences between these features, Fig. 4.6 and Fig. 4.7 show an example of each of them for a clean utterance from the TIMIT dataset [143] (left column) and the same utterance corrupted by a factory noise from the NOISEX dataset [144] at 0 dB SNR (right column), the waveforms of which are shown in Fig. 4.5. It can be seen that all of the auditory inspired time-frequency representations devote more effort to represent low frequency components than high frequency ones, compared to the traditional STFT, which is a direct consequence of the logarithmic-like auditory frequency scale followed in these filterbanks. Compared to Log-Mel spectrogram, Gammatone filterbank produces less sharply resolved spectral peaks, probably because of its wide bandwidth compared to the triangular Mel filterbank. The power normalized spectrogram is quite similar to the Gammatone representation, although spectral resolution and channel activity is further reduced, while onset of power envelop is enhanced. This is due to the asymmetric noise suppression, temporal masking and spectral weight smoothing modules of its implementation, which significantly reduce the effects of noise as shown in the right column of Fig. 4.6. In the filter cascade spectrogram, both spectral energy distribution and phoneme boundaries are smeared, and these phenomena result from the active gain control and inter-channel interactions mechanisms within the filter cascade model. For the MRCG feature, because the low number of filters used for each resolution, spectral representation of speech is mostly very coarse, but each level of resolution captures distinct characteristics of speech, which could be beneficial for speech detection.

Under additive factory noise, the Log-Power spectrogram displays corruptions that are more concentrated in the low frequency region, while Log-Mel feature equalizes such disturbance within the time-frequency (T-F) plane, showing more clearly the harmonic structure of voiced speech. This behaviour is also observed in Gammatone spectrogram, although the influence in the high frequency region is more pronounced in this case. The Power normalized spectrogram seems to have the highest level of noise robustness, because visually speaking, the noisy features

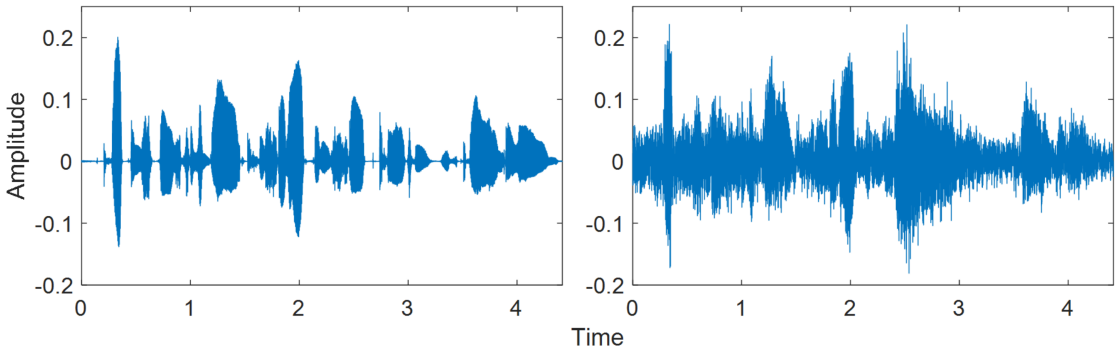


FIGURE 4.5: An example clean utterance from the TIMIT dataset and the same utterance corrupted by a factory noise from the NOISEX dataset at 0 dB SNR.

bear the highest level of resemblance to its clean counterparts compared to other all spectrogram features. The filter cascade model, on the other hand, seems to be the feature type that is most affected by additive noise, because almost the entire T-F plane is filled with high level of disturbance activity. Finally, MRCG behaves similarly to Gammatone spectrogram in the first resolution, but different resolutions are affected by noise to different extents.

It is worth noting that all of the spectrograms shown in Fig. 4.6 and Fig. 4.7 are feature vectors without any post processing. In this work and most other studies, feature vectors are firstly normalized to zero mean and unit variance before using them to train and test neural network acoustic models. Thus, it is useful to view how various spectrograms change after this normalization procedure. Fig. 4.8 and Fig. 4.9 show the same set of spectrogram features for the same clean and noisy utterances but after mean and variance normalization (MVN) using sentence wide statistics. It is worth noting that after normalization, all T-F representations are less affected by additive noise and obtain a certain level of noise resistance. This is best reflected in the extra resolutions included in the MRCG feature.

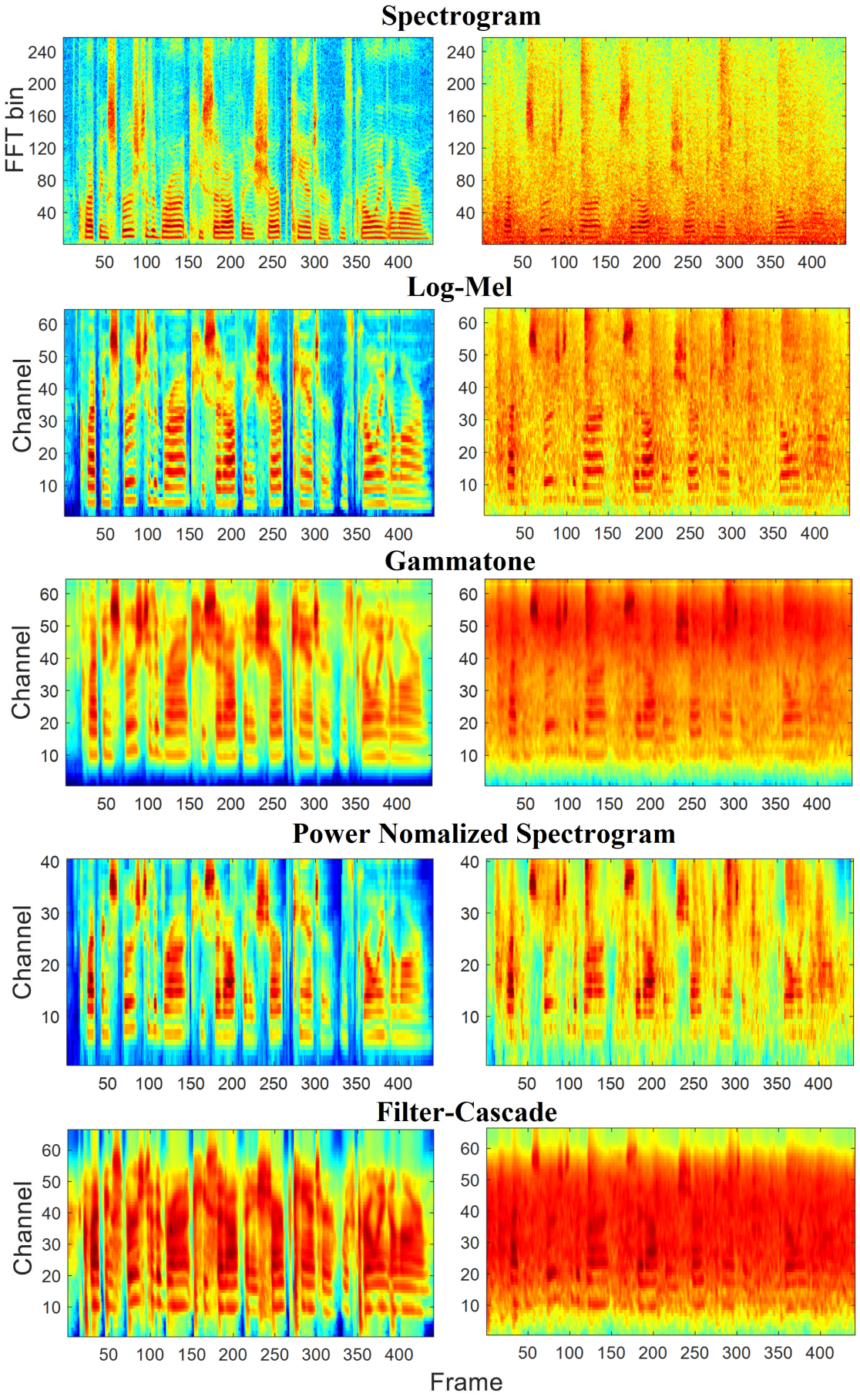


FIGURE 4.6: Visualization of various un-normalized time-frequency representations for an example clean utterance from the TIMIT dataset (right column) as shown in Fig. 4.5 and the same utterance corrupted by a factory noise from the NOISEX dataset at 0 dB SNR (left column).

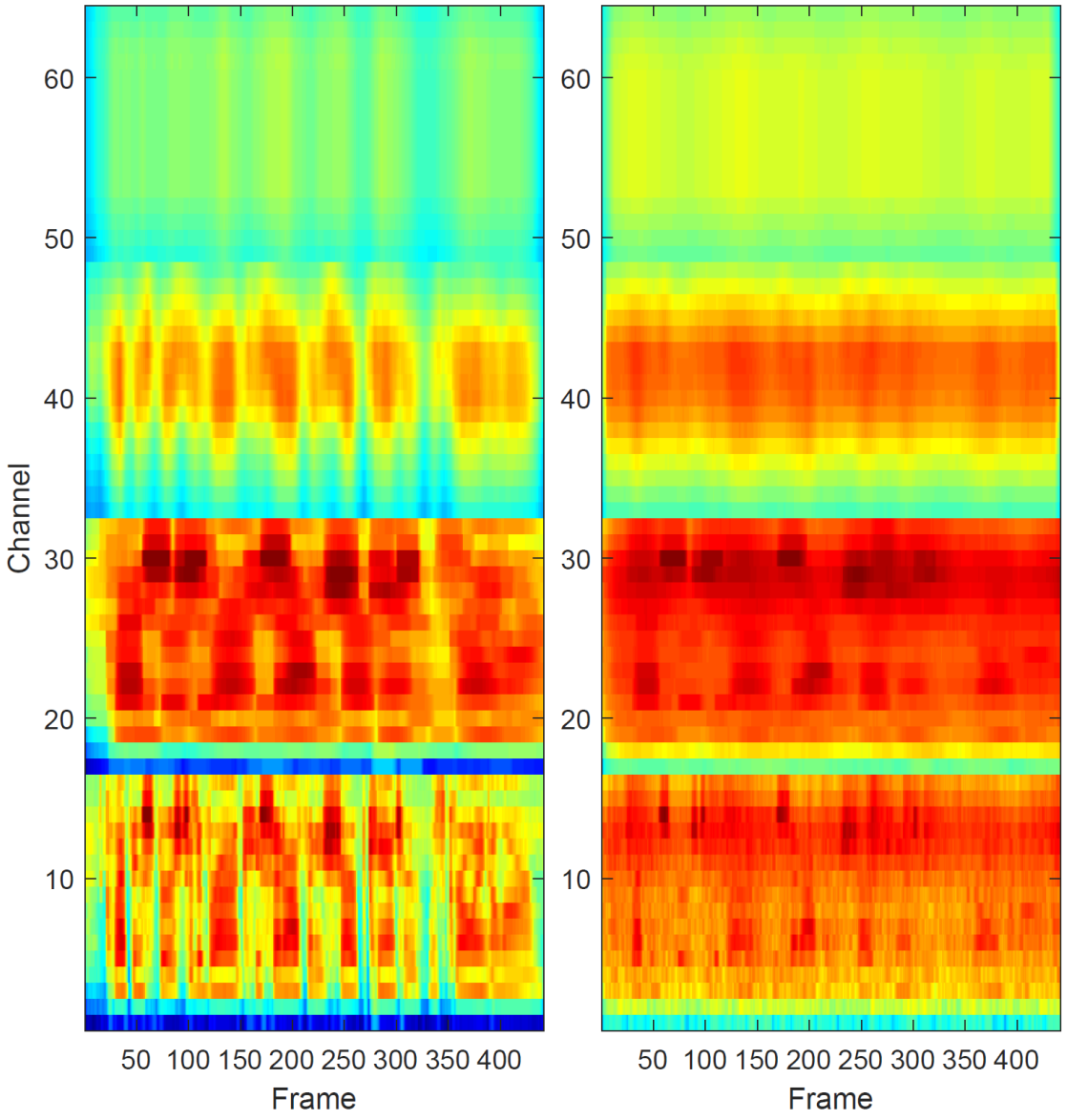


FIGURE 4.7: Visualization of un-normalized MRCG representation of an example clean utterance from the TIMIT dataset (right column) as shown in Fig. 4.5 and the same utterance corrupted by a factory noise from the NOISEX dataset at 0 dB SNR (left column).

4.4 Simulation Setup

4.4.1 Datasets

The clean speech utterances used in the following experiments are selected from the TIMIT dataset [20], which contains in total 6,300 sentences spoken by 630 speakers (192 females and 438 males) from 8 different dialect regions in north America, with

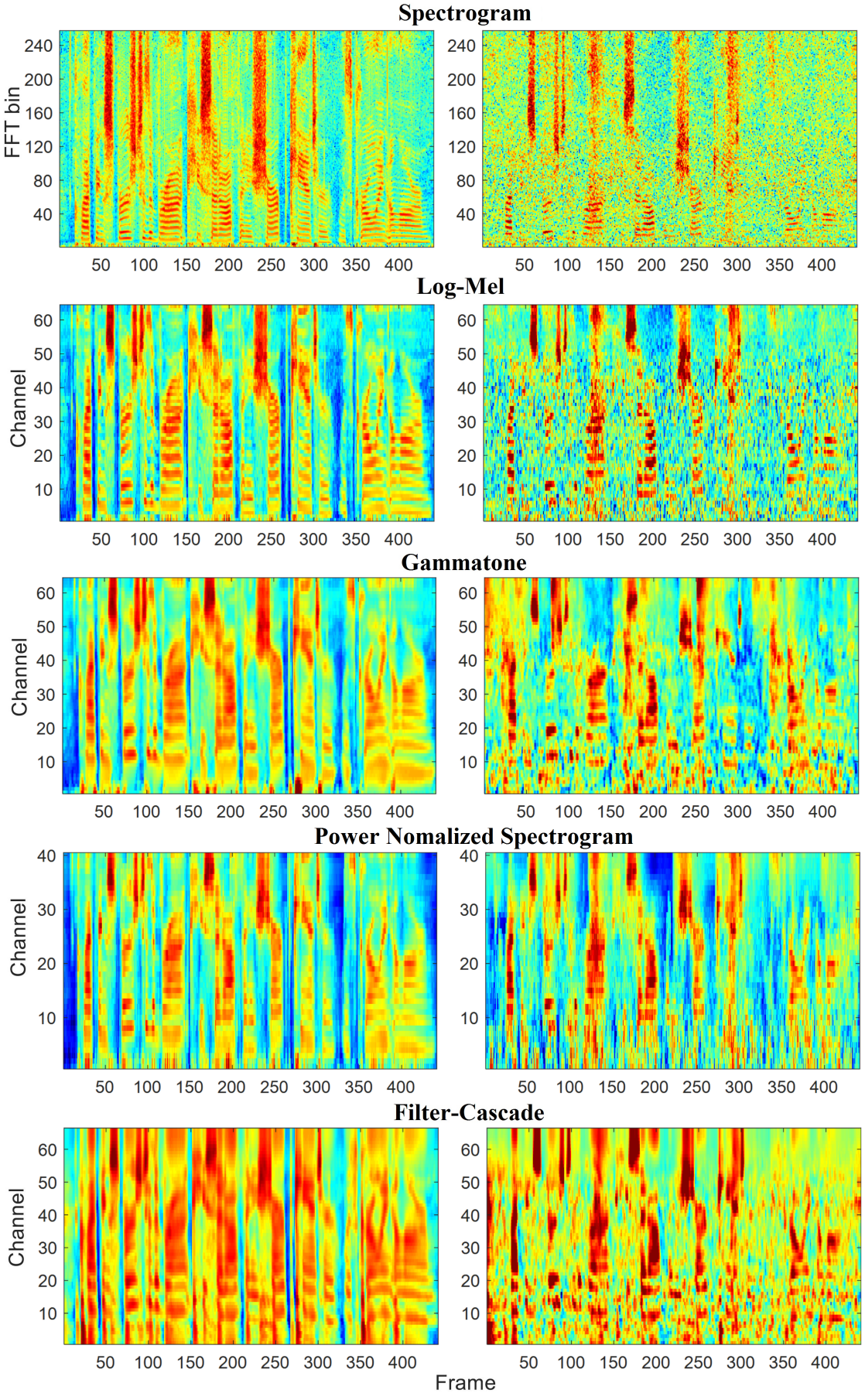


FIGURE 4.8: Visualization of various mean-variance normalized time-frequency representations for an example clean utterance from the TIMIT dataset (right column) and the same utterance corrupted by a factory noise from the NOISEX dataset at 0 dB SNR (left column).

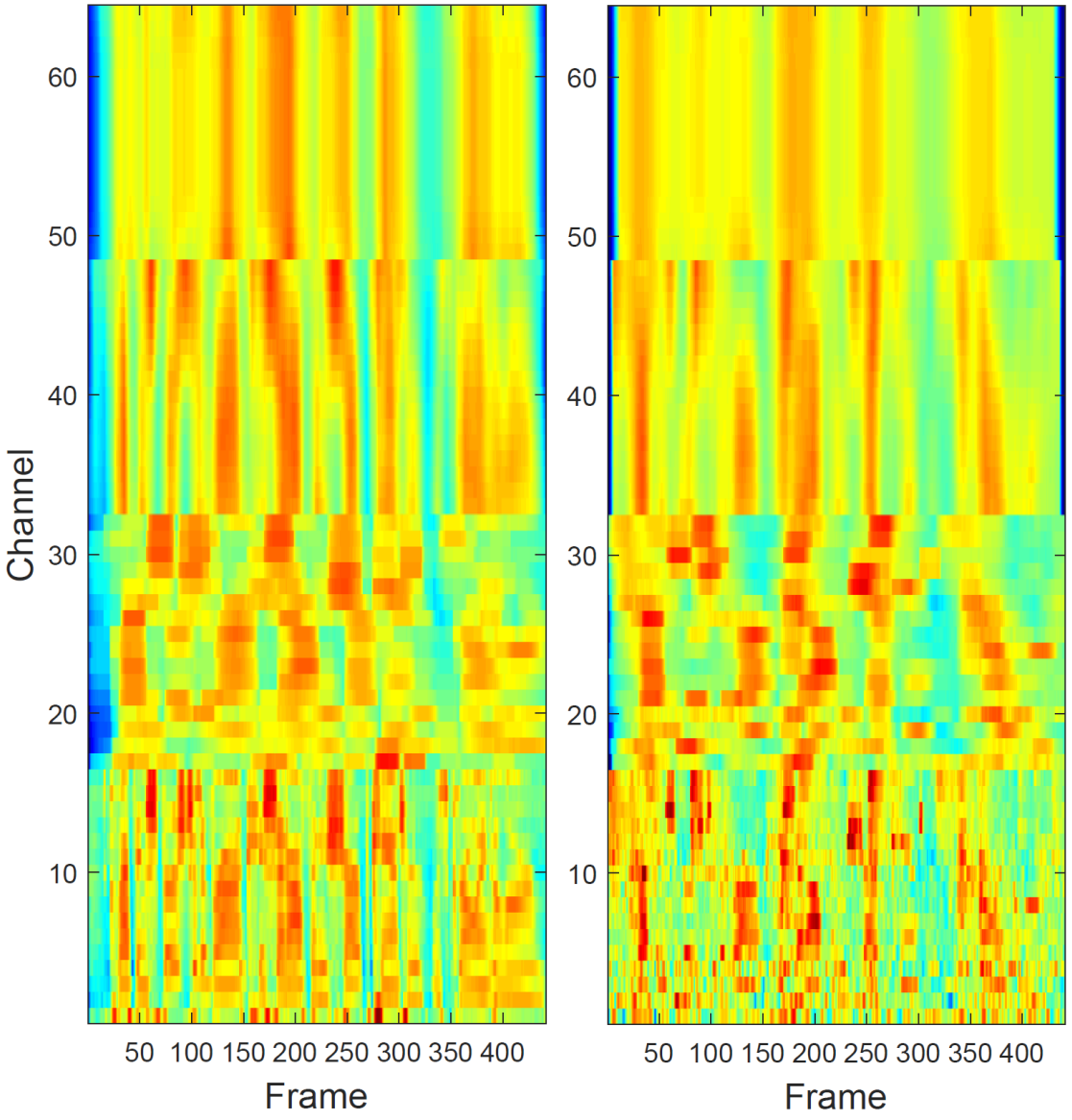


FIGURE 4.9: Visualization of mean-variance normalized MRCG representation of an example clean utterance from the TIMIT dataset (right column) and the same utterance corrupted by a factory noise from the NOISEX dataset at 0 dB SNR (left column).

each speaker speaking 10 sentences. There are three types of utterances in this dataset: SA (2 sentences), SX (450 sentences) and SI (1890 sentences), and they were specifically designed to balance the conflicting desires of compact phonetic coverage, contextual diversity, and speaker variability. This database is already divided into training (4620 sentences) and test (1680 sentences) subsets, with specially chosen 24 speakers or 240 sentences from the test subset forming the so-called core-test set. In this work, we call the utterances in the test set excluding

the core-test set the validation set. The average duration of an audio file is about 3 s, but it contains significantly longer duration of speech than nonspeech. To balance the number of speech and nonspeech frames, a 0.5 s of silence was added to both the start and end of each audio file.

To cover a wide range of practical noise scenarios, three noise datasets are considered. The first one is the well-known NOISEX-92 database [144], from where we selected seven types of noise, including babble, tank (m109), fighter plane (F16), two factory noises (labelled as factory1 and factory2) and two ship noises (labelled as ship oproom and ship engine). The second noise dataset is the QUT-NOISE corpus [145] which is specifically developed for evaluation of VAD algorithms. It consists of five categories of noise scenarios: café, car, home, street and reverberant environment. In each category, two separate noise sessions of at least 30 minutes each were recorded at each of two different locations, for example, in kitchen and living room in the home noise scenario, resulting in a collection of 20 noise sessions. But for the simulations performed in this work, only the four recordings from Cafe-indoor and Car-Window down scenarios were used. Finally, three samples of train noise recorded inside a train from the Loizou noise corpus [146] were utilized to further enrich the noise types. Sampling frequencies of the recordings in these three noise datasets are different, so they were all resampled to 16 kHz before mixing with the clean speech materials in the TIMIT database which have a sampling frequency of 16 kHz. A summary of the noise databases and noise types used is given in Fig. 4.10,

For training and model testing, we perform both noise-dependent and noise-independent training and testing. The term noise-dependent means that noise type and SNR are the same in training and test sets (different noise segments are used) , i.e., matched condition. This is to establish the upper limit of various combinations of features and neural networks, because training and test conditions are fully matched. Noise-independent training means that a neural network is trained with a large number of noise types with a wide variation of SNR levels, and it is also tested in a range of unseen and challenging noise scenarios. This is mainly used to evaluate the generalisation ability of different features and networks, which

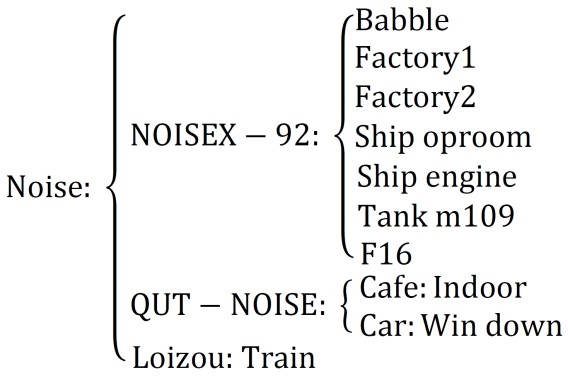


FIGURE 4.10: A summary of the noise datasets used in experimental simulations. "Win down" refers to driving the car with windows down and "Win up" means the opposite.

is one of the main goals for using machine learning in real-world applications. Ideal targets used in all training sessions are obtain by applying the G.729B VAD algorithm [116] to clean TIMIT utterances. This is considered to be reasonable, as clean TIMIT utterances have nearly infinite signal-to-noise ratio, which ensures high accuracy of such automatic labelling. However, the most accurate training targets should be obtained from the phonetic transcription that accompanies each utterance in the TIMIT database.

In noise-dependent training condition, all of the clean sentences in TIMIT dataset, excluding the core test set (6300-240 = 6060 sentences) are used, from which 90% of the feature vectors computed from these utterances are employed for training neural networks, while the remaining 10% are reserved as a validation set to regularise the training process. Finally, all of the 240 sentences from the TIMIT core-test set are employed to evaluate the performance of each trained neural network model. To create noisy sentences, we select two types of noise from both the NOISEX-92 database (Factory1 and Ship oproom) and the QUT-NOISE corpus (Cafe and Car), so the total number of noise types is four. For each noise type, every sentence from training and validation sets is mixed with a randomly selected segment of either the first half of noise recordings from the NOISEX-92 database or the first recording for noises from the QUT-NOISE corpus, at 4 SNRs of -5, 0, 5 and 10 dB. Note that SNR is computed using the original TIMIT utterances, that is prior to the addition of extra 0.5 seconds of silence to the start and the end

of each audio file. The test set is created similarly, except that noise segment is random cut from either the second half of or the second noise recording for noises from the NOISEX-92 database and the QUT-NOISE corpus respectively. This gives rise to a total number of 16 noisy scenarios, in each of which a DNN model is trained to discriminate speech-in-noise from pure noise for each type of feature.

In noise-independent training, 400 clean sentences are randomly chosen from TIMIT training set, with 50 sentences selected from each of the eight dialect regions. The noisy utterances are created by mixing each of the clean sentences with each of five types of noise, Cafe, Car, Factory1 , Ship oproom and Ship engine noises, at SNR levels varying from -6 dB to 12 dB in step of 2 dB. This creates a total number of 20,000 noisy utterances for training. The validation noisy utterances are obtained in a similar way, except that only 40 clean sentences are randomly selected from the TIMIT validation set, which gives rise to 2,000 utterances. As followed in noise-dependent training scenario, noise segments added to training and validation utterances are restricted to be selected from the first half of the recordings for noises from the NOISEX-92 database and the first recording for noises from the QUT-NOISE corpus. For constructing the noisy test sentences, apart from adopting the 16 test noisy scenarios created in noise-dependent training, similar noisy set is created for Ship engine noise and five other novel noise types, including Babble, Factory2, Tank-m109, F16 and Train noises. Since these five new types of noise are unseen in the training set, noise segments are selected randomly from the entire length of these recordings. Thus the total number of noisy test sets is 40, with 10 noise types each at 4 SNR levels (-5 dB, 0 dB, 5 dB and 10 dB).

4.4.2 Neural Network Structures and Training.

We first use a DNN to classify speech from non-speech frames given the calculated features, which consists of one input layer, three fully connected hidden layers each with 256 ReLU units and an output layer having two units and softmax

nonlinearity, which presents approximate probabilities of speech presence and absence. In order to capture context information effectively and efficiently, CNN is also adopted and we experiment with two different structures: single resolution CNN (CNN-SR) and multi-resolution CNN (CNN-MR). The single resolution-CNN has a standard CNN topology, whose convolutional layer is composed of 32 2-dimensional (2D) filters with size of 9×9, followed by ReLU activation function. The multi-resolution CNN also consists of 32 2D filters with ReLU nonlinearity, but each 8 filters of them have the following 4 resolutions: $5 \times 5, 9 \times 9, 15 \times 15$ and 23×23. Note that the sizes of multi-resolution filters are inspired by settings in MRCG feature, because of its remarkable performance as detailed in the following sections. Only valid convolution operation is used, meaning that no padding was added to either the horizontal or vertical boundary of features within a context window. Subsequently, maximum pooling along the frequency dimension only, with size of 2 and no overlapping, is performed in all CNNs, which effectively reduces the size of output from the convolutional layer by a factor of two. Finally, two fully connected hidden layers each with 256 ReLU units and an output layer with two units and softmax nonlinearity are added in both CNN-SR and CNN-MR.

Since generative model based pretraining is often found to be unnecessary, given a large enough training dataset, we initialize our models from scratch using initialization techniques proposed in [147]. Both DNNs and CNNs are trained with an adaptive learning rate method, Adadelta [60], to minimise the cross-entropy cost function given the predicted and ideal target labels. Mini-batch size is set to 256, and the maximum number of training epoch is 100. At the end of each epoch of training, the cross-entropy loss is measured on the validation set with the current model parameters. If the validation loss does not reduce for 10 consecutive epochs compared to previously saved lowest validation loss, the training process is terminated, a scheme known as early stopping. Moreover, dropout regularization technique is also employed in each hidden layer to deactivates its units randomly with a probability of 0.2. Both early stopping and dropout training contribute to increasing the generalization ability of trained neural networks and reduce their

degree of overfitting. All implementations of these neural networks and their training and testing are realized through the Keras deep learning toolbox [148] on top of Theano backend [149].

4.4.3 Comparison with Other Methods

To further understand the relative advantages of the systems investigated in this work, we also compare them with VAD methods that were developed in other studies. It worth noting that in many recent VAD researches, authors tend to compare their proposed methods with popular but not very strong baselines, including G729B VAD and statistical-model based VADs. In this work, we only consider two recently developed systems, the second one of which was shown to significantly outperform the traditional methods.

The first method is that proposed by Van Segbroeck et al. [150] as part of the Robust Automatic Transcription of Speech program of DARPA, as introduced in 4.2.2.2, and is thus referred to as RATS-VAD in this work. This method fuses a total number of four streams of features using a MLP classifier, including Gammatone Frequency Cepstral Coefficients (GFCC) [151], spectro-temporal modulations extracted by Gabor filters [152], harmonicity computed from correlogram [153] and a measure of the long-term signal variability [154]. For GFCC and spectro-temporal modulation streams, a separate MLP is trained for each of these two features, and the stream feature is simply the single output of each MLP, that measures the posterior probability of speech presence in the current observation. This is the so-called neural network based data-driven feature extraction that was initially proposed by TRAP features [155] for speech recognition. Speech segments are then determined by thresholding the ratio of the two outputs of the final merger MLP.

Another method is the VAD system developed by Thomas Drugman et al. [156] and included in the Covarep (A Cooperative Voice Analysis Repository for Speech Technologies) project [157], which is an open-source repository of advanced speech

processing algorithms. This method also utilizes multiple types of features which are classified as source-based and filter-based features, following the classical source-filter model of speech production and is referred to as SF-VAD in the following sections. Three types of filter-based features are considered, MFCC, PLP and the Chirp Group Delay of the zero-phase signal which is a high-resolved representation of the filter resonances [158], although only MFCC was chosen for the final system. Additionally, there are two sets of source-based features. The first set is that proposed by Sadjadi and Hansen [159] also for the RATS project, which includes 4 voicing measures that are called, harmonicity, clarity, linear prediction gain and the harmonic product spectrum. The second set of source-related features was not originally developed for VAD, but was selected for their robustness properties. It consists of the Cepstral Peak Prominence [160] which is a measure of the amplitude of the cepstral peak at the hypothesized fundamental period, and the Summation of the Residual Harmonics [161] that quantifies the level of voicing by taking into account the harmonics and inter-harmonics of the residual spectrum. Each feature type was fed to a specific MLP with a single output neuron with a sigmoid activation function producing the posterior probability of speech presence. The trajectories from all MLPs are further merged by geometrical mean to derive one final posterior speech probability. Note that this method is similar to the RATS-VAD algorithm in that it utilizes neural networks to derive data-driven features for final classification, although the final decision fusion is not carried out by a MLP.

4.4.4 Evaluation Metric.

The effectiveness of different VAD systems investigated in this work are compared using the area under the receiver operating characteristic (ROC) curve, i.e., the AUC. The ROC curves show the relationship between rates of correct speech detection or HIT rates and the rates of false speech acceptance or false alarm (FA) rates, for varying values of decision threshold. Thus an optimal VAD algorithm would achieve an AUC of 1. In order to obtain an operating threshold value for a

certain model in a specific application scenario, some optimization criterion needs to be applied, e.g., achieving equal MISS and FA rate or equal error rate, highest HIT-FA rate, or setting FA rate below a pre-defined value. In this work, we are only using the AUC metric to evaluate our models. Furthermore, no hangover scheme is used to post-process the outputs from different neural networks.

4.5 Results for Noise-dependent Training

In the noise-dependent training scenario, the test set fully matches with the training set in terms of noise type and SNR. In other words, there is one specifically trained neural network for each combination of feature type, noise type and SNR. Fig. 4.11 and Fig. 4.12 show all of the ROC curves for the six different spectrogram features with DNN backend under cafe, car, factory1 and ship oproom noises at four SNR conditions, -5 dB, 0 dB, 5 dB and 10 dB. Fig. 4.13 plots the AUC values in all of the 16 noisy environments and Fig. 4.14 plots the average AUC values at different SNRs across all four noise types for all of the spectrogram features tested. It can be seen that in almost all noisy scenarios, using the MRCG features yields the best performance in terms of AUC. The next two best performing features are either filter cascade or Gammatone spectrogram depending on noise type. For power normalized spectrogram, it can be nearly as effective as filter cascade and Gammatone features under factory1 noise, but it performs the worst in ship oproom noise condition. On average, MRCG remains the optimal feature type across all SNR levels tested; the filter cascade and Gammatone spectrogram features are roughly equally effective, so are the power normalized and Log-Mel spectrogram features. Finally, log-Power spectrogram produce the lowest speech detection accuracy in most noisy environments considered. Since the feature vector dimension of the single resolution Gammatone spectrogram is the same as that of MRCG, the extra performance improvements obtained from MRCG should mainly result from the multi-resolution analysis or the inclusion of contextual information. Although we can integrate this temporal context by simply feeding the DNN with feature vectors concatenated from multiple frames within a temporal window,

which is adopted in many other studies. This would result in a significant increase in the number of parameters in the input layer. The use of dimensionality reduction techniques could mitigate this issue, but also introduces higher computational complexity and possibly suboptimal integration of context information for the task at hand. Thus, in the following, we investigate various types of context expansion techniques based upon a more efficient and potentially more effective architecture, CNN. Compared to the hand-picked smoothing coefficients in MRCG, kernel or filter weights of CNN are directly learned from training data. It is thus reasonable to expect that CNN based methods could produce even higher speech detection accuracy than MRCG with DNN. Note that we do not perform experiments with CNNs in noise-dependent training scenario, because it is of limited practical importance since the training and testing conditions are fully matched. This is rarely the case in real environment.

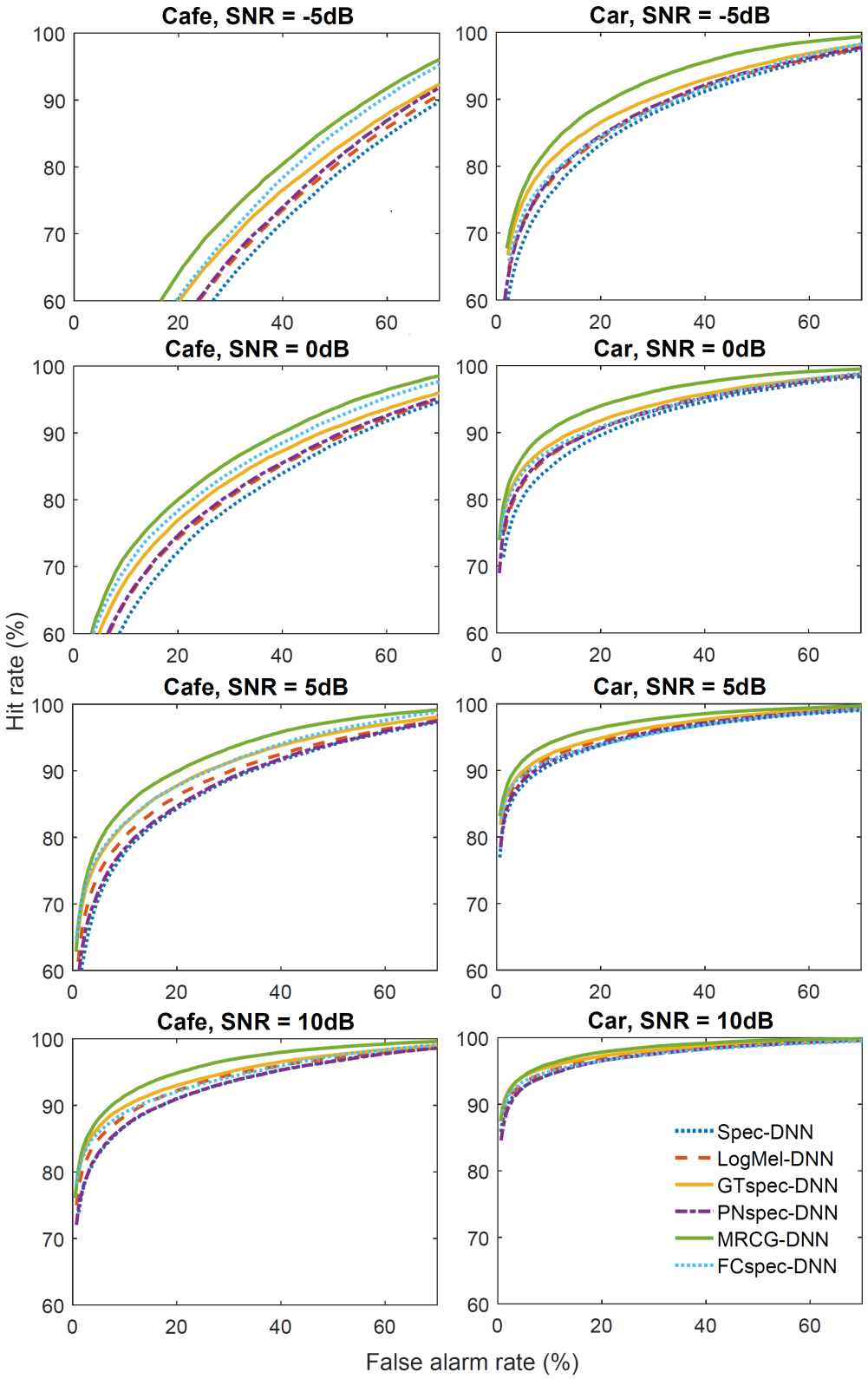


FIGURE 4.11: ROC curves for six different feature types with DNN back-end under cafe and car noises at four SNR conditions. Spec: FFT based Log-Power spectrogram, LogMel: Log-Mel spectrogram; GTspec: Gammatone spectrogram; PNspec: Power normalized spectrogram; MRCG: multi-resolution cochleagram, FCspec: filter cascade spectrogram. DNNs were trained noise-dependently.

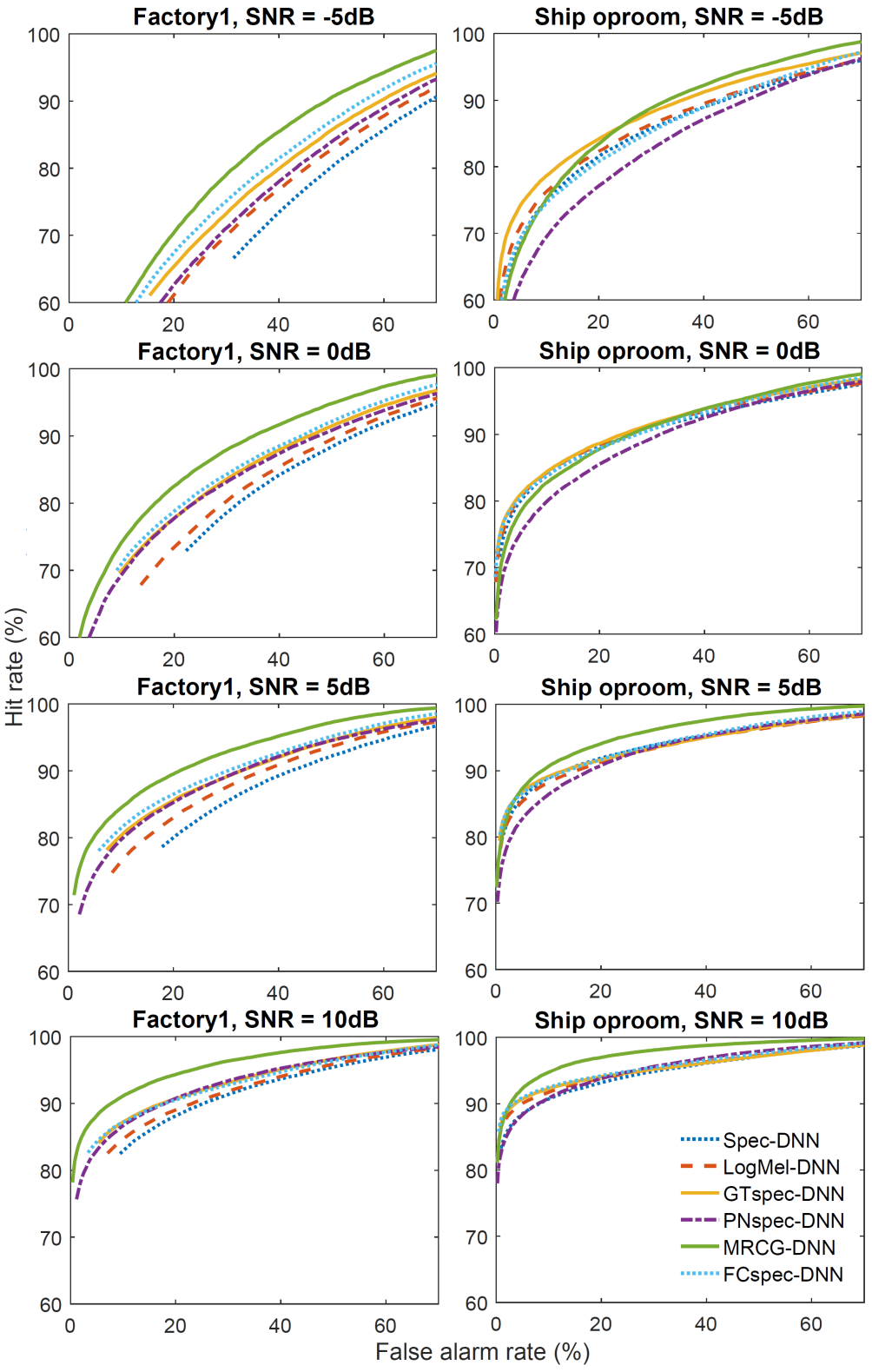


FIGURE 4.12: ROC curves for six different feature types with DNN back-end under factory1 and ship oproom noises at four SNR conditions. Spec: FFT based Log-Power spectrogram, LogMel: Log-Mel spectrogram; GTspec: Gammatone spectrogram; PNspec: Power normalized spectrogram; MRCG: multi-resolution cochleagram, FCspec: filter cascade spectrogram. DNNs were trained noise-dependently.

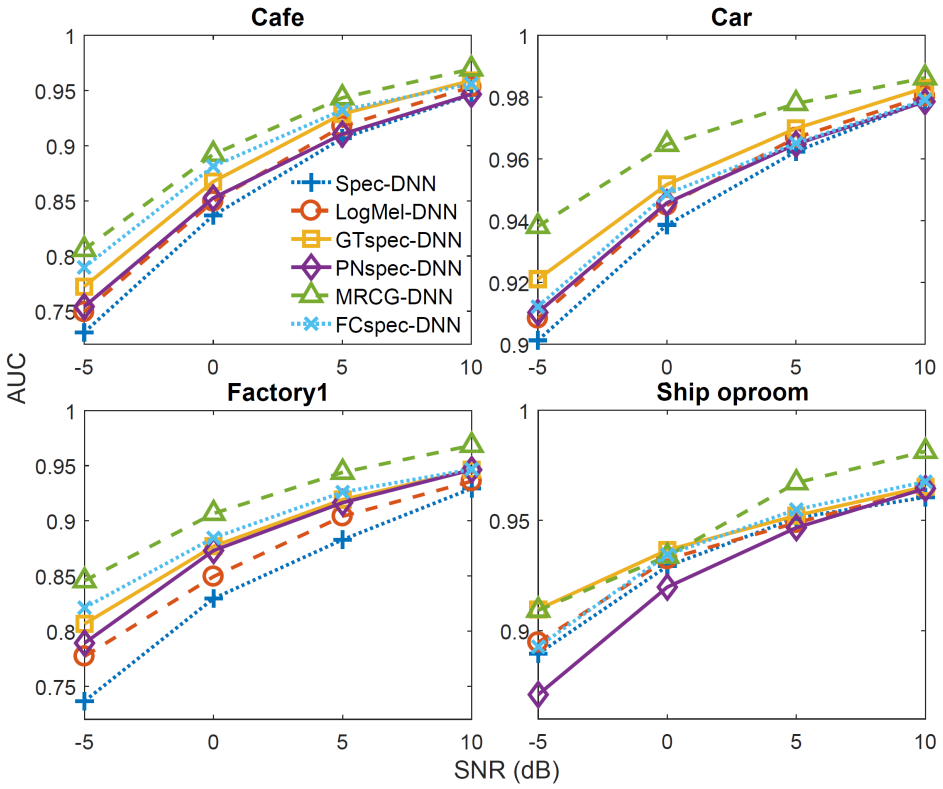


FIGURE 4.13: Comparison of AUC metric for six types of spectrogram features with DNN backend under four types of noise and four SNR levels. DNNs are trained noise-dependently. The legend follows those in Fig. 4.11 and Fig. 4.12.

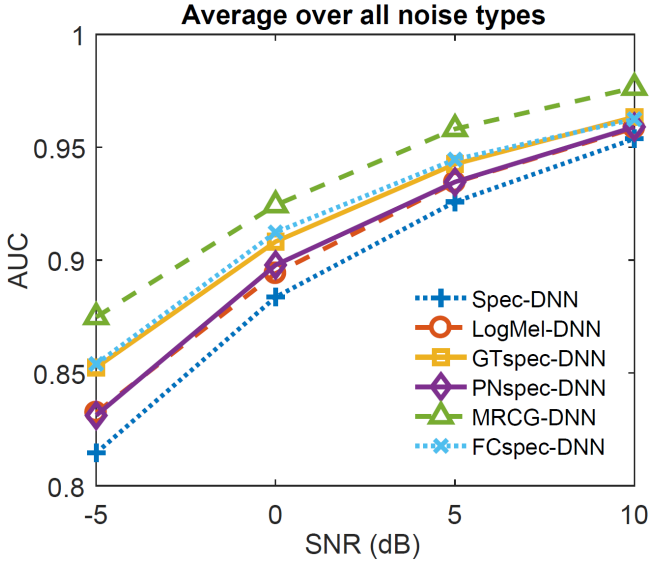


FIGURE 4.14: Comparison of average AUC metric for six types of spectrogram features with DNN backend at four SNR conditions. Average AUC values at each SNR were computed as the arithmetic mean of the results shown in Fig. 4.13 across all four noise types. The legend follows those in Fig. 4.11 and Fig. 4.12.

4.6 Results for Noise-independent Training

4.6.1 DNN Results

In this section, we present simulation results for the noise-independently trained neural networks, to investigate how well different features generalize to unseen noise conditions and how much they differ from their noise-dependently trained counterparts. This is of great significance for practical applications of such systems. Results from DNN models are presented first, followed by those from CNN models. We first test each model using utterances corrupted by matched noises, i.e., the same types of noise used for training of neural networks, although different random segment of each noise recording is chosen. Note that this is different from the noise-dependent training scenario, where one neural network is trained with only one noise type at one SNR level. In this case, a single network is trained with multi-condition dataset that includes multiple noise types at 10 SNR levels. Next, the model performances are evaluated under five novel and unseen noise types to determine their generalization capabilities.

Fig. 4.15 and Fig. 4.16 show the AUC metric for all of the spectrogram features with the DNN classifier under five matched and five unmatched noise scenarios respectively. The average AUC values across these matched and matched noise types are shown in Fig. 4.17. It can be seen that compared to the results obtained from noise-dependently trained DNNs, AUC values for all features decreases in almost all noisy conditions. This is not unexpected since the DNN is not specifically trained for a certain noisy type at a specific SNR any more. However, the MRCG remains the most effective feature type in both matched and unmatched noise testing conditions across all SNR levels. The filter cascade spectrogram can be nearly as effective as MRCG in matched Cafe, Factory1 and Ship engine noises, and is the second best performing feature type in most other noisy scenarios, especially at low SNR. It is interesting to observe that Gammatone feature performs more robustly than Power-normalized spectrogram in noise-dependent training condition, but in noise-independent training scenario, Power-normalized

spectrogram surpasses Gammatone and becomes the third most effective feature type. This probably results from the noise suppression module built in its implementation, which offers some advantages in varying noisy conditions. Finally, Log Power spectrogram, LogMel and Gammatone are the three least discriminative features for voice detection and on average, they are basically equal in terms of AUC metric. These experiments further confirm the benefits of including context information for VAD task.

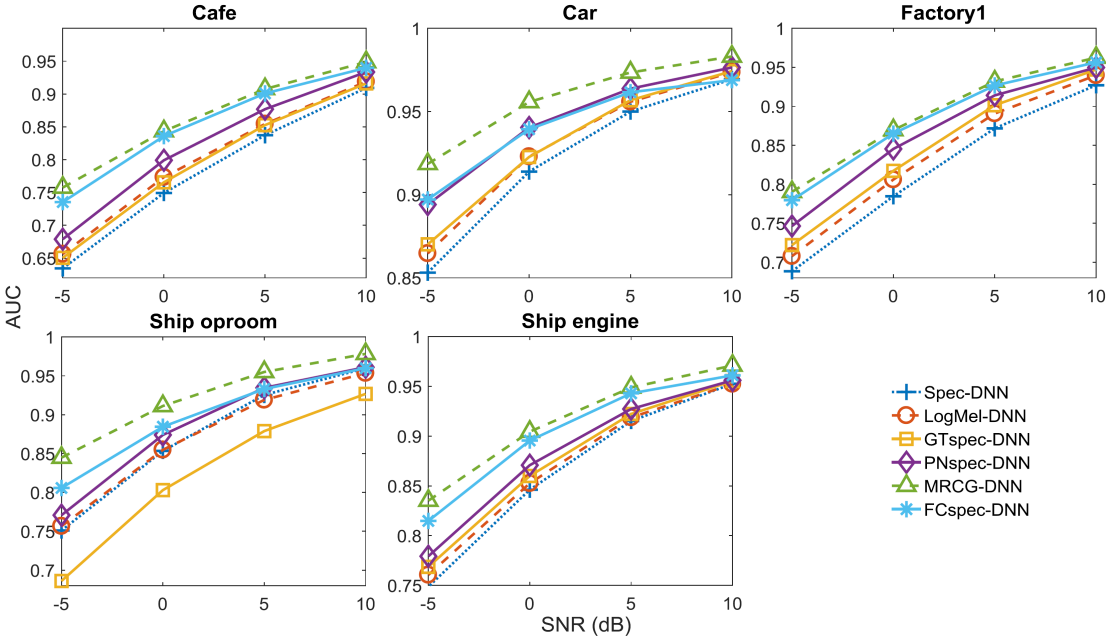


FIGURE 4.15: Comparison of AUC metric obtained from different spectrogram based features with DNN backend under (matched) noise testing conditions. The legend follows those in Fig. 4.11 and DNN is trained noise-independently or multi-conditionally.

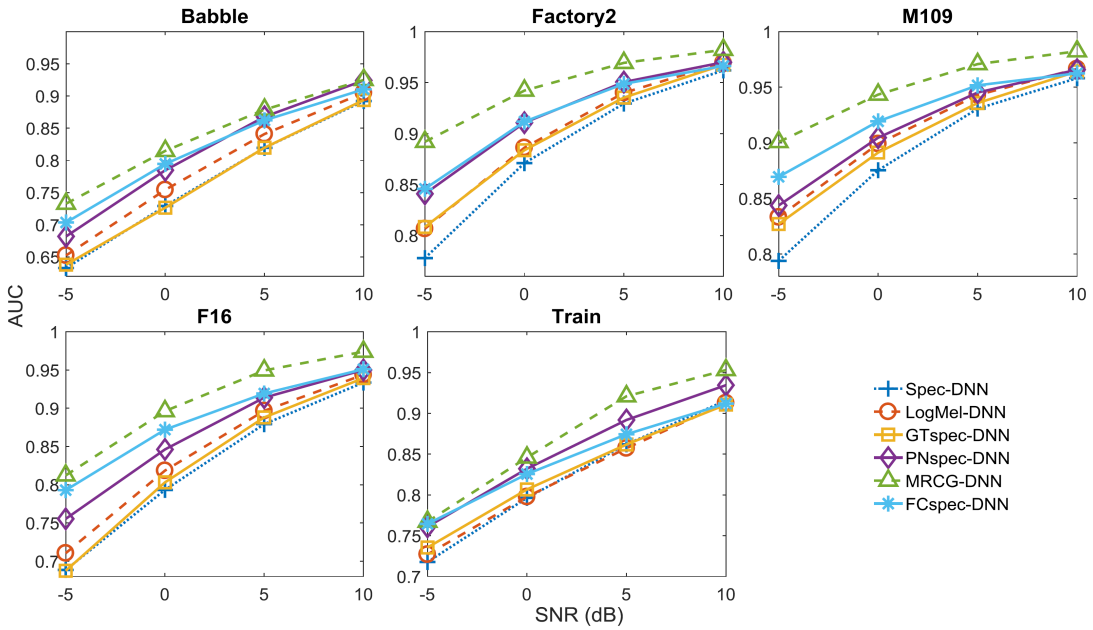


FIGURE 4.16: Comparison of AUC metric using different spectrogram based features with DNN backend under (unmatched) noise testing conditions. The legend follows those in Fig. 4.11 and DNN was trained noise-independently or multi-conditionally.

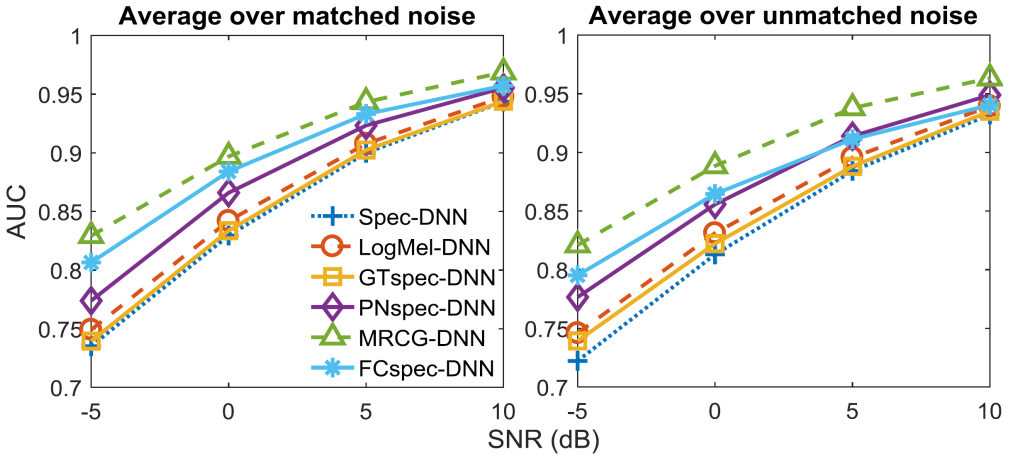


FIGURE 4.17: Average AUC metric across all (matched) and (unmatched) noise types, obtained from different spectrogram based features with DNN backend. Average AUC values at each SNR are computed as the arithmetic mean of the results shown in Fig. 4.15 and Fig. 4.16. Legend meanings follow those in Fig. 4.11 and DNN was trained noise-independently or multi-conditionally.

4.6.2 CNN results

Driven by the remarkable performance of the MRCG feature, it is interesting to investigate if a CNN based context expansion technique can produce further improvements in VAD performance. Because of its low average effectiveness and highest feature dimensionality, Log-Power spectrogram will not be considered in the following experiments. Fig. 4.18 and Fig. 4.19 show the AUC metric for four spectrogram features with the single resolution CNN classifier under five matched and five unmatched noise scenarios respectively. The average AUC values across these matched and matched noise types are shown in Fig. 4.20. In order to facilitate the investigation of relative advantages of CNNs, the results from MRCG feature with the DNN backend are also included in these figures. It can be seen that with single resolution CNN, the performances of all auditory based spectrograms have improved significantly in almost all noisy conditions, and become very close to those of the MRCG feature. In fact, several feature types have outperform MRCG in a number of noisy testing scenarios. For instance, in Cafe noise, the filter cascade spectrogram produces the most accurate VAD results at most SNR levels. In Factory1 noise, all feature types have surpassed the MRCG at all SNR levels, except for power normalized spectrogram at the lowest SNR, -5 dB. On average, we can observe in Fig. 4.20 that the filter cascade model produces higher AUC values than MRCG in matched noise conditions, while in mis-matched scenario, the filter cascade model still performs the best at low SNR levels, while at SNR of 5 dB and 10 dB, power normalized spectrogram becomes the most effective feature. Results from multi-resolution CNN are shown in a similar way in Fig. 4.21, Fig. 4.22 and Fig. 4.23. We can see that multi-resolution CNN is able to yield more improvements than single resolution CNN for all feature types, except for the filter cascade model. Because of this, simple auditory inspired spectrograms are able to outperform MRCG by a large margin in Car, Factory1 and Ship oproom noises. Gammatone spectrogram becomes the most discriminative feature type in nearly all ten noise types across all four SNR levels. It is surprising to see that multi-resolution CNN can actually degrade the performance of the filter cascade model to be even lower than that with DNN backend. Possible reasons for this

include overfitting during training and not enough filters in the convolutional layer. However, more simulations need to be performed to confirm this.

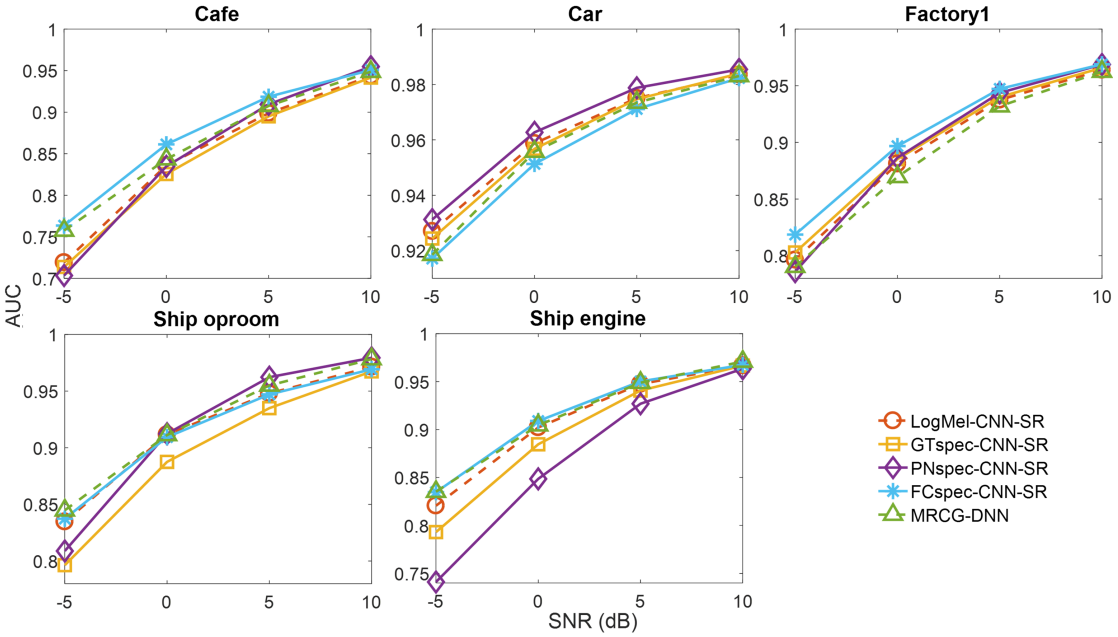


FIGURE 4.18: Comparison of AUC obtained from different spectrogram features with **CNN-SR** backend under **matched** noise conditions. Neural network is trained using multi-conditional dataset. Results from the MRCG with the DNN backend are also shown.

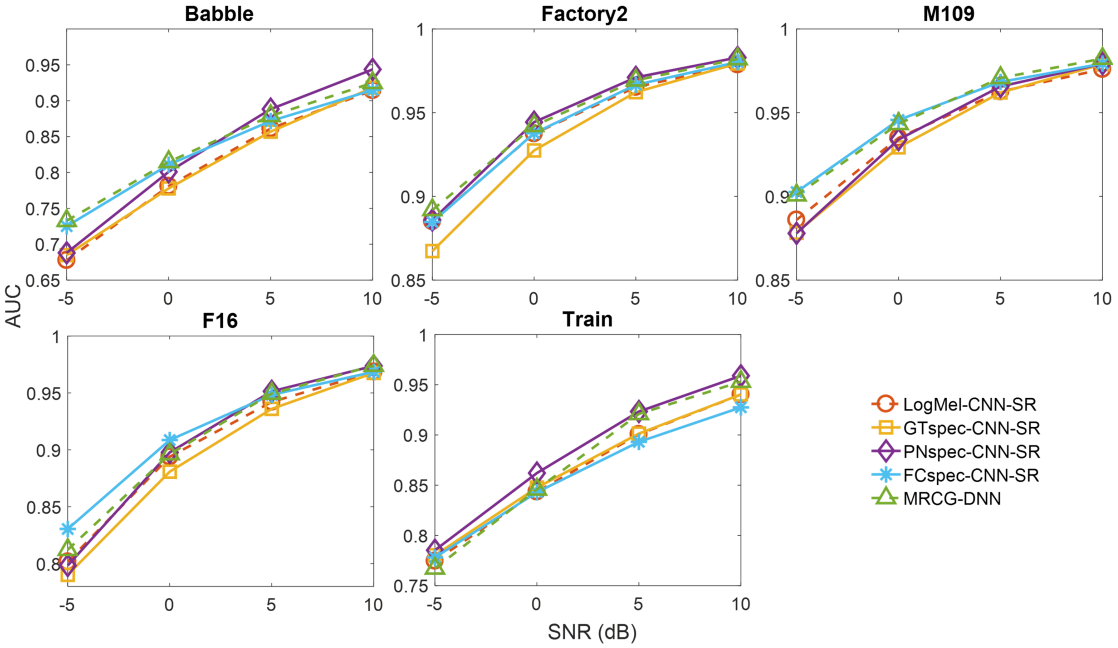


FIGURE 4.19: Comparison of AUC obtained from different spectrogram features with **CNN-SR** backend under **unmatched** noise conditions. Neural network is trained using multi-conditional dataset. Results from the MRCG with the DNN backend are also shown.

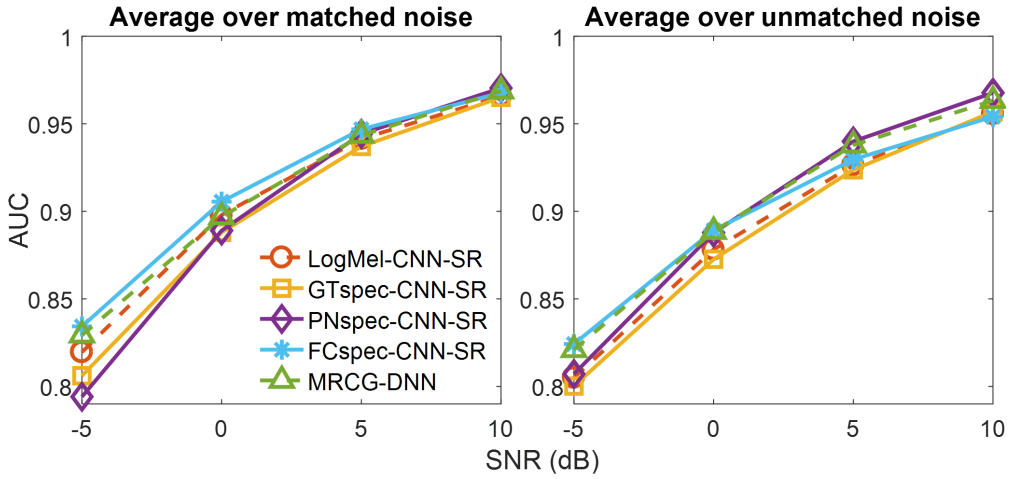


FIGURE 4.20: Average AUC metric across all **matched** and **unmatched** noise types, obtained from different spectrogram features with **CNN-SR** backend. Average AUC values at each SNR are computed as the arithmetic mean of the results shown in Fig. 4.18 and Fig. 4.19. CNN is trained using multi-conditional dataset. Results from the MRCG feature with the DNN backend are also shown.

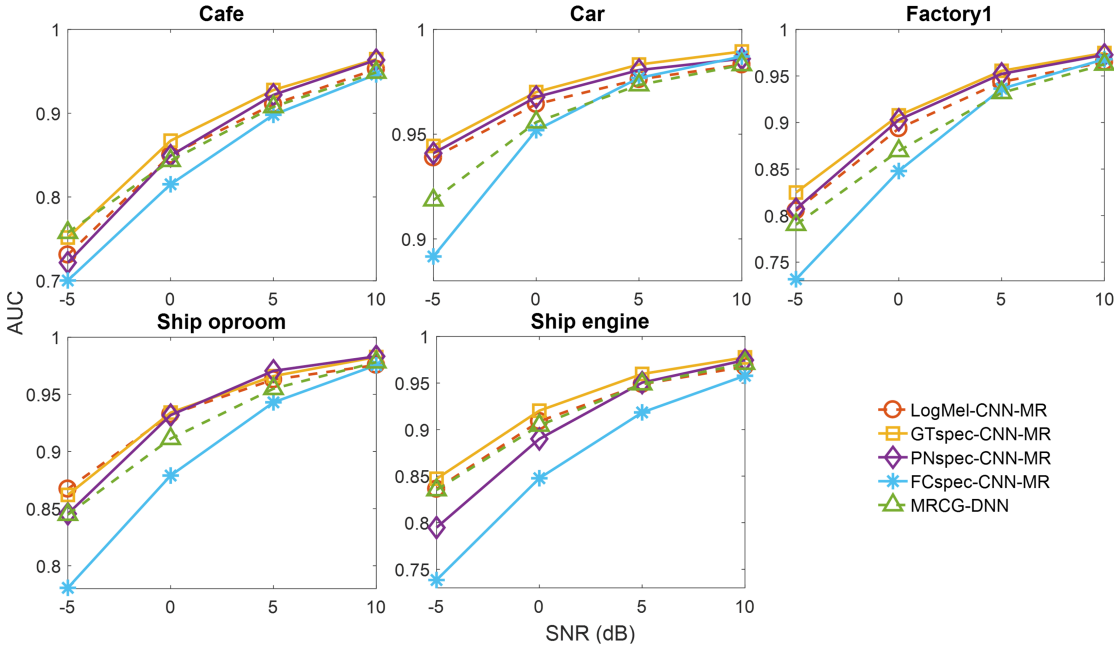


FIGURE 4.21: Comparison of AUC metric obtained from different spectrogram based features with **CNN-MR** backend under **matched** noise testing conditions. Neural network is trained noise-independently or multi-conditionally. Results from the MRCG feature with a DNN backend are also shown to facilitate comparison.

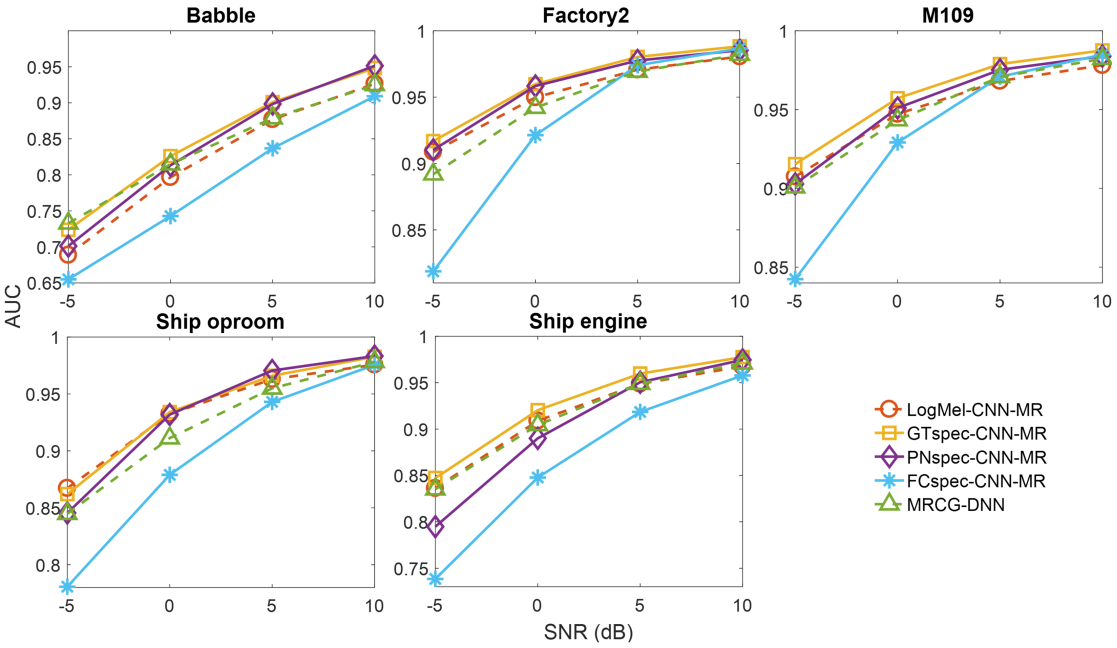


FIGURE 4.22: Comparison of AUC metric obtained from different spectrogram based features with **CNN-MR** backend under **unmatched** noise testing conditions. Neural network is trained noise-independently or multi-conditionally. Results from the MRCG feature with a DNN backend are also shown to facilitate comparison.

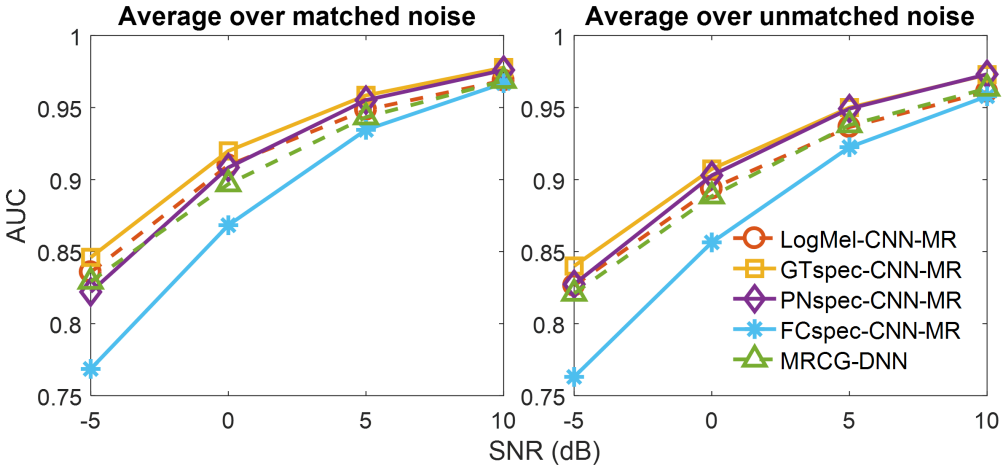


FIGURE 4.23: Average AUC metric across all matched and unmatched noise types, obtained from different spectrogram based features with **CNN-MR** backend. Average AUC values at each SNR are computed as the arithmetic mean of the results shown in Fig. 4.21 and Fig. 4.22. CNN is trained noise-independently or multi-conditionally. Results from the MRCG feature with a DNN backend are also shown to facilitate comparison.

Fig. 4.24 shows the average AUC metric across all matched and unmatched noise types and different SNR levels (computed from results presented above) for four auditory spectrogram features with three distinct neural network classifiers. The average result from MRCG feature with DNN backend is also shown for easy comparison. It is obvious that single resolution CNN (CNN-SR) significantly boosts the performance of all of the auditory spectrogram features, but average AUC values are still slightly lower than that from MRCG in combination with DNN classifier, except for the filter cascade model. The use of multi-resolution CNN (CNN-MR) produces further enhancements to LogMel, Gammatone and power normalized spectrograms, and makes them more robust than the very strong MRCG system. However, the performance of filter cascade model is seriously degraded with the use of CNN-MR.

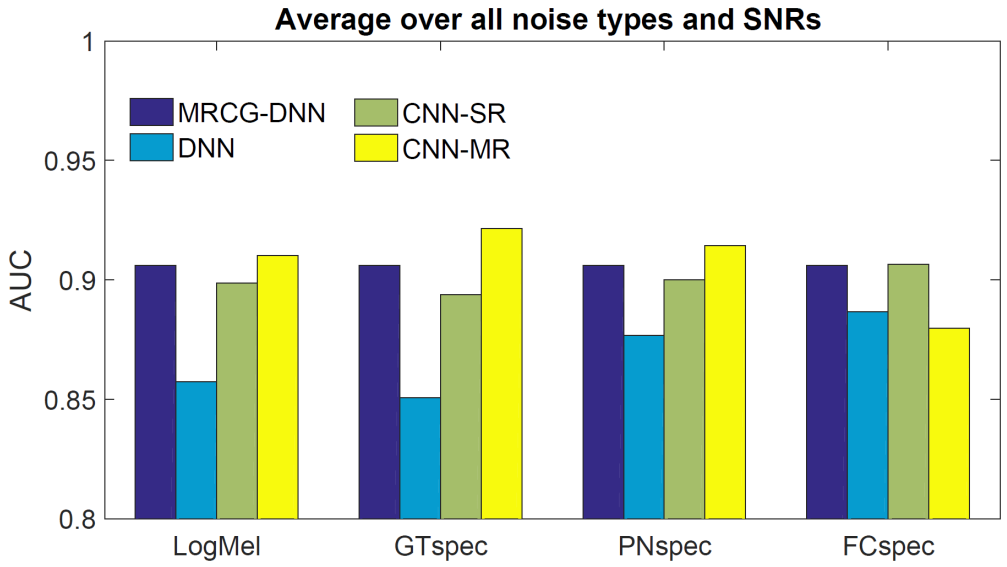


FIGURE 4.24: Average AUC metric across all matched and unmatched noise types and different SNR levels for four auditory spectrogram features with three neural network classifiers, DNN, CNN-SR and CNN-MR. Results from the MRCG feature with a DNN backend are also shown to facilitate comparison and all neural networks are trained noise-independently or multi-conditionally.

4.7 Further Improvements

Since the filter cascade model yields the lowest AUC value with CNN-MR, this motivates us to investigate alternative multi-resolution analysis strategies to close the gap between other feature types. In the above simulations, all spectrogram features except the MRCG use a single window size of 25 ms. In MRCG, a longer window size of 200 ms is also employed. Thus we propose a similar strategy using windows with multiple sizes to compute the spectrogram features. Specifically, four different window lengths are employed, including 25 ms, 50 ms, 100ms and 200 ms, and the resulting features are called multi-window-length Cochleagram (MWLCG), LogMel (MWLLM) and filter cascade (MWLFC) spectrograms. Note that these window lengths are loosely inspired by the ranges of temporal context captured by MRCG and CNNs discussed above. We have not performed this multi-window-length analysis with the Power-normalized spectrogram because its internal processing, such as noise suppression and temporal masking, is designed and optimized to the short window length of around 25 ms. Thus, the use of longer temporal window of 100 ms and 200 ms would seriously disrupt its representational power, similar to what has been observed above when increasing its channel size. To further confirm the relative advantage of CNN based multi-resolution analysis, the same mechanism adopted in MRCG is also applied to other features types, by replacing the Gammatone filterbank with another auditory filterbank. These features are referred to as MRLM and MRFC for LogMel and filter cascade spectrograms respectively. We then train noise-independent DNN, CNN-SR and CNN-MR models using these newly proposed features and test them in the same manner as described above.

Fig. 4.25, Fig. 4.26 and Fig. 4.27 show the average AUC value across five matched and five unmatched noise types obtained by applying various multi-resolution analysis methods to LogMel, Gammatone and filter cascade filterbank respectively. Further average results across all SNR levels are shown in Fig. 4.28. For LogMel and filter cascade filterbanks, the combination of multi-window-length spectrogram and CNN-MR classifier lead to the highest performance in both matched

and unmatched noise conditions. But for Gammatone filterbank, single-window-length spectrogram with CNN-MR remains to be the most effective combination for SNRs higher than 0 dB. From the average results in Fig. 4.28, it is interesting to observe that when applying the same multi-resolution analysis adopted in MRCG, VAD performance can also be improved for both LogMel and filter cascade models with DNN backend. But this becomes less accurate when CNN based classifiers are used, demonstrating the benefits of filter learning directly from the task at hand. Overall, the most robust strategy for LogMel filterbank is MWLLM with CNN-MR, which is slightly worse than the best strategy of Gammatone filterbank, i.e. GTspec with CNN-MR. Finally, MWLFC with CNN-MR is the most effective combination for VAD from all of the methods tested, although the difference between the top three strategies is not significant.

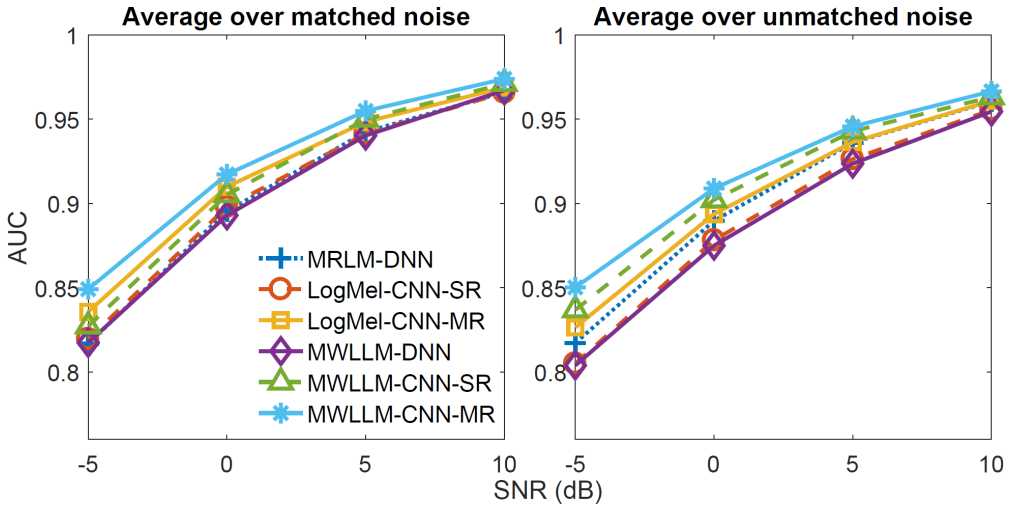


FIGURE 4.25: Comparison of various strategies in capturing context information with the LogMel filterbank in terms of AUC under **matched** and **unmatched** noises. MRLM means adopting the same analysis method as used in MRCG, but the Gammatone filterbank is replaced by a LogMel filterbank. MWLLM means the spectrogram feature is computed using multiple window lengths.

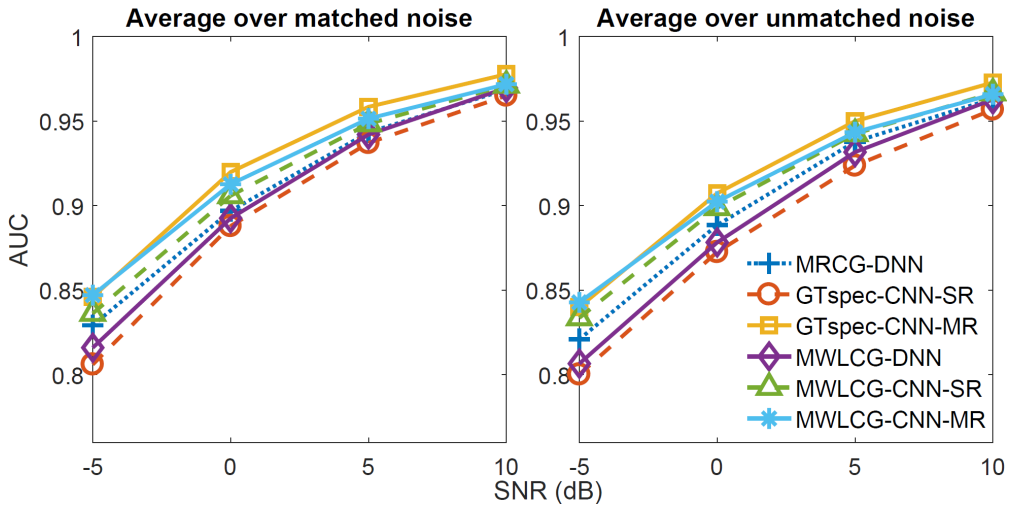


FIGURE 4.26: Comparison of various strategies in capturing context information with the Gammatone filterbank in terms of AUC under **matched** and **unmatched** noises. MWLCG means the cochleagram feature is computed using multiple window lengths.

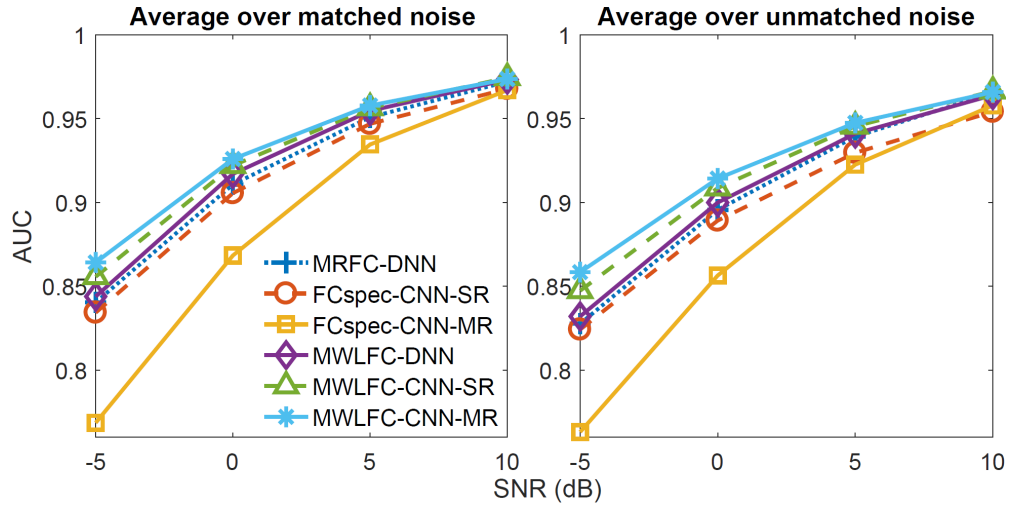


FIGURE 4.27: Comparison of various strategies in capturing context information with the filter cascade model in terms of AUC under **matched** and **unmatched** noises. MRFC means adopting the same analysis method as used in MRCG, but the Gammatone filterbank is replaced by a filter cascade filterbank. MWLFC means the spectrogram feature is computed using multiple window lengths.

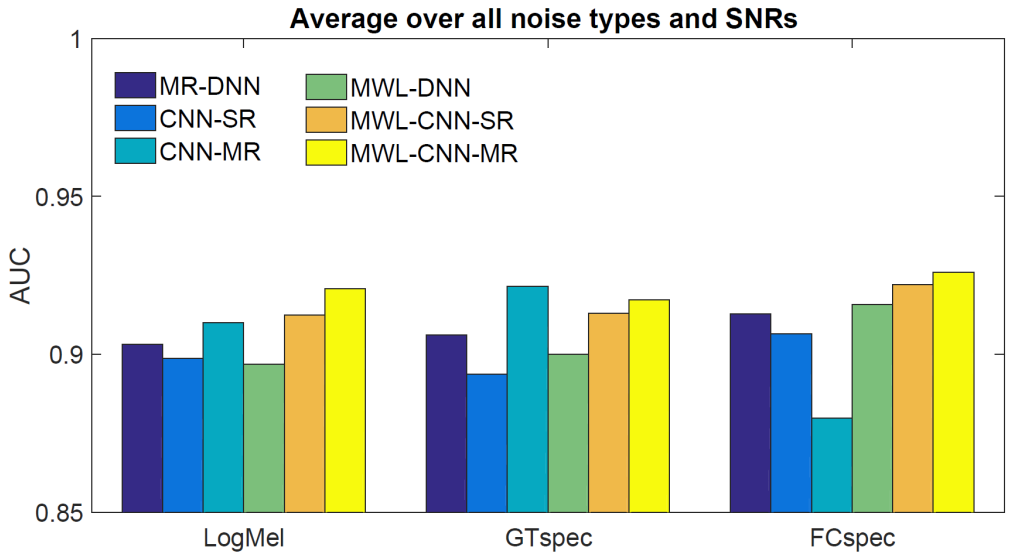


FIGURE 4.28: Average AUC values obtained by applying various context expansion strategies to three different auditory filterbank spectrogram features. The legend follows those in Fig. 4.24, Fig. 4.25, Fig. 4.26 and Fig. 4.27.

4.8 Comparison with Other Methods

Since both RATS-VAD and SF-VAD are also based on neural networks (although only shallow MLP) that are trained with training data, for a fair comparison, we test them under four noise types that are seen to none of them during training. Specifically, we choose Factory2, Tank-M109, Train and Speech-shaped (SSN) noises, where the first three noise types are the same as those used above for unmatched noise testing, while the SSN is taken from the Loizou noise database [146], but solely used in this section. Babble and Factory1 noises are not used because they were employed for training SF-VAD system as noted in the original paper [110]. For the DNN and CNN method presented above, we choose the best performing system that can be achieved with each of the four auditory filterbanks, as shown in Fig .4.24 and Fig .4.28, which are the following, MWLLM-CNN-MR, GTspec-CNN-MR, PNspec-CNN-MR, and MWLLFC-CNN-MR. Note that all of the noise types are not seen during training for these systems either. Fig. 4.29 shows the detailed results of all of the systems compared under four noise types and four SNR levels, Fig 4.30 shows the average results across different noises

and Fig. 4.31 shows the average results across all noise types and SNR levels. It can be seen that the RATS-VAD method performs significantly worse than all other methods in almost all noisy conditions. For the SF-VAD system, although it performs the best under Train noise at low SNRs (-5 dB, 0 dB and 5 dB), it becomes less effective than the systems proposed in this work in nearly all other noisy environments. On average, DNN and CNN based systems perform rather similarly especially at high SNR levels and are more accurate than the SF-VAD to some extent. This demonstrates the advantages of deep architectures over small and shallow networks. But it is interesting to observe that the RATS-VAD system is significantly less effective than other methods under comparison. One possible reason for such lack of accuracy is that this is not the optimal method among all of the systems developed for this project. We choose this one specifically, because of lack of implementations of other methods and the evaluation datasets. Furthermore, this RATS-VAD was trained with audio recordings transmitted through 8 different communication channels, the distortions induced in these datasets could be different from those that result from pure additive noise.

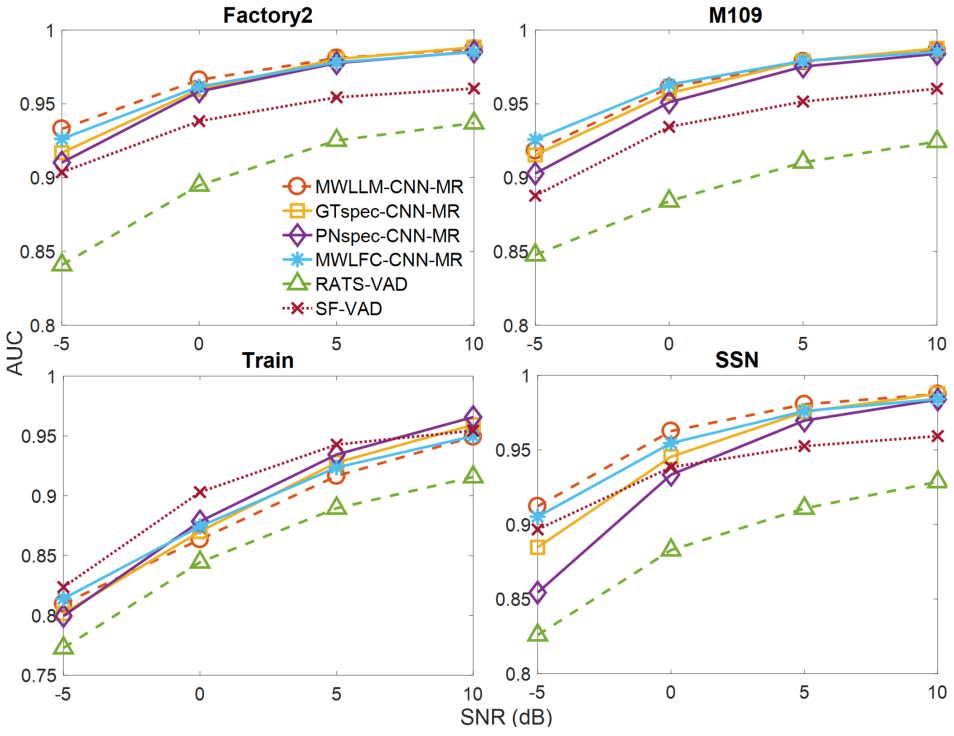


FIGURE 4.29: Comparison of the AUC values obtained by various VAD systems under four noise types and SNR levels. The legend follows those in Fig. 4.24, Fig. 4.25, Fig. 4.26 and Fig. 4.27.

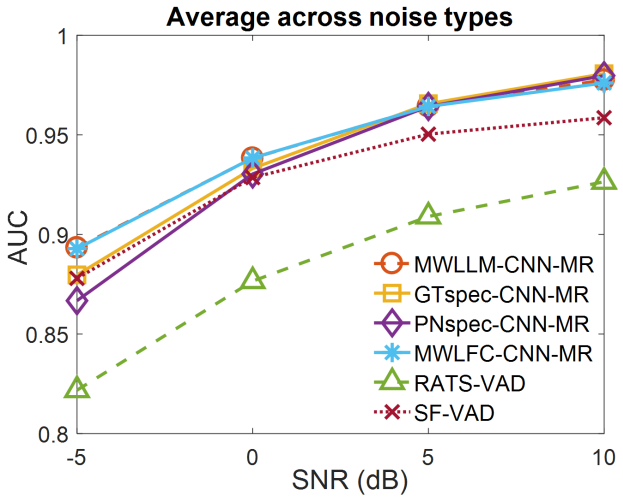


FIGURE 4.30: Comparison of the average AUC values across four noise types obtained by various VAD systems. The legend follows those in Fig. 4.24, Fig. 4.25, Fig. 4.26 and Fig. 4.27.

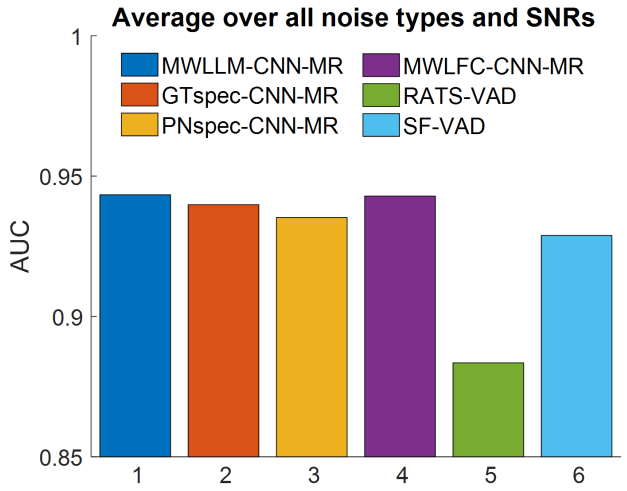


FIGURE 4.31: Overall comparison of the average AUC values across four noise types and four SNR levels obtained by various VAD systems. The legend follows those in Fig. 4.24, Fig. 4.25, Fig. 4.26 and Fig. 4.27.

4.9 Summary

In this chapter, the use of the filter cascade spectrogram has been investigated for neural network based voice activity detection, under a wide range of noisy scenarios. A number of other time-frequency representation based features have been investigated for comparison, and simulation results show that the performance

difference between them is generally small. Relatively large gains in detection robustness can be obtained by incorporating an appropriate form of temporal context information for each feature type. Two types of context expansion techniques have been investigated, one is the multi-window-length extension of spectrogram features and the other is CNNs, which includes two versions, one has fixed size 2D filters (single-resolution) and the other has the same number of 2D filter but with varying sizes (multi-resolution). It is found that CNNs are almost always more effective than DNN in speech and non-speech classification. But different features perform the best with different temporal expansion techniques investigated. Once the most suitable one is selected, all feature types perform rather similarly, although the filter cascade spectrogram possess a marginal advantage.

Two other state-of-the-art voice activity detection algorithms have also been compared with four of the systems proposed in this chapter. These are the ones with optimal context expansion settings for each feature type, as determined by the previous experiments. It is shown that these proposed systems also significantly outperform the reference methods.

Chapter 5

Cochlea Modelling for Supervised Speech Separation

In the previous chapter, various auditory inspired front-end processing methods are compared in terms of their effects on the VAD tasks. In this chapter, we further investigate their potentials for single-channel supervised speech separation under additive noisy conditions.

5.1 Introduction

Speech is often regarded as one of the most natural and effective way of human communication. However, speech signals can be easily corrupted by various interferences such background noise and undesirable reverberation, causing troubles to both human and machine listeners. The set of algorithms which aim to extract clean speech signals embedded in such sound mixtures are called speech enhancement or speech separation methods. In this chapter we investigate single-channel methods, that are based on neural networks and designed mainly for additive noise reduction. Speech de-reverberation is out of the scope of this thesis, but has been considered in many other studies, for instance [162]. For human listeners, these algorithms are usually evaluated in terms of improvements to the perceived

quality and intelligibility of the separated speech signal, while for machine hearing systems, these systems can be jointly optimized with a back-end stage, such as a speech recognition engine or a natural language processing engine, so as to optimize the complete system for the final task. In the following section, a brief overview of existing speech separation methods is presented first, with a particular focus on supervised learning based algorithms, which is the theme of this chapter. Readers are referred to [146] for a comprehensive and detailed review of traditional speech enhancement methods.

5.2 An Overview of Classical Speech Separation Methods

5.2.1 Spectral Subtractive Methods

Probably the most classical and popular type of method in speech separation are spectral subtractive methods. Assuming additive and uncorrelated interference, the general idea of these methods is to recover the clean speech spectrum by subtracting the estimated noise spectrum from the mixture spectrum. The noise-corrupted signal, $y(n)$ is written as,

$$y(n) = x(n) + d(n) \tag{5.1}$$

where $x(n)$ is the desired clean speech, $d(n)$ is the interference signal and n is discrete sample index. Taking discrete-time Fourier transform of both sides of Eq. 5.1 gives,

$$Y(\omega) = X(\omega) + D(\omega) \tag{5.2}$$

According to the derivations given in Chapter 5.1 of [146], the generalised form of spectral subtraction algorithm can be written as,

$$|\hat{X}(\omega)|^p = |Y(\omega)|^p - |\hat{D}(\omega)|^p \tag{5.3}$$

Or,

$$|\hat{X}(\omega)| = H(\omega)|Y(\omega)| \tag{5.4}$$

$$H(\omega) = \sqrt[p]{\left(1 - \frac{|\hat{D}(\omega)|^p}{|Y(\omega)|^p}\right)} \tag{5.5}$$

where $\hat{X}(\omega)$ and $\hat{D}(\omega)$ are estimates of the clean speech and noise spectrum respectively; p is a customisable power exponent, and popular choices are: $p = 1$, giving the magnitude spectrum subtraction and $p = 2$, giving the power spectrum subtraction; $H(\omega)$ is referred to as the gain or suppression function. To obtain time-domain signal, the estimated clean speech spectrum is usually combined with noisy phase, before using inverse Fourier transform and overlap and add method, for example. The choice of noisy phase is partly justified by its relative insignificance in affecting speech intelligibility [163].

Despite its conceptual simplicity and ease of implementation, spectral subtraction algorithm has several drawbacks. Since the noise spectrum has to be estimated, inaccuracies in its estimation may lead to negative values in the subtracted spectrum in Eq. 5.3. Simple solutions such as half-wave-rectification can create small, isolated and random peaks in the spectrum, giving rise to the well-known musical noise artefacts [164]. To cope with these problems, various improvements have been proposed to the basic spectral subtraction paradigm, such as spectral over-subtraction with flooring [165], nonlinear spectral subtraction [166] and adaptive gain averaging spectral subtraction [167].

5.2.2 Wiener Filter based Methods

Compared to the heuristics-based principles in spectral subtraction algorithms, Wiener filtering presents an optimal linear estimate of the complex clean speech spectrum in the sense of minimum-mean-square-error (MMSE). Assuming speech and noise signals are wide-sense stationary, uncorrelated and have zero means, the time domain finite-impulse-response (FIR) Wiener filter can be derived as,

$$\mathbf{h} = \left(\mathbf{R}_{xx} + \mathbf{R}_{nn}\right)^{-1} \mathbf{r}_{xx} \tag{5.6}$$

Here $\mathbf{R}_{xx}$ is the auto-correlation matrix of the clean speech signal, $\mathbf{R}_{nn}$ is the autocorrelation matrix of the noise signal and $\mathbf{r}_{xx}$ is the auto-correlation vector. If both past and future samples of noisy speech are used to estimate the current speech sample, the frequency domain Wiener filter can then be expressed as,

$$H(\omega) = \frac{P_{xx}(\omega)}{P_{xx}(\omega) + P_{nn}(\omega)} \tag{5.7}$$

where $P_{xx}(\omega)$ and $P_{nn}(\omega)$ is the power spectrum of the clean speech and noise signal respectively. It can be observed that since we have access to neither the clean speech nor the noise signal, neither their auto-correlations nor power spectra are directly available. However, numerous techniques have been proposed for estimating these quantities from the noisy speech signal in order to perform Wiener-filter based speech enhancement [168, 169, 170].

5.2.3 Statistical Model based Methods

As noted in Chapter 4, various statistical distributions have been employed to model the speech and noise spectrum in order to determine speech presence likelihood. In fact, these statistical models were first applied to the speech enhancement problem, which then inspired their usage in VAD methods. The concept of *a priori* and *a posteriori* SNRs and their estimations were first introduced and also play

a central role in the field of speech enhancement. As its name suggests, these methods pose speech enhancement in a statistical estimation framework, and two popular techniques borrowed from the estimation theory are maximum-likelihood and Bayesian estimation. Compared to the Wiener filter introduced in the previous section, these estimators are nonlinear.

In maximum-likelihood estimation, the clean speech features we want to estimate, for instance magnitude and phase spectra, are assumed to be unknown but deterministic. In Bayesian methods, however, the parameter of interest is assumed to be a random variable with some *a priori* distribution and the estimation is performed based upon the Bayes' theorem. Typically, Bayesian estimators perform better than ML estimators, because they make use of prior knowledge. They have been proposed in various forms, and mainly differ between each other in terms of the target they aim to estimate (magnitude and phase, or real and imaginary parts of complex STFT), the optimization criterion employed (MMSE or maximum a posteriori) and speech prior distribution assumed (Gaussian, Gamma, Rayleigh etc.). For example, Ephraim and Malah [124] developed the MMSE STFT amplitude estimator by using Rayleigh distribution prior, considering the importance of magnitude relative to phase in speech perception.

Despite their promising results, the theoretical limitations of these estimators are that they are only optimal when the assumed statistical models are true and the speech and noise spectral variances are known. These can be easily violated in real data, and hence most of the statistics-based methods suffer to some extent from inadequate noise suppression (particularly for fast varying non-stationary noises) or introducing annoying artifacts in the recovered signal [171].

5.3 Supervised Speech Separation

Recent development in supervised speech separation originates from the field of computational auditory scene analysis (CASA), which utilizes auditory principles

to analyse and understand complex acoustic environments in order to achieve near-human performance. Based on results from psychoacoustic research, the main goal of CASA systems was proposed to be the so-called ideal binary mask (IBM) [172], due to its ability to yield substantial speech intelligibility improvements for noisy speech for both normal-hearing and hearing-impaired listeners [173, 174, 175]. It is defined using premixed target clean speech and interference signals. Specifically, with a time-frequency (T-F) representation of a sound mixture, each T-F unit is either preserved, with a mask value of one, or discarded, with a mask value of zero, depending on the unit-level SNR and a local criterion. Mathematically, IBM is expressed as the following,

$$IBM(t,f) = \begin{cases} 1, & if \quad SNR(t,f) >= LC \\ 0, & otherwise \end{cases} \tag{5.8}$$

where $SNR(t,f)$ denotes local SNR (in decibels) within the T-F unit at time frame t and frequency bin or channel f; LC is the local criterion used to determine whether this unit is speech-dominant or interference-dominant. This effectively formulates speech separation as a binary classification problem, opening up opportunities for a wide range of increasingly powerful machine learning techniques, especially deep architectures as reviewed in Chapter 2. It worth noting that the concept of ideal spectral masking is similar to various optimal spectral gain functions or the suppression rules that many traditional methods aim to estimate, as reviewed in section 5.2. The main difference between them is that classical approaches rely heavily on statistical assumptions on the properties of clean speech and noise signals, whereas supervised speech separation directly learns such transformation functions from training data, with little or no such assumptions. Fig. 5.1 shows a comparison of the block diagrams for these different approaches.

Although the original IBM-based supervised systems have shown great success in improving the intelligibility of separated speech, they often results in poor perceptual quality, due to estimation errors and hence the removal of speech units. To further improve their performances, various alternative training targets have been

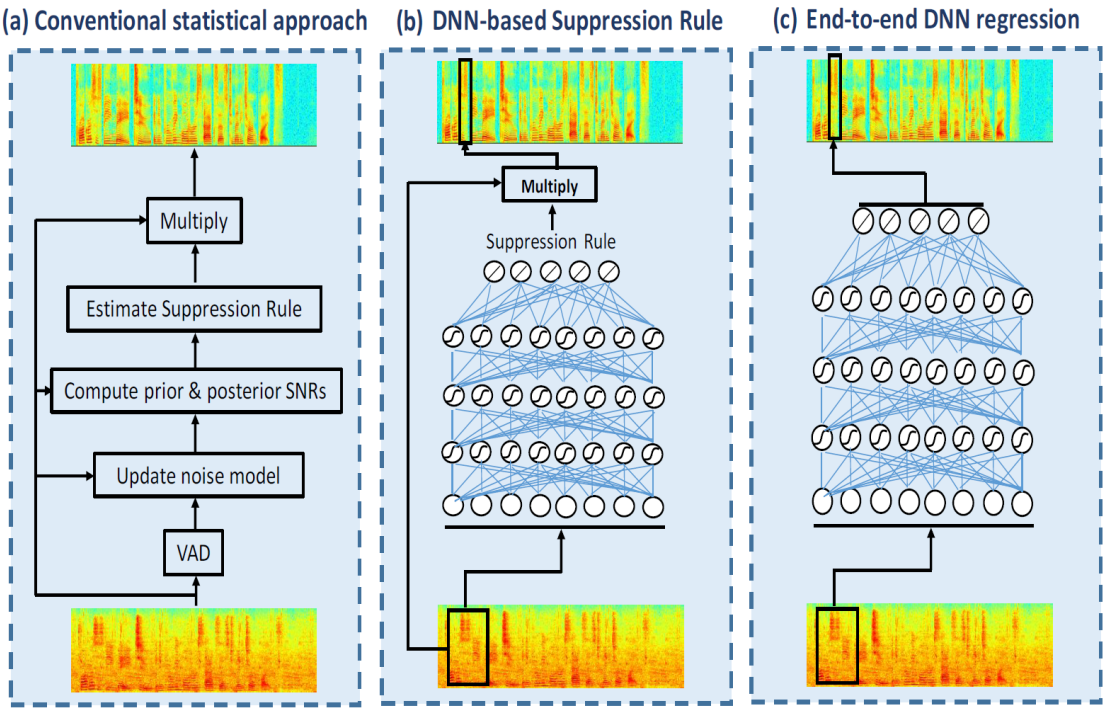


FIGURE 5.1: Comparison of the (a) conventional statistics-based method and (b, c) two types of DNN-based methods for speech separation or enhancement. Adapted from Fig. 1 of [171] with permission.

explored, and depending on the specific choice of those targets, supervised separation methods can be classified into two categories: (i) masking-based methods and (ii) mapping- or regression-based methods, as also illustrated in Fig. 5.1.

As introduced above, masking-based methods aim to produce a time-frequency mask, such as the IBM, and then use the estimated mask to separate the mixed signal, usually by multiplying with the noisy spectrum, as indicated in Fig. 5.1 (b). Wang *etal*. [176] examined a numbers of training targets and proposed an ideal ratio mask (IRM) that should be preferred over the IBM in terms of speech quality. This IRM is computed from the ratio of target speech energy and mixture energy, and can be defined as in Eq. 5.9 [176], following the notation used in section 5.2.1 and Eq. 5.8. But other formulations of the ratio mask are also possible, such as Eq. (8) in [177]. Instead of enhancing only the magnitude spectrum, as IBM and IRM do, Williamson and Wang [178] proposed to jointly enhance the magnitude and phase response of noisy speech by estimating the complex ideal ratio mask (cIRM) that predicts both the real and imaginary components of

complex STFT spectrogram of clean speech in the Cartesian coordinate system. Results show that it substantially outperforms the magnitude-only IRM, probably because cIRM allows a full reconstruction of clean speech in the ideal case.

$$IRM(t,f) = \sqrt{\frac{X^2(t,f)}{X^2(t,f) + D^2(t,f)}} \tag{5.9}$$

Mapping-based methods, however, learn a regression function that transforms noisy speech features to clean speech features directly, Fig. 5.1 (c). For instance, Xu *etal*. [179] proposed to use a DNN as a regression machine to estimate the clean magnitude spectrum from the noisy magnitude spectrum. Subsequently, the approach was extended with noise-adaptive training [180] and global variance equalization [181] to alleviate the distortions in estimated clean magnitude spectrum. Since the main goal of this chapter is to investigate the benefits of advanced modelling of cochlear nonlinearity in supervised speech separation, we only choose the IRM as the training target in the following experiments, for its simplicity and proved effectiveness [176], although it worth bearing in mind that more advanced masks could change the simulation results presented below.

Apart from training target, there are another two fundamental elements in supervised speech separation systems, i.e. feature extraction and backend machine learning model, as shown in more detail in Fig 5.2. As in general machine learning tasks, feature extraction is normally the first stage of the signal processing chain. Early studies [182, 183] adopted pitch-based features as harmonic structure is a prominent characteristic in voiced speech. For unvoiced speech separation, spectrogram statistics [184] and amplitude modulation spectrum (AMS) [185, 186, 187] were later incorporated. To further improve speech separation robustness, various features that were originally developed in the fields of speech and speaker recognition were also investigated, including MFCC, PLP, RASTA-PLP, GFCC, PNCC. Using Gaussian-kernal SVMs as subband classifier, they are shown to outperform the earlier pitch and AMS features in terms of classification accuracy and HIT-FA rate [188]. A complementary feature set that concatenates AMS, RASTA-PLP

and MFCC feature vectors has also been identified as the optimal feature combination of all of the feature types tested for IBM estimation based on a group lasso approach, and is also shown to perform the best in most of test conditions [188]. In [119], a significantly more extensive list of features were tested and the MRCG was shown to achieve the best separation performance among all of them, even better than the complementary feature set recommended in [188]. However, it is worth noting that features were compared in [119] using the same type of noise for training and testing, it is thus not clear how well can MRCG generalize to unseen noise types for supervised speech separation, although it is indeed found to do this effectively in VAD task as presented in Chapter 4. Since only a MLP was adopted as a binary mask classifier, it will be worthwhile to investigate how MRCG will interact with deep architectures, such as DNN and CNN, for speech separation, as also pointed out in [119]. Finally, since the primary evaluation metric adopted is the HIT-FA rate for IBM estimation, the relative advantages among various features could change when different masks and speech separation performance metrics were used.

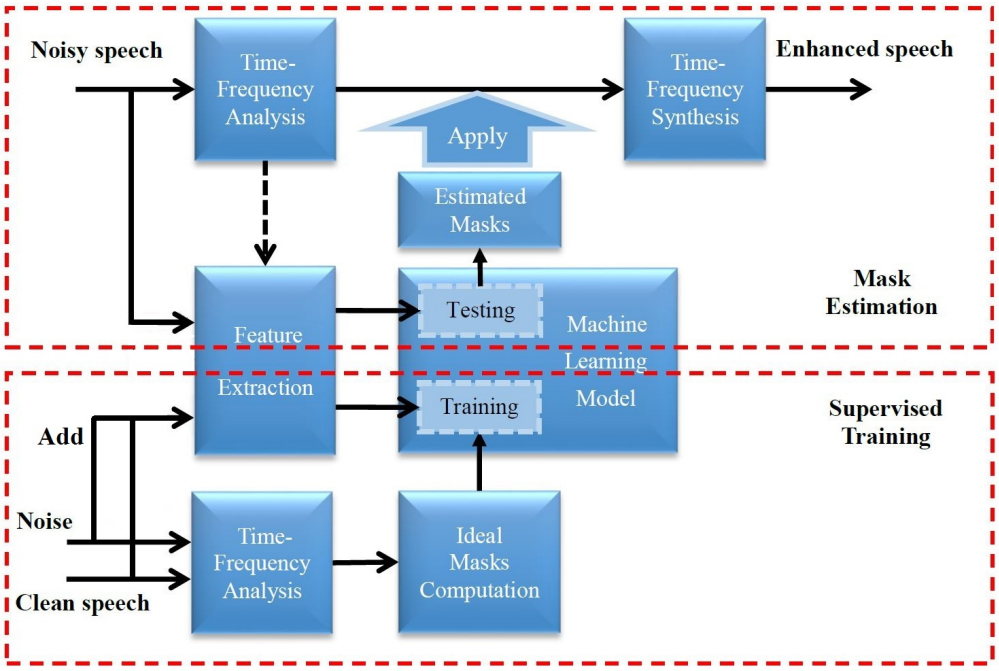


FIGURE 5.2: Block diagram of a typical supervised speech separation system. Common choice for Time-Frequency analysis and synthesis is the Gammatone filterbank. The dashed line means that the Time-Frequency analysis representation can sometimes be used in Feature extraction module as well.

The final consideration is the choice of backend machine learning. Early studies used generative models such as Gaussian Mixture Model [187] or shallow discriminative models such as MLP[119] and Support Vector Machines [188] for binary mask classification, but their practicability is somewhat limited due to the lack of generalization ability of these systems. Motivated by their success in acoustic modelling for speech recognition in the last few years, recent researches focus on deep models, including DNN, CNN, LSTM-RNN and BLSTM-RNN and they were demonstrated to be superior to earlier systems by a large margin, even when tested in mismatched noisy conditions. In formal listening tests, deep models enhanced speech shows substantial improvements in speech intelligibility for both normal-hearing and hearing-impaired listeners. For these reasons, deep models are adopted in the following experiments for comparing various auditory inspired time-frequency representations for supervised speech separations.

5.4 Filter Cascade Spectrogram for Supervised Speech Separation

5.4.1 Feature Extraction

The feature extraction procedure is basically the same as that adopted in section 4.3. Apart from the filter cascade filterbank (FCspec), four other time-frequency representations are also considered for comparison, including FFT based Log-Power spectrogram (Spec), Log-Mel spectrogram (LogMel), Gammatone spectrogram or cochleagram (GTspec) and MRCG. The Power normalized spectrogram has also been investigated but it yielded unacceptable performance compared to other features and hence is not considered. Different spectrogram features are computed using an analysis window of 25 ms (or 400 samples) with a window shift of 10 ms (or 160 samples) and feature configurations follow those detailed in section 4.3. Moreover, multi-window-length extensions of LogMel (MWLLM),

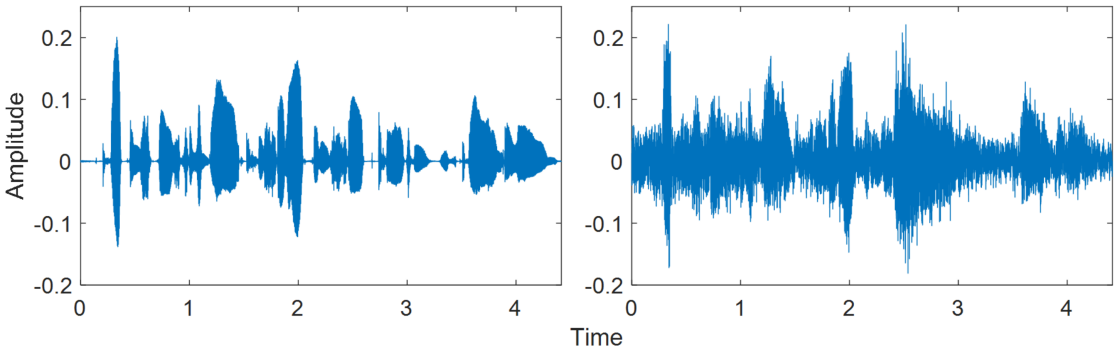


FIGURE 5.3: An example clean utterance from the TIMIT dataset and the same utterance corrupted by a factory noise from the NOISEX dataset at 0 dB SNR.

Gtspec (MWLCG) and FCspec (MWLFC), as introduced in section 4.7, are also included.

It is worth noting that the MRCG feature has been used in both speech detection and speech separation tasks, but different settings were followed in these applications. In a speech detection task [6], one set of 8-channel Gammatone filterbank was used to construct each of the four resolutions, while in speech separation, the authors in [119] proposed to use four sets of 64-channel Gammatone filterbanks (referred to as MRCG-64 in this chapter), making the dimensionality of static feature vector to be $64 \times 4 = 256$. In this work, apart from using the original implementation for speech separation, we also considered an variant that employs only 16 Gammatone filters in each resolution, resulting in a static feature vector having the size of 64. This configuration is called MRCG-16 in this chapter and is the same MRCG feature setting as adopted in the speech detection investigation presented in Chapter 4. It is included here for the same reason: to reduce the difference in feature dimensionality between various spectrogram features. Fig. 5.4 shows the MRCG feature with 256 channels for a clean utterance from the TIMIT dataset [143] and the same utterance corrupted by a factory noise from the NOISEX dataset [144] at 0 dB SNR, the waveforms of which are displayed in Fig. 5.3. It can be seen that the 256-channel MRCG shows much more spectral details than its 64-channel counterpart, Fig. 4.7, but at the cost of nearly four times of computational complexity. Since visualizations of all other spectrograms features have been shown in Chapter 4, Fig. 4.6 and Fig. 4.7, they are not repeated here.

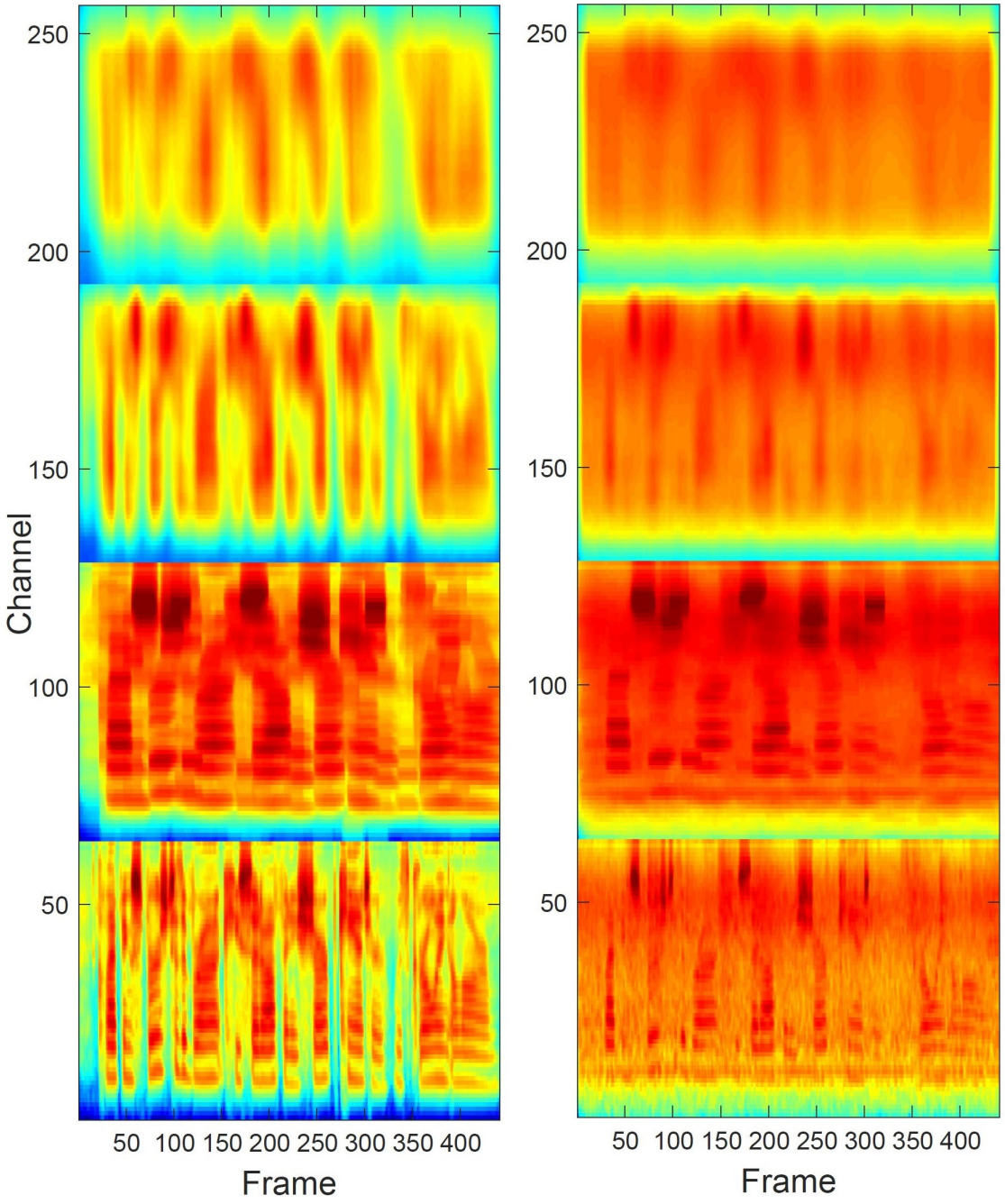


FIGURE 5.4: Visualization of MRCG representation of a clean utterance from the TIMIT dataset (right column) and the same utterance corrupted by a factory noise from the NOISEX dataset at 0 dB SNR (left column).

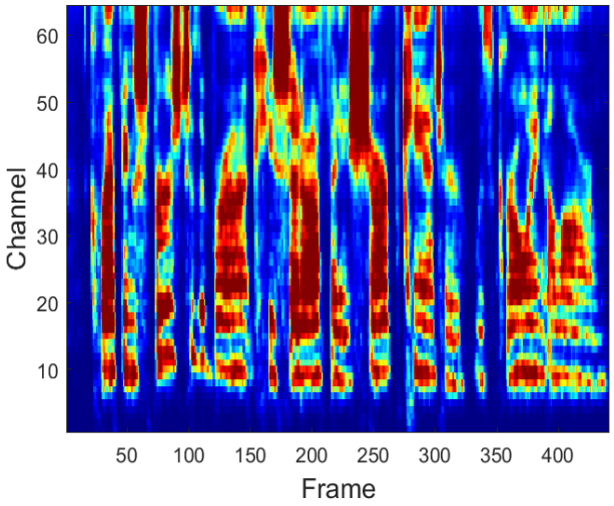


FIGURE 5.5: The Ideal Ratio Mask for estimating clean speech signal shown at the left hand side of Fig. 5.3 from its noisy counterpart shown at the right hand side of Fig. 5.3.

As mentioned above, the IRM is used as the training target for neural network acoustic models in this work. Fig. 5.5 shows an example of such an IRM for the clean utterance and its factory noise corrupted counterpart, as illustrated in Fig. 5.3, which has been computed using Eq. 5.9. Note that IRM can be computed in different domains, such as Fourier transform and Gammatone domain. In this work, we follow the practice adopted in [176], which formulated the IRM using a 64-channel Gammatone filterbank.

5.4.2 Experimental Settings

5.4.2.1 Datasets

In a similar way to the settings used in Chapter 4, clean speech utterances are selected from the TIMIT [143] database. Training sentences consist of 1,600 randomly chosen utterances, with 200 utterances from each of the 8 dialect region. Another 160 utterances, with 20 from each dialect region, are also randomly selected from the TIMIT validation set (original test set excluding the core test set), to form the validation dataset. All input clean speech signals are sampled at 16 kHz and rescaled to 60 dB SPL before mixing with various types of noises at a number of SNR levels.

A total number of 10 noise types are selected from three noise databases, which are the NOISEX-92, the QUT-NOISE and the Loizou noise corpora, as shown in Fig. 4.10. From these, five types of noise, including Cafe, Car, Factory1, Ship oproom and Ship engine noises, are used to create a multi-condition training set, by mixing each of the training utterances with a random segment of each noise type at two SNR levels of -5 dB and 0 dB, which gives rise to 16,000 noisy sentences in total. The validation dataset is obtained in a similar way, except that only 160 clean utterances are used, resulting in 1600 noisy sentences. For testing, apart from using the five noise types that are employed in training, the other five types of unseen noises are also adopted, consisting of Babble, Factory2, Tank-m109, F16 and Train noises. This sets up the matched and unmatched testing scenarios. These 10 types of noise are mixed with the 240 core test set of the TIMIT database at four SNR levels of -5 dB, 0 dB, 5 dB and 10 dB, where 5 dB and 10 dB are two unseen SNR conditions. Note that only noise-independent training and testing are performed for this speech separation task, because noise-dependent training is of limited importance for practical applications.

5.4.2.2 Neural Network Structures and Training

Both DNN and CNN architectures are used to estimate the IRM given calculated feature vectors. The DNN is composed of one input layer, three fully connected hidden layers, each with 1024 ReLU units and one output layer with sigmoid units. For the CNN, both single resolution (CNN-SR) and multi-resolution (CNN-MR) structures are adopted. The convolutional layer of CNN-SR consists of 64 2-dimensional (2D) filters with size of 9×9, while that of CNN-MR also has 64 2D filters but with each 16 of them having the following 4 resolutions: $5 \times 5, 9 \times 9, 15 \times 15$ and 23×23. Valid convolution was performed between filter weights and input features. Subsequently, after ReLU nonlinear activation and maximum pooling along the frequency dimension with size of 2 and no overlapping, two fully-connected hidden layers each with 1024 ReLU units and one output layer with sigmoid units are added in both CNN-SR and CNN-MR. It can be seen that network structures employed in this chapter are the same as those used in Chapter

4, except that networks sizes are increased to tackle the more challenging task of IRM estimation (i.e. larger dimensionality of the target).

The network training procedure is essentially the same as that present in section 4.4.2, except the following changes: (a) mean-square-error (MSE) between the predicted ratio masks and IRMs is used as the cost function. (b) The maximum number of training epoch is set to 300 and the patience for early stopping is set to 30, i.e., training is stopped if the validation loss does not reduce for 30 consecutive epochs. Note that these settings are larger than those used in VAD tasks as presented in Chapter 4. (c) Finally, as suggested in [176] for performance improvement, a temporal context of 5-frame of features, centred around the current frame, are concatenated as input to the DNN, which are in turn trained to predict the corresponding 5-frame of targets at the same time. The multiple target estimates obtained for each frame are then averaged to produce the final mask estimate. Note that for CNNs, there is no need for feature concatenation for the input because context information is already included implicitly by 2D filters, but the training targets are also the surrounding 5 frames of IRM connected together. Therefore the output size of DNN and CNNs is $64 \times 5 = 320$.

5.4.2.3 Evaluation and Comparison

Evaluation of the performances of various speech enhancement systems investigated in this chapter are conducted by measuring the quality and intelligibility of processed speech. Although perhaps the most reliable method for assessing these two attributes is subjective listening tests, these tests are costly and time-consuming. Therefore, two objective metrics are employed to predict speech quality and intelligibility of enhanced speech respectively, which are Perceptual Evaluation of Speech Quality (PESQ) [189] and Short-Time Objective Intelligibility score (STOI) [190]. They are chosen because they all have been shown to be highly correlated with subjective measurements obtained from listening tests, and are more accurate than a number of other objective metrics in their respective paper.

PESQ was originally designed for predicting quality of speech passed through narrow-band telecommunication networks [189] and was selected as the ITU-T recommendation P.862 [191]. To accommodate wideband signals, some changes were made to the original narrow-band implementation and this wide-band PESQ is documented in ITU-T recommendation P.862.2 [192]. In this chapter, we use this wide-band version of PESQ, since all speech signals are sampled at 16 kHz. This metric takes in both the clean reference signal and the degraded or processed signal to produce a prediction of subject Mean Opinion Score (MOS) based on their differences. Its computation consists of the following four main steps: (1) pre-processing stage which is responsible for level normalization and time alignment of the two signals; (2) auditory transform that produces perceptual loudness spectra; (3) symmetric and asymmetric disturbance computation between the pair of signals and time and frequency averaging to obtain the raw PESQ score, which ranges from -0.5 to 4.5; (4) conversion of raw PESQ score to MOS (ranges from 1 to 5) using a logistic-type function as proposed in [193]. Higher MOS means better speech quality.

STOI also compares the clean reference signal with the degraded or processed signal. It first decomposes both signals into one-third octave bands by grouping frequency components obtained from STFT. After normalization and clipping of the short-time (384 ms or 30 frames) temporal envelope of the degraded speech, a time-frequency dependent intermediate intelligibility measure, $d_{j,m}$, is obtained for each segment j and each one-third octave band m, as the correlation coefficient between the short-time temporal envelop of the clean speech and that of the degraded speech. The global STOI score is then obtained as the average of this intermediate intelligibility measure over all bands and all frames, and has a monotonic increasing relationship with subjective speech intelligibility. The range of the STOI score is from 0 to 1. It can also be mapped to actual intelligibility score (measured as percentage of words correctly recognised), using a logistic function as proposed by its original authors [190]. The degree of correlation between STOI and subjective intelligibility scores obtained from listening tests conducted by Kjems *et al.* [194] and root-mean-square prediction errors were investigated

and compared with three other objective intelligibility measures in [190]. The main results figure is reproduced in Fig. 5.6. It can be seen that STOI achieves the highest level of correlation with subjective intelligibility scores and the lowest prediction error. The three reference objective measures used for comparison are: (1) DAU: metric based on a sophisticated perceptual model developed by Dau *et al.* [195]; (2) NSEC: metric based on the normalized subband envelope correlation [196]; (3) CSTI: normalized covariance-based Speech Transmission Index (STI)-procedure [197].

Three other speech enhancement or separation algorithms are also employed for comparison with the systems proposed in this chapter. The first one is a supervised learning based method that uses the complementary feature set proposed in [188, 5] and the same DNN as adopted in this chapter to estimate the IRM (referred to as Comp_Feat-DNN). This feature set includes amplitude modulation spectrum, relative spectral processing with PLP, MFCC and 64-channel gammatone filterbank energies, although the original post-processing auto-regressive moving average filter is not used. Another two methods are classical speech enhancement algorithms. One is the MMSE log-spectral amplitude estimator (denoted as Log-MMSE) from [198] and the second one is the audible noise suppression algorithm (denoted as AudNoiseSup) from [199] that aims to suppress only the audible noise spectrum based on psychoacoustics masking principles. Implementations of these two methods are obtained from [146].

5.5 Results

In this section, speech separation results are presented according to the neural network acoustic model used to predict the IRMs, and only average results across several noise types are given in order to save space. Detailed results for each individual noise type are presented in the Appendix A.

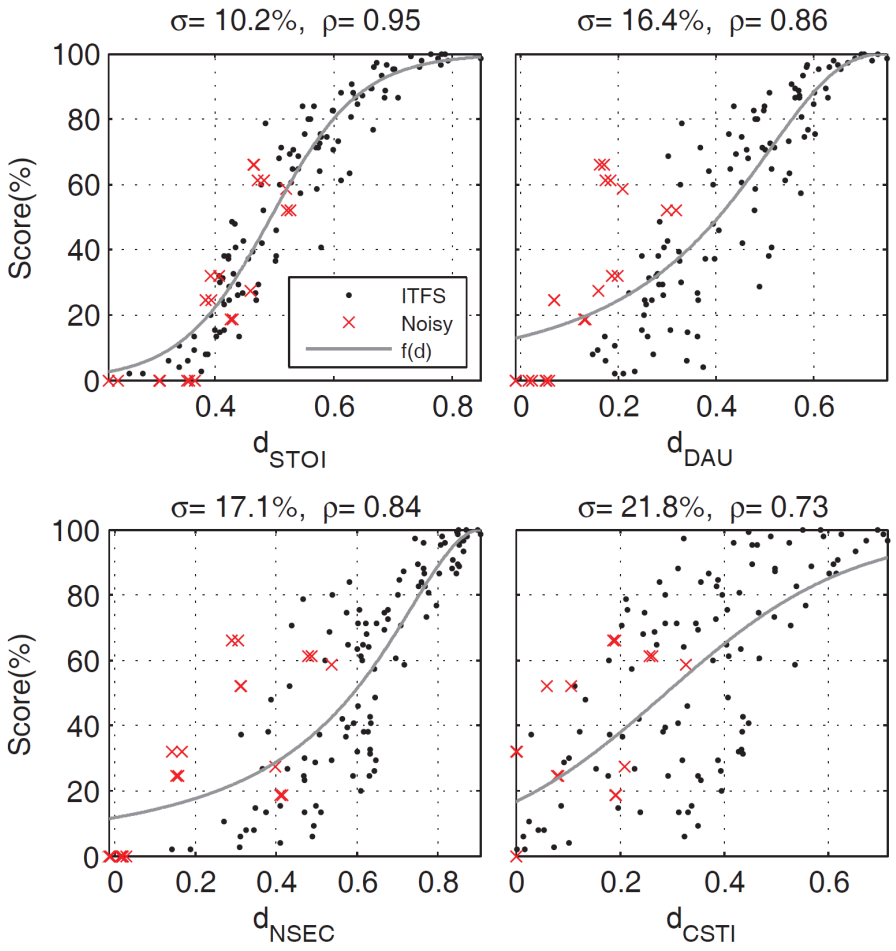


FIGURE 5.6: Comparison of STOI and three other reference objective intelligibility measures in prediction of subjective speech intelligibility scores. The unprocessed noisy speech conditions are denoted by the crosses, and the ITFS-processed conditions are represented by the dots, where ITFS means ideal time frequency segregation, which is just using ideal time-frequency mask for speech separation as introduced in section 5.3. The gray line denotes the mapping function used to convert the objective output to an intelligibility score. Root-mean-square prediction error δ, and the correlation coefficient ρ, between the subjective and objective intelligibility scores are shown in the title of each plot. Adapted from Fig. 1 of [190] with permission.

5.5.1 DNN Results

Fig. 5.7 shows the average PESQ and STOI scores across matched and unmatched noise types for test utterances enhanced by DNN-estimated IRMs. Since in many scenarios, the results from different spectrogram-based features are very similar, a magnified view of the scores around 0 dB SNR is also added in the corner of each sub-plot. It can be seen that all systems significant improve both PESQ and STOI

scores at all SNR levels compared to the original noisy test utterances. However, MRCG features do not show significant advantages compared with other standard auditory filterbanks, which is the opposite of what has been observed in VAD tasks in Chapter 4. In fact, MRCG-16 and MRCG-64 perform the worst in almost all noisy scenarios, with the exception of MRCG-64, which outperforms GTspec slightly in PESQ scores at very low SNR levels (-5 and 0 dB), although its feature size is four times that of GTspec. The performance of GTspec is very similar to that of MRCG-64, while being a little better in many cases. But it is worse than the other three feature types, i.e., Spec, LogMel and FCspec. It is interesting to note that Spec feature performs the best and better than FCspec in terms of PESQ score under matched noise conditions, but is slightly surpassed by FCspec under unmatched noise scenarios. For LogMel, it is the third most effective feature type with DNN acoustic model in most cases. But it should be reminded again that the differences between various features are very small. Fig. 5.8 shows the average PESQ and STOI scores over all noise types and SNR levels for difference features, indicating the same order of effectiveness as discussed above.

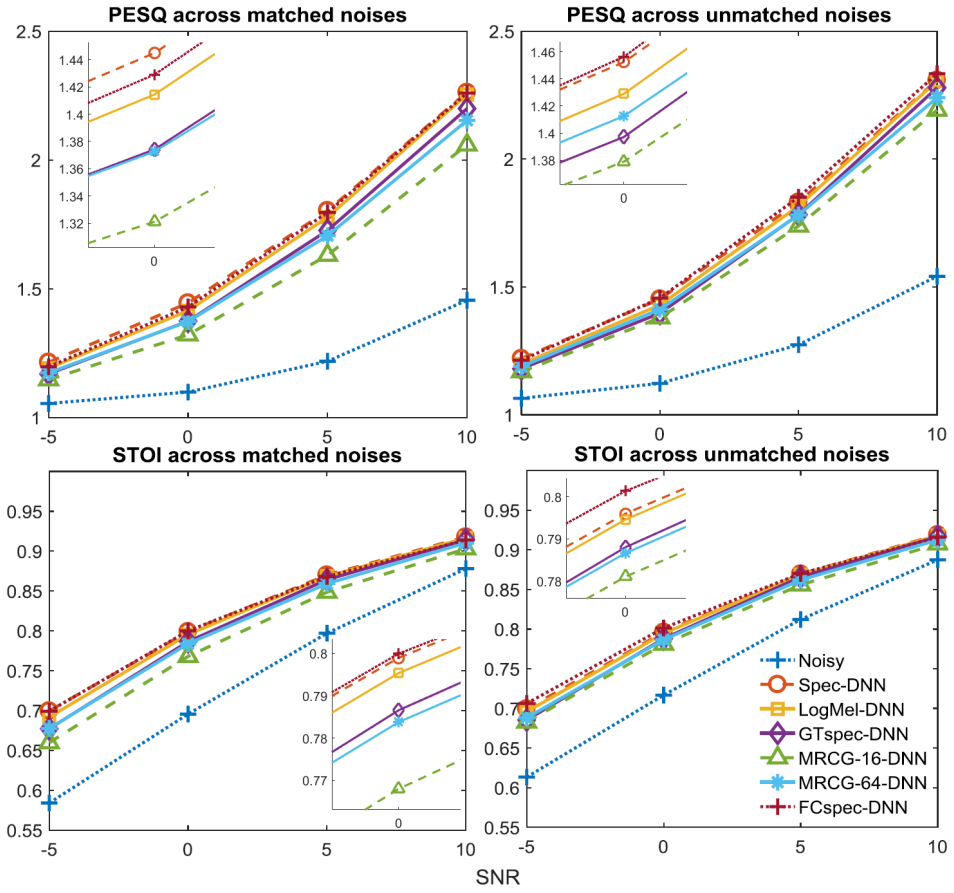


FIGURE 5.7: Average PESQ and STOI scores across matched and unmatched noise types for DNN enhanced test utterances. A magnified view of the scores at 0 dB SNR is shown in the corner of each sub-plot.

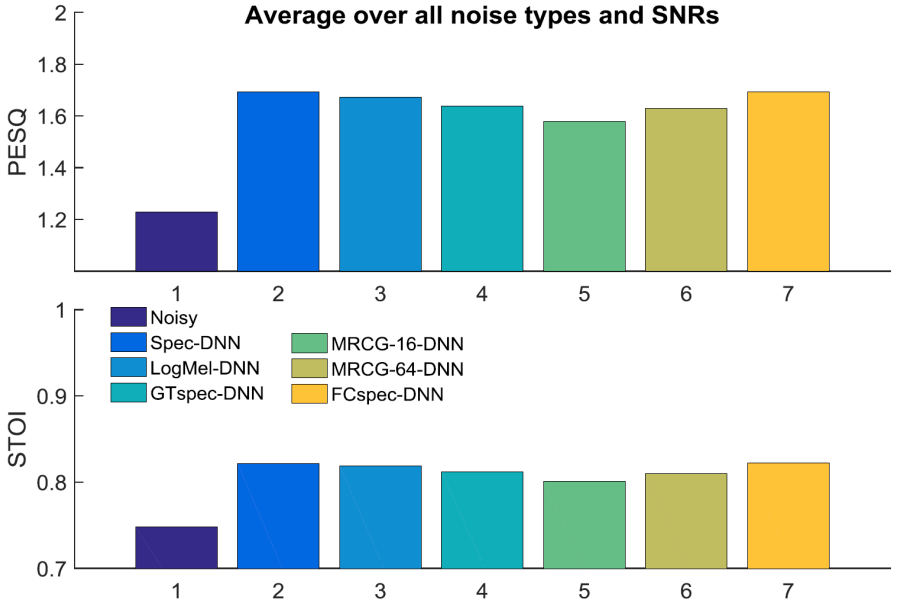


FIGURE 5.8: Average PESQ and STOI scores over all noise types and SNR levels for DNN enhanced test utterances.

5.5.2 CNN Results

The results of PESQ and STOI scores for single resolution CNN and multi-resolution CNN are illustrated in Fig. 5.9, Fig. 5.10 and Fig. 5.11, Fig. 5.12, respectively. Only LogMel, GTspec, FCspec and their multi-window-length extensions are used with CNN structures, while the Spec is not included, in spite of its best performance with DNN backend, because of its much larger feature size. Similar to DNN systems, all CNN based systems produce substantial improvements to both PESQ and STOI scores across all SNR levels, and the performances difference between various feature types are even smaller. FCspec based features become the most effective feature types in almost all noisy test scenarios. And it is interesting to observe that multi-window extensions often lead to lower PESQ and STOI scores compared to their standard filterbank counterpart, contrary to what has been seen in VAD studies presented in Chapter 4. In particular, multi-window-length features only show advantages relative to the standard filterbanks in terms of PESQ under unmatched noise conditions for both CNN-SR and CNN-MR acoustic models. But one exception is MWLFC, it produces the highest scores across most SNR levels with CNN-MR backend. On average, we can see from Fig. 5.10 and Fig. 5.12 that multi-window-length processing is only slightly beneficial for FCspec, but not for LogMel and GTspec.

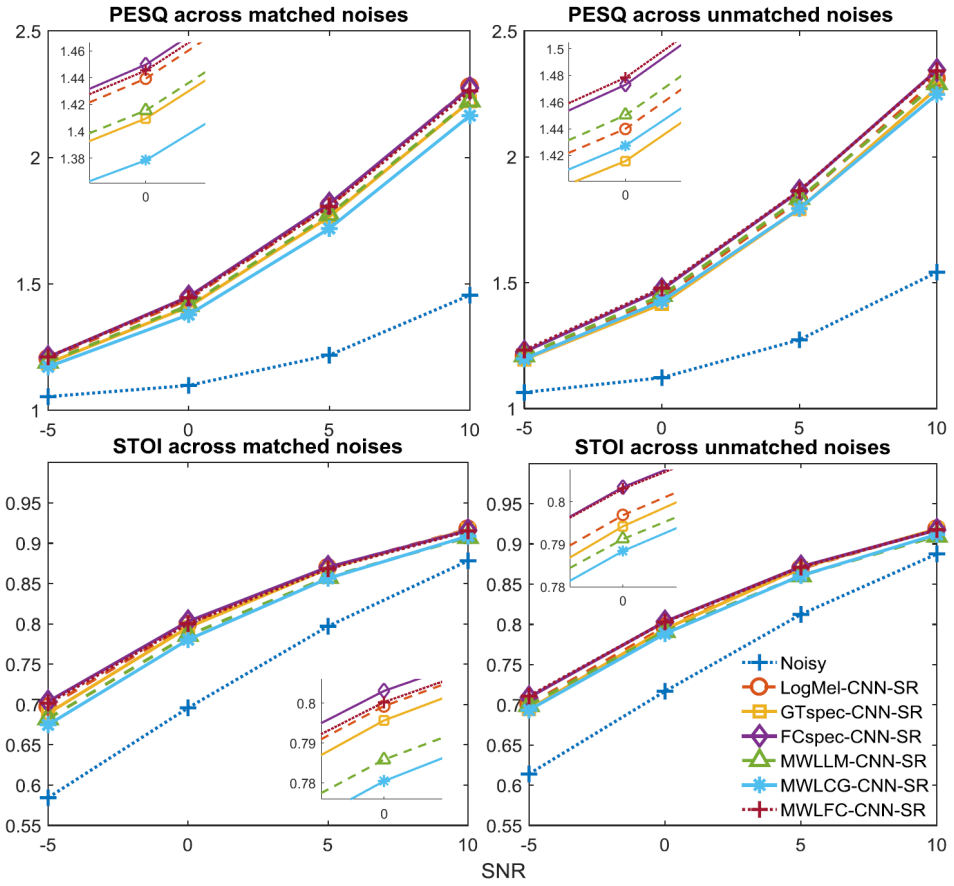


FIGURE 5.9: Average PESQ and STOI scores across matched and unmatched noise types for single resolution CNN enhanced test utterances. A magnified view of the scores at 0 dB SNR is shown in the corner of each sub-plot.

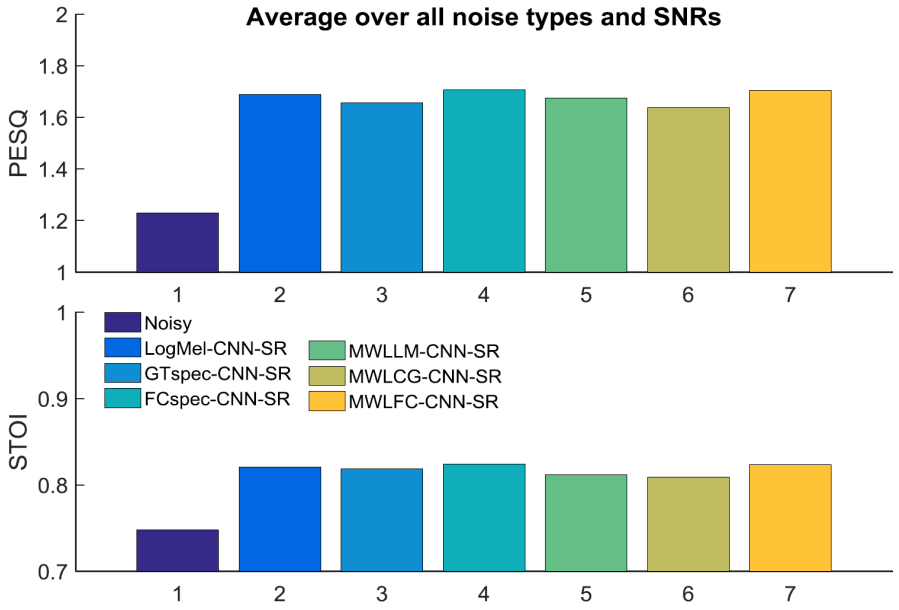


FIGURE 5.10: Average PESQ and STOI scores over all noise types and SNR levels for single resolution CNN enhanced test utterances.

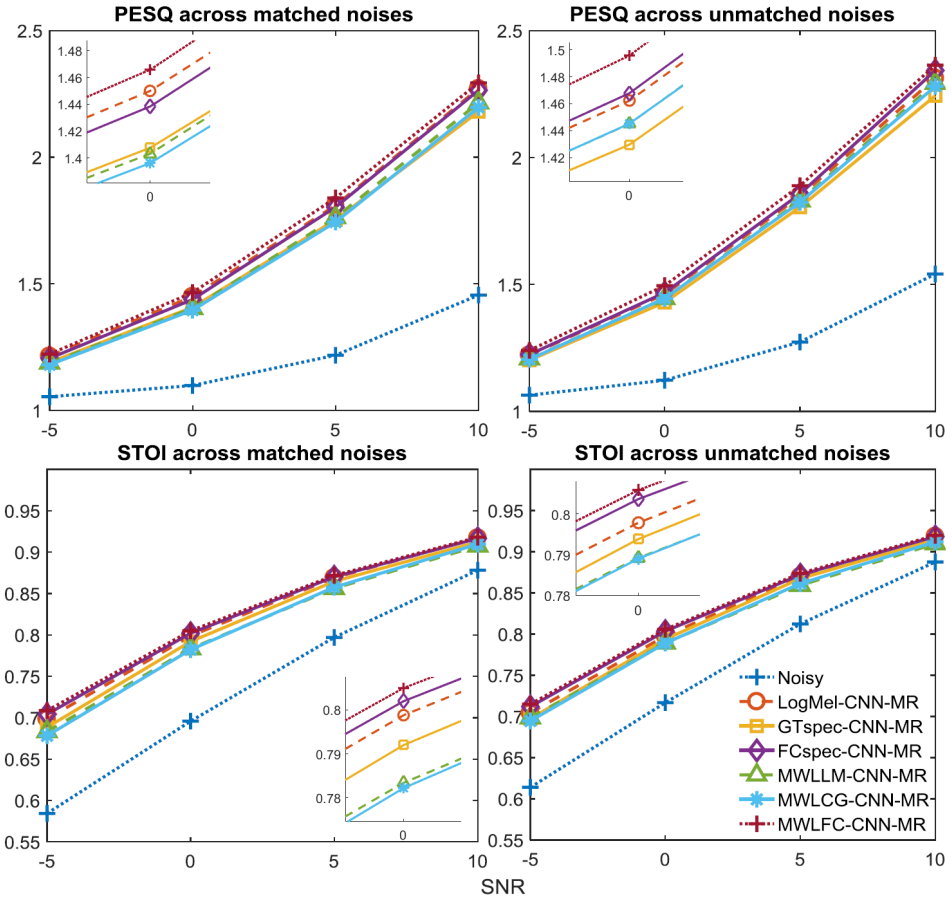


FIGURE 5.11: Average PESQ and STOI scores over matched and unmatched noise types for multi resolution CNN enhanced test utterances. A magnified view of the scores at 0 dB SNR is shown in the corner of each sub-plot.

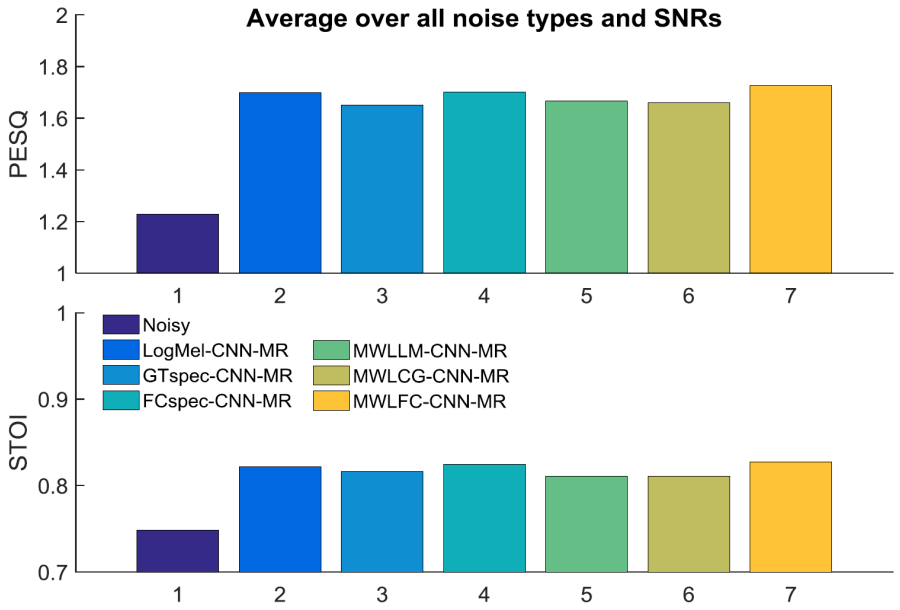


FIGURE 5.12: Average PESQ and STOI scores over all noise types and SNR levels for multi resolution CNN enhanced test utterances.

5.5.3 Comparison

Before comparing with other denoising algorithms, average PESQ and STOI scores over all noise types and SNRs for DNN and CNN acoustic models are compared in Fig. 5.13 to better understand the difference between network structures. It can be seen that CNNs perform slightly better than DNN for all feature types and CNN-MR is better than CNN-SR in most cases. We can also observe more clearly that multi-window length extensions of auditory spectrum can not provide extra improvements in both PESQ and STOI scores, except for FCspec feature.

Fig. 5.14 and Fig. 5.15 shows the comparison of performance between three methods that are proposed in this chapter and three other reference methods. The three proposed methods selected are the best performing system for each auditory filterbank feature as determined from Fig. 5.13. Traditional method like LogMMSE is able to improve speech quality significantly as well, but not speech intelligibility, which is a common drawback of classical approaches. However, all of the proposed supervised learning based methods are capable of increasing both PESQ and STOI scores by large amounts in all noisy test scenarios.

The performance of the complementary feature set with DNN acoustic model, referred to as Comp_Feat-DNN, is quite poor. It can only improve objective speech quality slightly at very low SNR levels (-5 and 0 dB) and could make processed utterances even less intelligible compared to the original noisy ones. The audible noise suppression algorithm also performs poorly, as it is not able to improve either PESQ or STOI in nearly all testing conditions. To better understand their limitations, an example noise utterance (Factory1 noise, mixed at 0 dB SNR) that have been enhanced by proposed and reference methods is shown in Fig. 5.16 and Fig. 5.17, respectively. The IRM and clean spectrogram for this sentence are displayed in the first row of Fig. 5.16, and the estimated IRM and enhanced spectrogram obtained by various methods are shown subsequently. It can be observed that the proposed supervised algorithms are able to estimate the IRM fairly accurately, although the MWLFC-CNN-MR system reduces larger amount of noise

than the LogMel-CNN-MR system, as judged visually from the enhanced spectrogram. For the reference methods, Comp_Feat-DNN fails to recover a substantial area of IRM, introducing a number of gaps within the utterance, which could explain the low scores in PESQ and STOI obtained from processed test sentences. This also demonstrates that the more raw spectrogram based features are more effective than the more sophisticated, hand-engineered, feature extraction methods in neural network based IRM estimation. Although the LogMMSE indeed suppresses a substantial amount of noise, it also distorts the spectrogram structure, by discarding a large portion of mid-to-high frequency speech elements, which is detrimental to speech intelligibility improvement. For AudNoiseSup algorithm, it is strange that it destroys speech information almost completely.

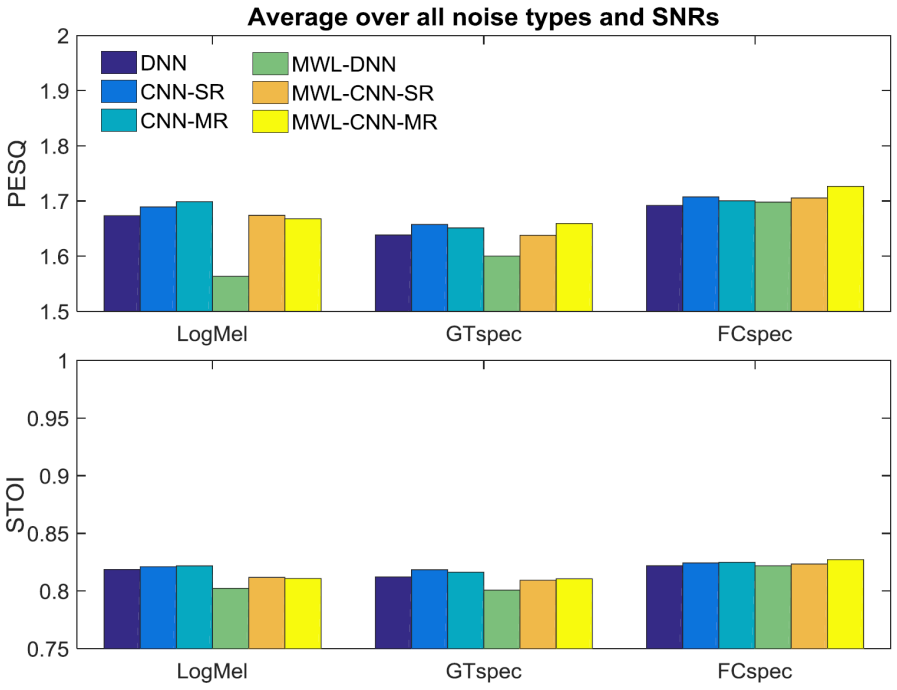


FIGURE 5.13: Average PESQ and STOI scores over all noise types and SNR levels for DNN and CNNs.

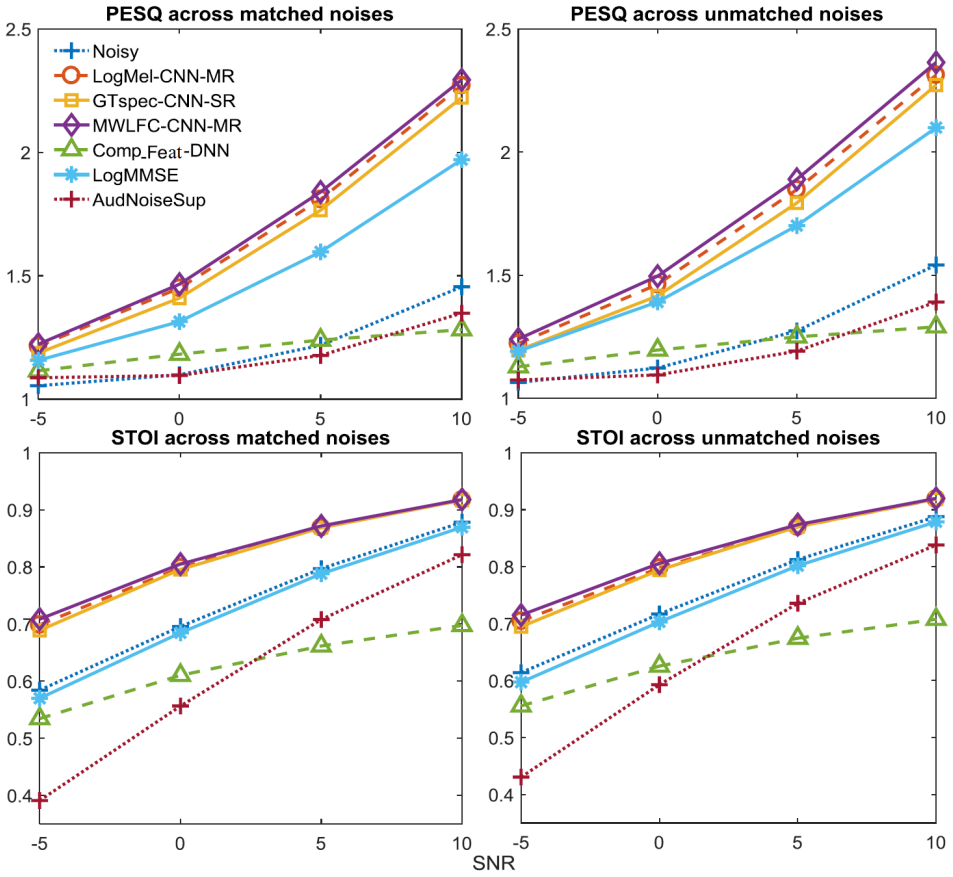


FIGURE 5.14: Comparison of average PESQ and STOI scores over matched and unmatched noise types obtained from three proposed methods and three reference methods.

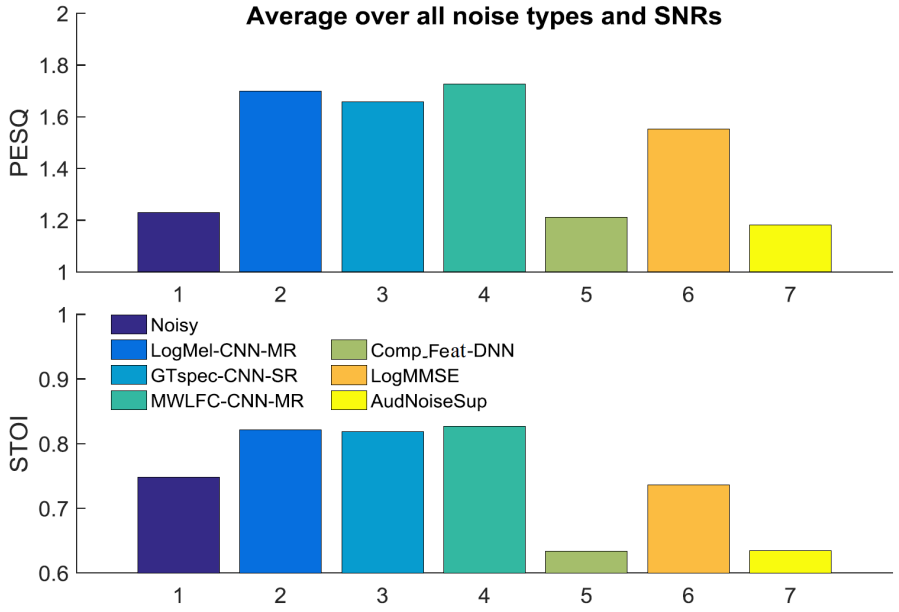


FIGURE 5.15: Comparison of average PESQ and STOI scores over all noise types and SNR levels for three proposed methods and three reference methods.

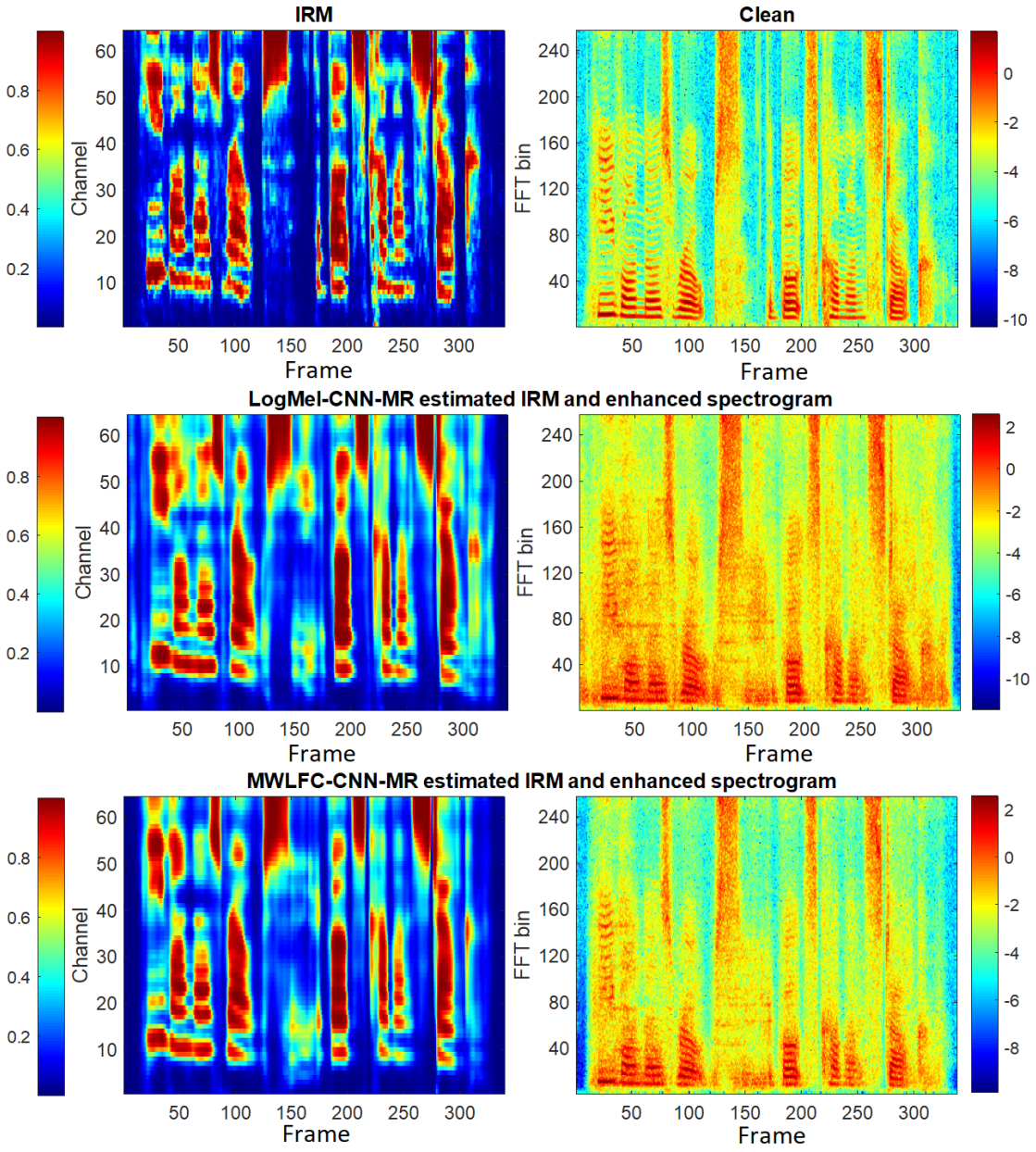


FIGURE 5.16: An example of test noisy utterance and its enhancement by two supervised learning based methods proposed in this chapter. The test utterance is corrupted by factory noise at 0 dB SNR. The IRM of this utterance and its clean spectrogram are shown in the top two plots. The estimated IRM and the corresponding enhanced spectrogram, obtained by LogMel-CNN-MR and MWLFC-CNN-MR, are shown in the middle and bottom two plots, respectively.

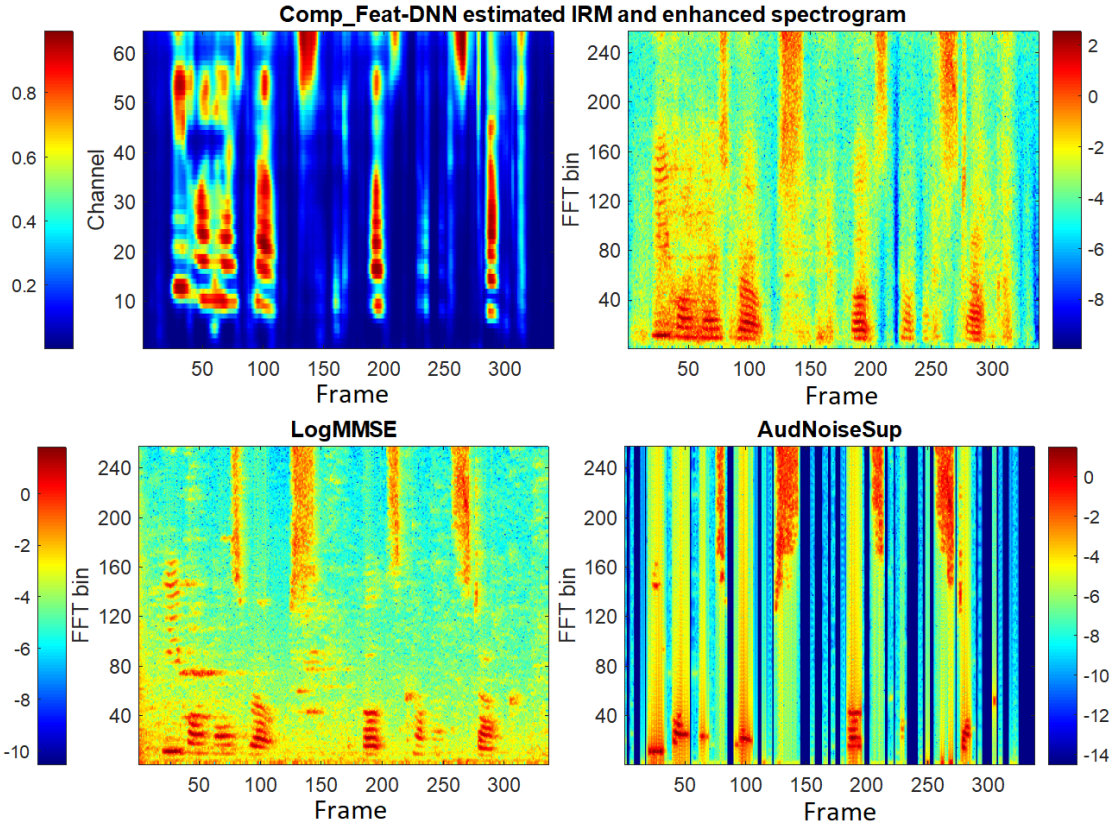


FIGURE 5.17: The same test noisy utterance as used in Fig. 5.16 but enhanced by the three reference methods. The estimated IRM and the corresponding enhanced spectrogram, obtained by Comp_Feat-DNN are shown in the top two plots, while the enhanced spectrograms obtained by LogMMSE and AudNoiseSup are shown in the bottom two plots.

5.6 Summary

In this chapter, the use of filter cascade spectrogram feature for supervised speech separation has been investigated. Both DNN and CNNs have been adopted to estimate the IRMs which are then used to enhance the cochleagrams of noisy test utterances. Similar to what has been observed in VAD studies, CNN models almost always perform better than DNN models, so does multi-resolution CNN to single resolution CNN, even though the amount of performance improvement is marginal. Contrary to the findings presented in Chapter 4, multi-window-length extension of auditory features cannot produce extra gains in performance in many test scenarios, except for the filter cascade feature. Comparison of the systems proposed in this chapter with three reference denoising algorithms has also been performed, illustrating the biggest advantage of the supervised methods proposed with respect to traditional methods is their ability to significantly improve both speech quality and intelligibility under either seen and unseen additive noises.

Chapter 6

Conclusions and Suggestions for Future Work

The mammalian cochlea is a fascinating sensory organ, because of the way its nonlinearity contributes to its remarkable sensitivity and dynamic range. The work presented in this thesis addresses the problem of nonlinear cochlea modelling and its potential application to some aspects of speech processing. In this chapter, important findings of this work are summarized, followed by suggestions for future work.

6.1 Conclusions

6.1.1 Cochlear Modelling

Cochlear modelling is not only an important tool for better understanding of cochlear functions, but also has been inspiring novel algorithms for speech processing. A state space model of the nonlinear cochlea has been proposed by Elliott et al. [1], based on the 1D transmission-line model developed by Neely and Kim [9]. This reformulation allows convenient simulations of the time domain responses of both linear and nonlinear model configurations. However, the computational

burden of this model is extremely high, seriously limiting its applicability to either scientific or practical problems. To improve its computational efficiency, a revised formulation of the original state space model has been developed here [95], that is able to reduce the simulation time by around a factor of 40, by taking advantages of the sparsity pattern of system matrices.

Even with this modified formulation, this type of cochlear model is still too complicated to be applied to practical speech processing problems. Thus an alternative nonlinear filter cascade model of the cochlea, the CARFAC model [98, 12], has also been investigated. This filter cascade model is significantly more efficient to run than the state space model while still being more physiologically plausible compared to other parallel filterbank based models, because it aims to simulate the forward travelling wave inside the cochlea, inspired by the WKB approximate solution to the cochlear wave equation. The analytical relationship between these two models has also been presented.

A systematic comparison of the outputs of these two nonlinear models was then undertaken, in response to single-tones and two-tones stimuli and these are validated against experimental measurements obtained from sensitive cochleae. Results show that both models are reasonably accurate in simulating the representative characteristics of cochlear responses to single-tone stimuli, but in two-tone suppression simulations, the CARFAC model produces results that are relatively closer to physiological measurements than those of the transmission-line model. It is also worth noting that both models cannot account for every aspects of the wide range of datasets. Additionally, tuning of the parameters of the CARFAC model is more flexible than that with the TL model, however, because of the smaller number of free parameters and its simpler stability criterion. Given these benefits of the CARFAC model, it is chosen as the front-end model for performing subsequent speech processing tasks, in comparison with other widely-used methods of front-end processing.

6.1.2 Application to Speech Processing

Two types of speech processing tasks have been investigated using the CARFAC model as a front-end processor: voice activity detection (VAD) and supervised speech separation. For both of these tasks, the CARFAC model serves as a feature extraction module, the output of which is fed to a neural network backend, which is trained to produce the desire final targets using a large pool of training data. In terms of VAD, the target is the probability of each frame of observation being either speech or non-speech, while for speech separation, it is the ideal ratio mask of each time-frequency unit. Since the filter cascade model provides a filterbank-based feature, various other popular parallel filterbank-based feature extraction methods have also been included for performance comparison. These features, although also motivated by auditory principles, provide a less-refined representation of cochlear functions. This offers a good opportunity to investigate whether the incorporation of cochlear nonlinearities can be advantageous over of the use of other simpler auditory filterbanks. Moreover, a number of network structures, including DNN and CNNs, have been investigated for each type of feature.

Voice Activity Detection. For the voice activity detection task, both noise-dependent and noise-independent training of neural networks have been investigated, and the performances on test datasets were evaluated from the Area Under the Receiver Operating Characteristic Curves (AUC). In noise-dependent training scenario, only the DNN was used as the backend and it was found that the nonlinear filter cascade model performed slightly better than other fixed-resolution auditory filterbank-based features in terms of average AUC across all noise types. However, the best overall performance was achieved by the multi-resolution cochleagram feature. Motivated by these results, CNN backends and multi-window-length extensions were also used in noise-independent training scenario, which provides a data-driven way of integrating multiple scales of resolutions and temporal information. Although a similar ranking of performances of different front-ends to that in noise-dependent training case was observed with the DNN backend, the use of CNNs and multi-window-length extensions significantly improved the effectiveness

of all filterbank-based features, making them very similar in terms of overall AUC value averaged over all noise types and SNR levels, as long as a suitable context expansion strategy has been selected, and outperforming the MRCG with DNN backend system. Comparison with another two recent state-of-the-art algorithms has also demonstrated the superiority of the proposed deep network based systems.

Supervised Speech Separation. For the supervised speech separation task, only the multi-condition noise-independent training was investigated, because the practical importance of noise-dependent training is limited. The DNN and CNNs (with some network structure changes), and multi-window-length extension techniques that were investigated in VAD study were all used to estimate the ideal ratio mask, that was then applied to the noisy cochleagram, from which the enhanced speech waveform was synthesized. The performances of different speech separation systems were determined by two objective metrics obtained from the enhanced speech signals: the Short Time Objective Intelligibility score and the Perceptual Evaluation of Speech Quality. In terms of average metric values, all filterbank features perform similarly, although the CARFAC model enjoys a marginal relative advantage. However, contrary to what has been observed in VAD study, CNN based systems are only slightly more effective than DNN based ones, which is probably because temporal context information has been integrated in DNN as well by concatenating features from a surrounding window of five frames. Furthermore, multi-window-length extension of auditory filterbank features cannot produce extra performance gains except for the filter cascade feature. Comparison with three other reference methods has illustrated that the advantage of the proposed supervised methods, with respect to traditional ones, is their capability in improving both intelligibility and quality of processed speech significantly.

Summary. Overall, results from these two speech processing tasks indicate that when no temporal context information is incorporated, the filter cascade front-end can be more effective than other simple auditory filterbanks. However, as long as suitable context expansion strategies have been integrated, all front-end

processors perform rather similarly. Therefore, at least in the neural network based framework considered in this thesis, the inclusion of temporal dynamics of speech signals and network structure engineering seem to be more beneficial than more refined modelling of cochlear nonlinearity in improving the performance of speech processing algorithms.

6.2 Suggestions for Future Work

In the light of the findings of this study, the author recommends that the following areas are worthwhile for further investigations.

- *Refinement of cochlear models.* In Chapter 3, we show that both the TL and CARFAC cochlear models are capable of simulating some of the key features of a normal cochlea. However, they also have some weaknesses in reproducing some other aspects of cochlear mechanics. Thus it would be beneficial to further tune the parameters of these models to make them more compatible with experimental measurements, which could also be valuable to speech processing tasks. Moreover, the evaluation of these models against physiological observations is not comprehensive. For example, the following aspects have not been investigated: the generation of harmonic distortions and combinational tones, and the simulation of the frequency modulation or glide in BM impulse responses. A good example of this kind of study is [3].

- *Inclusion of other acoustic disturbances in speech processing tasks.* For both of the speech processing tasks investigated, only additive background noise is dealt with. Other types of acoustic disturbances such as room reverberation and channel mismatch are also very common and are detrimental to system performance. Therefore, it is worthwhile to explore the potentials of cochlear models under these environments as well. The relative advantages of different front-end feature extractor could be changed in this case.

- *Alternative network structures.* In this work, DNN and several versions of CNN have been integrated with a number of cochlear inspired feature vectors for speech processing, which demonstrates that an important factor for performance improvement is the effective capture of the temporal dynamics of speech signals. However, DNN and CNN can only incorporate a fixed temporal context, which is computationally expensive and may also be sub-optimal in many scenarios. Another type of neural network, called the recurrent neural network, is able to learn a variable length of temporal dependences according to the task at hand, by feeding back network states from previous time steps to the current step. This is an extremely valuable feature for speech processing tasks, especially given the results presented in Chapter 4 and Chapter 5. Popular and successful examples of recurrent neural networks include Long Short-Term Memory [200] and Gated Recurrent Unit networks [201].

Appendix A

Detailed Results of Supervised Speech Separation Experiments

In Chapter 5, only the average metrics across a number of noise types are presented to provide a concise and focused representation of simulation results. In this appendix, detailed results at each noise type and SNR level, obtained from all methods investigated are presented. Table A.1 to A.4 show metric scores of enhanced test utterances corrupted by seen noise types while Table A.5 to A.8 show those corrupted by unseen noise types. The results highlighted in blue are the most effective strategy for each filterbank feature considered in this study. They are selected to compare with another three reference methods under the same testing scenarios, as presented in section 5.5.3. Each value in these tables is obtain by averaging the metric scores computed from 240 testing utterances (noisy or enhanced) created from the TIMIT core test set.

TABLE A.1: Average PESQ and STOI metrics over all noisy and enhanced test utterances obtained from all methods investigated in this work. The five noise types used to corrupt clean test utterances are those adopted during training and the SNR is -5 dB in this testing condition. Oproom: Ship operation room noise; Engine: Ship engine room noise.

Feature type	**Café**		**Car**		**Factory1**		**Oproom**		**Engine**		**Average**	
	PESQ	STOI	PESQ	STOI	PESQ	STOI	PESQ	STOI	PESQ	STOI	PESQ	STOI
Mixture	1.053	0.526	1.059	0.701	1.042	0.533	1.038	0.593	1.081	0.567	1.055	0.584
Spec-DNN	1.122	0.598	1.390	0.802	1.132	0.652	1.148	0.718	1.281	0.728	1.215	0.700
LogMel-DNN	1.110	0.595	1.391	0.803	1.122	0.651	1.116	0.708	1.216	0.702	1.191	0.692
GTspec-DNN	1.101	0.587	1.373	0.799	1.107	0.637	1.097	0.695	1.169	0.668	1.169	0.677
MRCG-16-DNN	1.096	0.579	1.309	0.782	1.088	0.609	1.083	0.678	1.158	0.651	1.147	0.660
MRCG-64-DNN	1.105	0.589	1.362	0.793	1.102	0.627	1.102	0.689	1.197	0.690	1.174	0.678
FCspec-DNN	1.126	0.618	1.404	0.806	1.130	0.665	1.120	0.720	1.205	0.687	1.197	0.699
MWLLM-DNN	1.096	0.591	1.298	0.789	1.086	0.620	1.078	0.682	1.158	0.645	1.143	0.665
MWLCG-DNN	1.100	0.578	1.317	0.782	1.094	0.614	1.086	0.680	1.162	0.654	1.152	0.662
MWLFC-DNN	1.130	0.616	1.417	0.807	1.137	0.668	1.116	0.719	1.223	0.697	1.205	0.701
LogMel-CNN-SR	1.126	0.606	1.397	0.806	1.140	0.663	1.127	0.708	1.231	0.703	1.204	0.697
GTspec-CNN-SR	1.111	0.602	1.393	0.805	1.120	0.654	1.108	0.703	1.202	0.681	1.187	0.689
FCspec-CNN-SR	1.132	0.622	1.417	0.808	1.145	0.671	1.131	0.720	1.228	0.697	1.211	0.704
MWLLM-CNN-SR	1.127	0.608	1.364	0.796	1.134	0.650	1.127	0.700	1.203	0.658	1.191	0.682
MWLCG-CNN-SR	1.116	0.596	1.336	0.787	1.123	0.638	1.097	0.685	1.190	0.668	1.172	0.675
MWLFC-CNN-SR	1.134	0.621	1.415	0.806	1.144	0.668	1.131	0.716	1.234	0.697	1.212	0.702
LogMel-CNN-MR	1.130	0.608	1.404	0.803	1.146	0.659	1.145	0.714	1.248	0.709	1.215	0.699
GTspec-CNN-MR	1.113	0.603	1.403	0.802	1.124	0.646	1.105	0.697	1.205	0.694	1.190	0.688
FCspec-CNN-MR	1.131	0.624	1.405	0.808	1.140	0.670	1.124	0.716	1.230	0.703	1.206	0.704
MWLLM-CNN-MR	1.122	0.610	1.366	0.794	1.126	0.645	1.121	0.695	1.203	0.672	1.188	0.683
MWLCG-CNN-MR	1.120	0.599	1.350	0.788	1.128	0.639	1.109	0.692	1.197	0.673	1.181	0.678
MWLFC-CNN-MR	1.138	0.626	1.433	0.809	1.155	0.674	1.137	0.726	1.251	0.708	1.223	0.708
Comb-DNN	1.097	0.486	1.164	0.605	1.088	0.497	1.091	0.546	1.135	0.538	1.115	0.534
LogMMSE	1.095	0.485	1.261	0.686	1.089	0.506	1.138	0.582	1.204	0.591	1.157	0.570
AudNoiseSup	1.118	0.284	1.087	0.567	1.068	0.307	1.072	0.424	1.089	0.373	1.087	0.391

TABLE A.2: Average PESQ and STOI metrics over all noisy and enhanced test utterances obtained from all methods investigated in this work. The five noise types used to corrupt clean test utterances are those adopted during training and the SNR is 0 dB in this testing condition. Oproom: Ship operation room noise; Engine: Ship engine room noise.

Feature type	**Café**		**Car**		**Factory1**		**Oproom**		**Engine**		**Average**	
	PESQ	STOI	PESQ	STOI	PESQ	STOI	PESQ	STOI	PESQ	STOI	PESQ	STOI
Mixture	1.093	0.652	1.127	0.790	1.072	0.652	1.070	0.695	1.132	0.689	1.099	0.696
Spec-DNN	1.287	0.737	1.726	0.864	1.316	0.768	1.371	0.808	1.524	0.818	1.445	0.799
LogMel-DNN	1.270	0.736	1.736	0.865	1.302	0.770	1.313	0.802	1.451	0.804	1.414	0.795
GTspec-DNN	1.249	0.728	1.720	0.863	1.274	0.763	1.273	0.795	1.355	0.784	1.374	0.787
MRCG-16-DNN	1.238	0.718	1.605	0.848	1.210	0.732	1.227	0.775	1.326	0.767	1.321	0.768
MRCG-64-DNN	1.259	0.729	1.676	0.856	1.242	0.751	1.272	0.787	1.414	0.796	1.373	0.784
FCspec-DNN	1.324	0.754	1.739	0.862	1.333	0.781	1.300	0.805	1.449	0.798	1.429	0.800
MWLLM-DNN	1.233	0.725	1.585	0.851	1.206	0.742	1.209	0.780	1.312	0.760	1.309	0.772
MWLCG-DNN	1.245	0.718	1.620	0.847	1.229	0.735	1.244	0.777	1.329	0.766	1.333	0.769
MWLFC-DNN	1.333	0.754	1.748	0.863	1.344	0.781	1.295	0.804	1.477	0.801	1.439	0.801
LogMel-CNN-SR	1.306	0.745	1.730	0.866	1.344	0.776	1.322	0.799	1.495	0.810	1.439	0.799
GTspec-CNN-SR	1.284	0.743	1.734	0.866	1.308	0.774	1.285	0.798	1.437	0.797	1.410	0.796
FCspec-CNN-SR	1.337	0.757	1.742	0.864	1.359	0.784	1.331	0.807	1.480	0.803	1.450	0.803
MWLLM-CNN-SR	1.318	0.741	1.665	0.855	1.347	0.767	1.330	0.792	1.418	0.774	1.416	0.786
MWLCG-CNN-SR	1.291	0.733	1.632	0.851	1.308	0.756	1.267	0.782	1.394	0.780	1.378	0.780
MWLFC-CNN-SR	1.341	0.756	1.735	0.862	1.355	0.780	1.323	0.803	1.473	0.800	1.445	0.800
LogMel-CNN-MR	1.313	0.745	1.734	0.864	1.345	0.773	1.348	0.800	1.510	0.812	1.450	0.799
GTspec-CNN-MR	1.287	0.741	1.726	0.862	1.303	0.766	1.275	0.791	1.447	0.800	1.408	0.792
FCspec-CNN-MR	1.330	0.757	1.723	0.863	1.346	0.781	1.311	0.803	1.482	0.806	1.438	0.802
MWLLM-CNN-MR	1.303	0.739	1.665	0.852	1.318	0.759	1.313	0.786	1.417	0.781	1.403	0.783
MWLCG-CNN-MR	1.299	0.735	1.651	0.851	1.319	0.756	1.299	0.786	1.413	0.783	1.396	0.782
MWLFC-CNN-MR	1.354	0.761	1.751	0.864	1.370	0.784	1.344	0.808	1.510	0.808	1.466	0.805
Comb-DNN	1.171	0.587	1.221	0.661	1.160	0.584	1.156	0.615	1.203	0.604	1.182	0.610
LogMMSE	1.095	0.485	1.261	0.686	1.089	0.506	1.138	0.582	1.204	0.591	1.157	0.570
AudNoiseSup	1.118	0.284	1.087	0.567	1.068	0.307	1.072	0.424	1.089	0.373	1.087	0.391

TABLE A.3: Average PESQ and STOI metrics over all noisy and enhanced test utterances obtained from all methods investigated in this work. The five noise types used to corrupt clean test utterances are those adopted during training and the SNR is 5 dB in this testing condition. Oproom: Ship operation room noise; Engine: Ship engine room noise.

Feature type	**Café**		**Car**		**Factory1**		**Oproom**		**Engine**		**Average**	
	PESQ	STOI	PESQ	STOI	PESQ	STOI	PESQ	STOI	PESQ	STOI	PESQ	STOI
Mixture	1.204	0.768	1.294	0.856	1.171	0.771	1.171	0.787	1.252	0.803	1.218	0.797
Spec-DNN	1.607	0.838	2.114	0.905	1.691	0.856	1.736	0.866	1.870	0.885	1.804	0.870
LogMel-DNN	1.595	0.837	2.128	0.905	1.687	0.858	1.664	0.863	1.816	0.879	1.778	0.868
GTspec-DNN	1.560	0.834	2.113	0.904	1.649	0.855	1.611	0.860	1.699	0.868	1.726	0.864
MRCG-16-DNN	1.529	0.821	1.973	0.892	1.511	0.832	1.517	0.844	1.621	0.854	1.630	0.849
MRCG-64-DNN	1.568	0.831	2.046	0.897	1.564	0.842	1.587	0.852	1.765	0.874	1.706	0.859
FCspec-DNN	1.691	0.847	2.089	0.898	1.764	0.860	1.618	0.861	1.820	0.875	1.796	0.868
MWLLM-DNN	1.511	0.824	1.980	0.893	1.497	0.835	1.478	0.845	1.606	0.851	1.614	0.850
MWLCG-DNN	1.551	0.820	2.000	0.891	1.557	0.831	1.555	0.845	1.626	0.853	1.658	0.848
MWLFC-DNN	1.702	0.846	2.098	0.899	1.769	0.860	1.627	0.861	1.850	0.877	1.809	0.869
LogMel-CNN-SR	1.637	0.841	2.106	0.905	1.737	0.858	1.657	0.860	1.894	0.884	1.806	0.870
GTspec-CNN-SR	1.608	0.841	2.100	0.905	1.691	0.859	1.617	0.861	1.811	0.877	1.765	0.869
FCspec-CNN-SR	1.703	0.848	2.086	0.900	1.778	0.862	1.671	0.863	1.848	0.879	1.817	0.870
MWLLM-CNN-SR	1.657	0.835	2.040	0.893	1.738	0.848	1.657	0.851	1.779	0.862	1.774	0.858
MWLCG-CNN-SR	1.624	0.831	1.975	0.894	1.687	0.846	1.582	0.847	1.728	0.864	1.719	0.856
MWLFC-CNN-SR	1.699	0.846	2.076	0.899	1.763	0.859	1.662	0.860	1.831	0.877	1.806	0.868
LogMel-CNN-MR	1.646	0.840	2.107	0.904	1.723	0.857	1.677	0.860	1.895	0.884	1.810	0.869
GTspec-CNN-MR	1.610	0.839	2.078	0.901	1.664	0.852	1.596	0.854	1.817	0.877	1.753	0.865
FCspec-CNN-MR	1.691	0.848	2.082	0.901	1.746	0.860	1.642	0.861	1.854	0.881	1.803	0.870
MWLLM-CNN-MR	1.638	0.832	2.063	0.893	1.695	0.845	1.637	0.847	1.776	0.864	1.762	0.856
MWLCG-CNN-MR	1.640	0.832	1.997	0.895	1.702	0.846	1.627	0.849	1.754	0.866	1.744	0.858
MWLFC-CNN-MR	1.729	0.850	2.097	0.902	1.787	0.862	1.701	0.864	1.882	0.881	1.839	0.872
Comb-DNN	1.251	0.657	1.252	0.695	1.225	0.647	1.216	0.662	1.253	0.647	1.239	0.662
LogMMSE	1.443	0.744	1.788	0.847	1.441	0.753	1.630	0.781	1.681	0.815	1.597	0.788
AudNoiseSup	1.098	0.640	1.390	0.817	1.111	0.666	1.159	0.726	1.128	0.691	1.177	0.708

TABLE A.4: Average PESQ and STOI metrics over all noisy and enhanced test utterances obtained from all methods investigated in this work. The five noise types used to corrupt clean test utterances are those adopted during training and the SNR is 10 dB in this testing condition. Oproom: Ship operation room noise; Engine: Ship engine room noise.

Feature type	**Café**		**Car**		**Factory1**		**Oproom**		**Engine**		**Average**	
	PESQ	STOI	PESQ	STOI	PESQ	STOI	PESQ	STOI	PESQ	STOI	PESQ	STOI
Mixture	1.448	0.866	1.578	0.906	1.372	0.864	1.393	0.864	1.485	0.890	1.455	0.878
Spec-DNN	2.074	0.905	2.591	0.934	2.166	0.911	2.204	0.910	2.277	0.929	2.262	0.918
LogMel-DNN	2.070	0.904	2.617	0.933	2.182	0.911	2.143	0.908	2.257	0.926	2.254	0.916
GTspec-DNN	2.036	0.903	2.573	0.932	2.144	0.909	2.094	0.906	2.151	0.921	2.200	0.914
MRCG-16-DNN	1.970	0.892	2.432	0.923	1.922	0.894	1.968	0.895	2.001	0.911	2.059	0.903
MRCG-64-DNN	2.023	0.899	2.507	0.926	2.010	0.902	2.057	0.901	2.171	0.923	2.154	0.910
FCspec-DNN	2.192	0.907	2.472	0.924	2.269	0.910	2.097	0.905	2.268	0.924	2.260	0.914
MWLLM-DNN	1.941	0.894	2.459	0.924	1.909	0.896	1.909	0.895	1.995	0.912	2.043	0.904
MWLCG-DNN	2.009	0.891	2.460	0.921	1.997	0.893	2.024	0.895	2.020	0.910	2.102	0.902
MWLFC-DNN	2.200	0.907	2.469	0.924	2.279	0.910	2.121	0.905	2.289	0.925	2.272	0.914
LogMel-CNN-SR	2.110	0.906	2.589	0.934	2.210	0.911	2.129	0.906	2.354	0.930	2.278	0.917
GTspec-CNN-SR	2.083	0.907	2.518	0.932	2.161	0.911	2.085	0.907	2.263	0.927	2.222	0.917
FCspec-CNN-SR	2.194	0.909	2.480	0.926	2.273	0.911	2.138	0.906	2.287	0.927	2.274	0.916
MWLLM-CNN-SR	2.121	0.899	2.502	0.922	2.185	0.901	2.107	0.897	2.204	0.918	2.224	0.907
MWLCG-CNN-SR	2.077	0.899	2.411	0.926	2.145	0.902	2.050	0.898	2.144	0.919	2.165	0.909
MWLFC-CNN-SR	2.188	0.908	2.480	0.927	2.250	0.910	2.136	0.905	2.262	0.926	2.263	0.915
LogMel-CNN-MR	2.103	0.906	2.599	0.933	2.178	0.910	2.142	0.907	2.332	0.930	2.271	0.917
GTspec-CNN-MR	2.060	0.903	2.460	0.928	2.101	0.905	2.039	0.901	2.240	0.925	2.180	0.912
FCspec-CNN-MR	2.179	0.909	2.487	0.928	2.236	0.911	2.112	0.906	2.307	0.929	2.264	0.917
MWLLM-CNN-MR	2.105	0.898	2.518	0.924	2.155	0.900	2.090	0.896	2.206	0.919	2.215	0.907
MWLCG-CNN-MR	2.105	0.900	2.442	0.927	2.164	0.902	2.102	0.899	2.171	0.920	2.197	0.910
MWLFC-CNN-MR	2.210	0.910	2.494	0.930	2.271	0.913	2.179	0.908	2.315	0.929	2.294	0.918
Comb-DNN	1.304	0.707	1.297	0.720	1.270	0.686	1.261	0.698	1.276	0.677	1.282	0.698
LogMMSE	1.790	0.847	2.171	0.903	1.774	0.847	2.049	0.858	2.068	0.893	1.970	0.870
AudNoiseSup	1.208	0.786	1.707	0.889	1.225	0.790	1.320	0.816	1.282	0.826	1.348	0.821

TABLE A.5: Average PESQ and STOI metrics over all noisy and enhanced test utterances obtained from all methods investigated in this work. The five noise types used to corrupt clean test utterances are not seen during training and the SNR is -5 dB in this testing condition.

Feature type	**Babble**		**Factory2**		**Tank**		**F16**		**Train**		**Average**	
	PESQ	STOI	PESQ	STOI	PESQ	STOI	PESQ	STOI	PESQ	STOI	PESQ	STOI
Mixture	1.072	0.544	1.066	0.651	1.050	0.679	1.046	0.556	1.087	0.637	1.064	0.613
Spec-DNN	1.142	0.599	1.304	0.760	1.299	0.772	1.162	0.671	1.192	0.695	1.220	0.699
LogMel-DNN	1.135	0.596	1.282	0.758	1.258	0.770	1.123	0.660	1.194	0.695	1.198	0.696
GTspec-DNN	1.126	0.589	1.256	0.750	1.224	0.758	1.102	0.636	1.191	0.694	1.180	0.685
MRCG-16-DNN	1.125	0.588	1.243	0.742	1.201	0.756	1.108	0.639	1.164	0.689	1.168	0.683
MRCG-64-DNN	1.131	0.594	1.277	0.749	1.240	0.762	1.118	0.645	1.177	0.691	1.189	0.688
FCspec-DNN	1.147	0.608	1.324	0.770	1.264	0.781	1.128	0.666	1.203	0.706	1.213	0.706
MWLLM-DNN	1.125	0.592	1.233	0.745	1.190	0.762	1.104	0.633	1.160	0.687	1.162	0.684
MWLCG-DNN	1.125	0.587	1.249	0.745	1.210	0.757	1.114	0.642	1.169	0.690	1.173	0.684
MWLFC-DNN	1.151	0.609	1.336	0.771	1.279	0.781	1.140	0.669	1.208	0.711	1.223	0.708
LogMel-CNN-SR	1.137	0.600	1.302	0.764	1.265	0.769	1.133	0.665	1.205	0.704	1.208	0.700
GTspec-CNN-SR	1.131	0.599	1.291	0.762	1.240	0.767	1.109	0.642	1.202	0.703	1.195	0.695
FCspec-CNN-SR	1.149	0.609	1.338	0.773	1.284	0.782	1.141	0.672	1.215	0.710	1.225	0.709
MWLLM-CNN-SR	1.141	0.602	1.308	0.761	1.266	0.773	1.139	0.655	1.197	0.705	1.210	0.699
MWLCG-CNN-SR	1.131	0.597	1.288	0.756	1.246	0.764	1.142	0.658	1.181	0.692	1.198	0.693
MWLFC-CNN-SR	1.152	0.613	1.339	0.772	1.301	0.783	1.151	0.675	1.217	0.711	1.232	0.711
LogMel-CNN-MR	1.144	0.601	1.315	0.763	1.284	0.773	1.152	0.674	1.217	0.711	1.222	0.704
GTspec-CNN-MR	1.134	0.598	1.302	0.762	1.244	0.766	1.117	0.655	1.212	0.705	1.202	0.697
FCspec-CNN-MR	1.146	0.611	1.333	0.774	1.281	0.783	1.143	0.679	1.217	0.713	1.224	0.712
MWLLM-CNN-MR	1.140	0.600	1.301	0.759	1.254	0.767	1.140	0.665	1.200	0.701	1.207	0.698
MWLCG-CNN-MR	1.134	0.596	1.297	0.758	1.257	0.767	1.149	0.664	1.190	0.688	1.205	0.695
MWLFC-CNN-MR	1.155	0.615	1.359	0.777	1.311	0.787	1.155	0.683	1.222	0.713	1.240	0.715
Comb-DNN	1.101	0.475	1.150	0.565	1.154	0.604	1.104	0.543	1.131	0.591	1.128	0.556
LogMMSE	1.108	0.494	1.274	0.641	1.264	0.670	1.154	0.575	1.155	0.606	1.191	0.597
AudNoiseSup	1.066	0.337	1.069	0.451	1.072	0.529	1.103	0.331	1.061	0.505	1.074	0.431

TABLE A.6: Average PESQ and STOI metrics over all noisy and enhanced test utterances obtained from all methods investigated in this work. The five noise types used to corrupt clean test utterances are not seen during training and the SNR is 0 dB in this testing condition.

Feature type	**Babble**		**Factory2**		**Tank**		**F16**		**Train**		**Average**	
	PESQ	STOI	PESQ	STOI	PESQ	STOI	PESQ	STOI	PESQ	STOI	PESQ	STOI
Mixture	1.113	0.664	1.145	0.752	1.118	0.769	1.081	0.678	1.155	0.717	1.123	0.717
Spec-DNN	1.307	0.731	1.608	0.837	1.596	0.843	1.377	0.786	1.374	0.783	1.452	0.796
LogMel-DNN	1.298	0.731	1.599	0.838	1.559	0.842	1.315	0.780	1.374	0.782	1.429	0.795
GTspec-DNN	1.277	0.724	1.559	0.834	1.509	0.836	1.264	0.764	1.377	0.782	1.397	0.788
MRCG-16-DNN	1.287	0.720	1.524	0.821	1.463	0.828	1.281	0.764	1.340	0.772	1.379	0.781
MRCG-64-DNN	1.300	0.728	1.576	0.828	1.523	0.835	1.302	0.766	1.362	0.776	1.413	0.787
FCspec-DNN	1.341	0.744	1.640	0.841	1.568	0.848	1.323	0.783	1.410	0.791	1.456	0.801
MWLLM-DNN	1.284	0.725	1.522	0.825	1.440	0.832	1.250	0.752	1.326	0.770	1.364	0.781
MWLCG-DNN	1.288	0.720	1.540	0.821	1.486	0.828	1.300	0.767	1.349	0.770	1.393	0.781
MWLFC-DNN	1.345	0.746	1.652	0.842	1.594	0.847	1.346	0.783	1.404	0.789	1.468	0.801
LogMel-CNN-SR	1.306	0.735	1.615	0.839	1.558	0.841	1.329	0.781	1.392	0.788	1.440	0.797
GTspec-CNN-SR	1.294	0.733	1.593	0.839	1.523	0.840	1.279	0.771	1.390	0.788	1.416	0.794
FCspec-CNN-SR	1.347	0.746	1.652	0.842	1.593	0.848	1.342	0.787	1.431	0.794	1.473	0.803
MWLLM-CNN-SR	1.326	0.734	1.628	0.832	1.553	0.838	1.350	0.772	1.394	0.780	1.450	0.791
MWLCG-CNN-SR	1.303	0.727	1.580	0.829	1.527	0.833	1.365	0.780	1.362	0.773	1.427	0.788
MWLFC-CNN-SR	1.352	0.747	1.648	0.841	1.612	0.848	1.360	0.787	1.420	0.792	1.478	0.803
LogMel-CNN-MR	1.319	0.735	1.635	0.837	1.577	0.842	1.369	0.786	1.411	0.789	1.462	0.798
GTspec-CNN-MR	1.306	0.734	1.599	0.835	1.527	0.837	1.302	0.775	1.414	0.788	1.430	0.794
FCspec-CNN-MR	1.340	0.745	1.640	0.842	1.585	0.848	1.347	0.789	1.427	0.794	1.468	0.804
MWLLM-CNN-MR	1.320	0.730	1.622	0.829	1.540	0.834	1.347	0.774	1.396	0.779	1.445	0.789
MWLCG-CNN-MR	1.310	0.727	1.600	0.829	1.547	0.834	1.385	0.783	1.384	0.772	1.445	0.789
MWLFC-CNN-MR	1.359	0.750	1.676	0.844	1.631	0.850	1.379	0.791	1.434	0.794	1.496	0.806
Comb-DNN	1.175	0.584	1.213	0.630	1.211	0.656	1.180	0.608	1.200	0.649	1.196	0.625
LogMMSE	1.244	0.629	1.529	0.738	1.507	0.759	1.359	0.694	1.316	0.695	1.391	0.703
AudNoiseSup	1.074	0.530	1.105	0.635	1.113	0.681	1.076	0.498	1.105	0.620	1.095	0.593

TABLE A.7: Average PESQ and STOI metrics over all noisy and enhanced test utterances obtained from all methods investigated in this work. The five noise types used to corrupt clean test utterances are not seen during training and the SNR is 5 dB in this testing condition.

Feature type	Babble		Factory2		Tank		F16		Train		Average	
	PESQ	STOI	PESQ	STOI	PESQ	STOI	PESQ	STOI	PESQ	STOI	PESQ	STOI
Mixture	1.248	0.774	1.329	0.840	1.297	0.850	1.181	0.791	1.232	0.763	1.273	0.812
Spec-DNN	1.653	0.835	2.074	0.895	2.023	0.897	1.721	0.865	1.686	0.855	1.831	0.869
LogMel-DNN	1.645	0.836	2.079	0.894	2.002	0.896	1.674	0.863	1.681	0.854	1.816	0.869
GTspec-DNN	1.613	0.833	2.053	0.893	1.942	0.894	1.613	0.855	1.691	0.854	1.782	0.866
MRCG-16-DNN	1.613	0.822	1.952	0.881	1.875	0.885	1.604	0.851	1.647	0.840	1.738	0.856
MRCG-64-DNN	1.645	0.831	2.009	0.887	1.942	0.891	1.638	0.855	1.671	0.846	1.781	0.862
FCspec-DNN	1.725	0.841	2.113	0.893	1.982	0.897	1.689	0.863	1.744	0.857	1.851	0.870
MWLLM-DNN	1.599	0.824	1.957	0.882	1.865	0.886	1.534	0.842	1.610	0.841	1.713	0.855
MWLCG-DNN	1.621	0.822	1.981	0.880	1.925	0.884	1.643	0.852	1.657	0.836	1.765	0.855
MWLFC-DNN	1.732	0.843	2.113	0.894	2.009	0.897	1.718	0.862	1.573	0.824	1.829	0.864
LogMel-CNN-SR	1.656	0.837	2.083	0.894	1.983	0.896	1.700	0.863	1.708	0.856	1.826	0.869
GTspec-CNN-SR	1.645	0.839	2.046	0.895	1.923	0.896	1.639	0.862	1.713	0.858	1.793	0.870
FCspec-CNN-SR	1.723	0.843	2.119	0.894	1.999	0.897	1.713	0.865	1.765	0.860	1.864	0.872
MWLLM-CNN-SR	1.675	0.830	2.085	0.885	1.995	0.888	1.711	0.854	1.710	0.846	1.835	0.861
MWLCG-CNN-SR	1.643	0.829	2.004	0.886	1.924	0.887	1.731	0.860	1.670	0.841	1.794	0.861
MWLFC-CNN-SR	1.726	0.843	2.105	0.893	2.020	0.897	1.726	0.864	1.752	0.858	1.866	0.871
LogMel-CNN-MR	1.670	0.837	2.102	0.894	2.021	0.896	1.726	0.866	1.719	0.856	1.848	0.870
GTspec-CNN-MR	1.663	0.837	2.050	0.891	1.915	0.892	1.664	0.861	1.731	0.856	1.805	0.867
FCspec-CNN-MR	1.711	0.843	2.110	0.895	1.993	0.897	1.715	0.866	1.756	0.859	1.857	0.872
MWLLM-CNN-MR	1.667	0.827	2.082	0.884	1.985	0.887	1.700	0.853	1.715	0.845	1.830	0.859
MWLCG-CNN-MR	1.658	0.829	2.040	0.886	1.962	0.888	1.755	0.862	1.703	0.842	1.824	0.861
MWLFC-CNN-MR	1.743	0.845	2.130	0.896	2.047	0.899	1.753	0.868	1.774	0.861	1.889	0.874
Comb-DNN	1.238	0.653	1.255	0.676	1.247	0.696	1.235	0.652	1.271	0.695	1.249	0.674
LogMMSE	1.489	0.751	1.874	0.828	1.856	0.841	1.687	0.800	1.599	0.788	1.701	0.801
AudNoiseSup	1.147	0.698	1.227	0.776	1.236	0.789	1.119	0.682	1.228	0.733	1.191	0.736

TABLE A.8: Average PESQ and STOI metrics over all noisy and enhanced test utterances obtained from all methods investigated in this work. The five noise types used to corrupt clean test utterances are not seen during training and the SNR is 10 dB in this testing condition.

Feature type	**Babble**		**Factory2**		**Tank**		**F16**		**Train**		**Average**	
	PESQ	STOI	PESQ	STOI	PESQ	STOI	PESQ	STOI	PESQ	STOI	PESQ	STOI
Mixture	1.507	0.859	1.627	0.904	1.608	0.913	1.394	0.880	1.571	0.881	1.541	0.887
Spec-DNN	2.112	0.901	2.581	0.932	2.572	0.935	2.192	0.919	2.085	0.908	2.308	0.919
LogMel-DNN	2.119	0.901	2.583	0.931	2.566	0.934	2.175	0.918	2.083	0.906	2.305	0.918
GTspec-DNN	2.088	0.900	2.563	0.929	2.509	0.933	2.125	0.915	2.110	0.906	2.279	0.917
MRCG-16-DNN	2.053	0.889	2.422	0.921	2.405	0.925	2.048	0.909	2.029	0.893	2.191	0.907
MRCG-64-DNN	2.091	0.895	2.473	0.925	2.469	0.930	2.106	0.913	2.062	0.899	2.240	0.912
FCspec-DNN	2.199	0.900	2.608	0.927	2.515	0.932	2.201	0.916	2.145	0.905	2.334	0.916
MWLLM-DNN	2.031	0.889	2.424	0.922	2.392	0.927	1.965	0.905	1.982	0.895	2.159	0.908
MWLCG-DNN	2.066	0.887	2.456	0.919	2.458	0.925	2.101	0.908	2.039	0.888	2.224	0.905
MWLFC-DNN	2.206	0.901	2.606	0.927	2.501	0.931	2.228	0.916	2.154	0.905	2.339	0.916
LogMel-CNN-SR	2.120	0.902	2.588	0.931	2.550	0.934	2.193	0.918	2.104	0.907	1.542	0.918
GTspec-CNN-SR	2.112	0.903	2.541	0.931	2.435	0.934	2.147	0.918	2.125	0.908	2.311	0.919
FCspec-CNN-SR	2.196	0.902	2.612	0.929	2.540	0.932	2.209	0.918	2.166	0.906	2.272	0.917
MWLLM-CNN-SR	2.120	0.892	2.535	0.923	2.516	0.927	2.184	0.910	2.104	0.897	2.345	0.910
MWLCG-CNN-SR	2.086	0.894	2.476	0.925	2.430	0.929	2.187	0.914	2.062	0.897	2.292	0.912
MWLFC-CNN-SR	2.190	0.902	2.592	0.929	2.548	0.933	2.215	0.917	2.161	0.907	2.248	0.918
LogMel-CNN-MR	2.119	0.901	2.579	0.931	2.559	0.934	2.202	0.920	2.104	0.907	2.313	0.919
GTspec-CNN-MR	2.113	0.900	2.473	0.926	2.396	0.929	2.131	0.915	2.105	0.905	2.244	0.915
FCspec-CNN-MR	2.187	0.903	2.617	0.930	2.536	0.933	2.212	0.919	2.163	0.907	2.343	0.918
MWLLM-CNN-MR	2.120	0.891	2.537	0.923	2.522	0.927	2.170	0.910	2.115	0.898	2.293	0.910
MWLCG-CNN-MR	2.108	0.894	2.506	0.925	2.466	0.929	2.215	0.915	2.108	0.897	2.281	0.912
MWLFC-CNN-MR	2.210	0.904	2.619	0.931	2.571	0.935	2.238	0.920	2.181	0.909	2.364	0.920
Comb-DNN	1.295	0.700	1.292	0.705	1.282	0.719	1.264	0.688	1.317	0.726	1.290	0.708
LogMMSE	1.841	0.846	2.286	0.895	2.319	0.903	2.102	0.882	1.947	0.867	2.099	0.879
AudNoiseSup	1.313	0.814	1.470	0.865	1.488	0.865	1.249	0.818	1.434	0.828	1.391	0.838

Appendix B

Publications

B.1 Time Domain Solutions for the Neely and Kim [9] Model of the Cochlea

B.2 Comparison of Two Cochlear Models

Efficient time-domain simulation of nonlinear, state-space, transmission-line models of the cochlea (L)

Shuokai Pan[a] and Stephen J. Elliott
Institute of Sound and Vibration Research, University of Southampton, Highfield Campus, Southampton SO17 1BJ, United Kingdom

Paul D. Teal
School of Engineering and Computer Science, Victoria University of Wellington, P.O. Box 600, Wellington 6140, New Zealand

Ben Lineton
Institute of Sound and Vibration Research, University of Southampton, Highfield Campus, Southampton SO17 1BJ, United Kingdom

(Received 18 February 2015; revised 1 May 2015; accepted 8 May 2015)

Nonlinear models of the cochlea are best implemented in the time domain, but their computational demands usually limit the duration of the simulations that can reasonably be performed. This letter presents a modified state space method and its application to an example nonlinear one-dimensional transmission-line cochlear model. The sparsity pattern of the individual matrices for this alternative formulation allows the use of significantly faster numerical algorithms. Combined with a more efficient implementation of the saturating nonlinearity, the computational speed of this modified state space method is more than 40 times faster than that of the original formulation.
© 2015 Author(s). All article content, except where otherwise noted, is licensed under a Creative Commons Attribution 3.0 Unported License. [http://dx.doi.org/10.1121/1.4921550]

[BLM] Pages: 3559–3562

I. INTRODUCTION

Active nonlinearity is an essential feature of cochlear mechanics that is generally best modelled in the time domain. A range of time domain numerical methods have been proposed and applied to a number of models over the years (Diependaal *et al.*, 1987; Elliott *et al.*, 2007). One widely adopted two-stage strategy (Diependaal *et al.*, 1987) first solves the model boundary value problem using the finite difference approximation, in order to compute the pressure difference, and then uses numerical integration techniques to solve the remaining initial value problem. More recently, a state space matrix formulation has been proposed in Elliott *et al.* (2007), which has also been applied in several studies such as Liu and Neely (2010), Sisto *et al.* (2010), Rapson *et al.* (2012), and Ayat *et al.* (2014). It expresses the model states as a single set of coupled first-order ordinary differential equations (ODEs), the time domain solution of which can then be obtained using well-established ODE solvers. An extra benefit of this approach is that it allows rigorous inspection of the stability of linearized models, which, for instance, facilitates the study of spontaneous otoacoustic emissions (Ku *et al.*, 2009). The relation between the state space formulation and Diependaal's method has previously been discussed in Rapson *et al.* (2012).

Despite the advantages of time domain solutions, they are usually very computationally challenging and memory demanding, because a sufficient spatial and time domain resolution is required to ensure simulation accuracy. Taking the standard state space method as an example, it is typically thousands of times slower than real time when implemented in the time domain on a desktop computer. A hybrid direct-iterative solver was developed in Bertaccini and Sisto (2011) to accelerate the simulation of a nonlinear feed-forward model within the standard state space framework. This letter proposes a modified state space (MSS) method for the time domain simulation of nonlinear one-dimensional (1D) transmission-line cochlear models. Numerical efficiency of this MSS is compared with both the standard state space (SS) and Diependaal's method, and it is found to be more than 40 times more efficient than the SS and marginally faster than Diependaal's method.

II. THE EXAMPLE COCHLEAR MODEL

The example model adopted here for comparison of algorithm complexity is that reported in Ku *et al.* (2009), which is a nonlinear adaptation to the linear active model originally proposed by Neely and Kim (1986) for a cat cochlea, but with parameters re-tuned to match human cochlear physiology. The micromechanics of this model is represented by an array of 500 discrete elements, with the first one modelling the middle ear, the last one the helicotrema and the remaining 498 elements modelling the cochlear partition, as shown in Fig. 4 of Elliott *et al.* (2007). The cochlear partition elements are modelled as two degree-of-freedom (DOF) lumped-parameter oscillators [Fig. 1 of Elliott *et al.* (2007)], coupled solely by the cochlear fluid. The saturating active mechanism is simulated by placing a compressive nonlinearity, a first-order Boltzmann function, before the active impedance in the micromechanical feedback loop, so that the input to the Boltzmann function is the difference between the displacements of the two masses of each oscillator. The dimensionless factor, γ, which regulates the strength of the active feedback

[a]Electronic mail: sp2g12@soton.ac.uk

Redistribution subject to ASA license or copyright; see http://acousticalsociety.org/content/terms. Download to IP: 46.233.116.170 On: Sun, 23 Oct 2016 23:08:58

force, is then determined as the absolute value of the ratio of the output and input of the Boltzmann function, as shown in Eq. (8) of Ku *et al.* (2009). A two-port network model of the ear canal and middle ear, based on the model of Kringlebotn (1988) and programmed by Ku (2008), is also included. A thorough description of the model and its parameter values can be found in Ku (2008).

III. MODIFIED STATE SPACE FORMULATION

The original state space formulation proposed by Elliott *et al.* (2007) used the finite difference approximation for the 1D wave propagation inside the cochlea, Eq. (1), and state space formulation for the element micromechanics, Eqs. (2) and (3), so that the final fluid-coupled state space equation can be derived as Eq. (4),

$$\mathbf{F}\mathbf{p}(t) - \ddot{\mathbf{w}}(t) = \mathbf{q}(t), \tag{1}$$

$$\dot{\mathbf{x}}(t) = \mathbf{A}_E\mathbf{x}(t) + \mathbf{B}_E\mathbf{p}(t), \tag{2}$$

$$\dot{\mathbf{w}}(t) = \mathbf{C}_E\mathbf{x}(t), \tag{3}$$

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{q}(t), \tag{4}$$

where $\mathbf{F}$ is the second-order finite difference matrix, representing the coupling between the cochlear partition acceleration vector $\ddot{\mathbf{w}}(t)$ and fluid pressure difference vector $\mathbf{p}(t)$; $\mathbf{q}(t)$ is the stapes acceleration vector, the only nonzero entry of which is the first one; $\mathbf{x}(t)$ is the complete state vector; $\mathbf{A} = [\mathbf{I} - \mathbf{B}_E\mathbf{F}^{-1}\mathbf{C}_E]^{-1}\mathbf{A}_E$ is the system matrix; and $\mathbf{B} = [\mathbf{I} - \mathbf{B}_E\mathbf{F}^{-1}\mathbf{C}_E]^{-1}\mathbf{B}_E\mathbf{F}^{-1}$ is the input matrix. Complete derivations and definitions of these matrices can be found in Elliott *et al.* (2007). An alternative method for deriving the final state space equation, with much better computational efficiency, is to first combine Eqs. (2) and (3) to eliminate variable $\dot{\mathbf{x}}(t)$, and then combine the result with Eq. (1) to eliminate dependence on $\ddot{\mathbf{w}}(t)$,

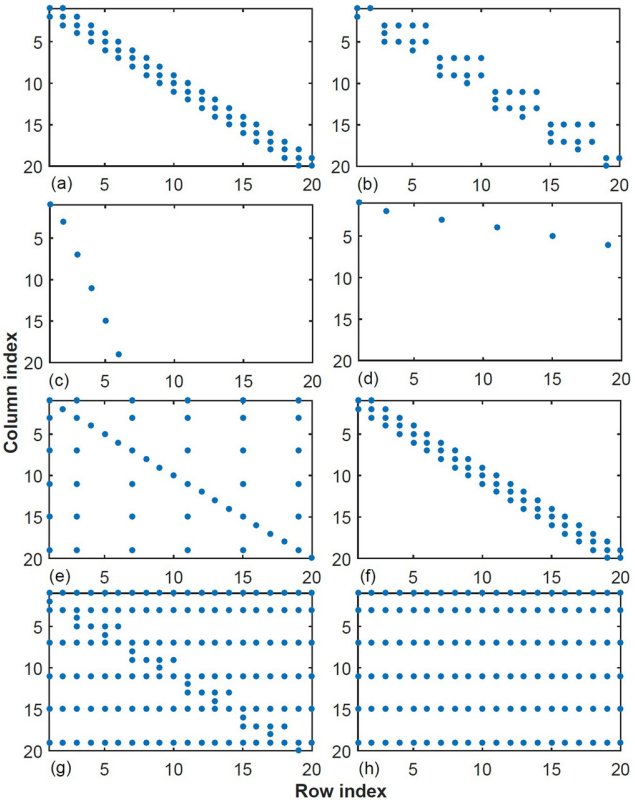


FIG. 1. (Color online) Sparsity patterns, sizes, nnz, and matrix densities of state space matrices. Only the first 20×20 elements of each matrix are displayed for better visualization of their internal patterns. (a) $\mathbf{F}$, 500×500, nnz $= 1498$, $\delta \approx 0.599\%$; (b) $\mathbf{A}_E$, 1996×1996, nnz $= 4985$, $\delta \approx 0.125\%$; (c) $\mathbf{B}_E$, 1996×500, nnz $= 500$, $\delta \approx 0.0501\%$; (d) $\mathbf{C}_E$, 500×1996, nnz $= 500$, $\delta \approx 0.0501\%$; (e) $\mathbf{I} - \mathbf{B}_E\mathbf{F}^{-1}\mathbf{C}_E$, 1996×1996, nnz $= 251496$, $\delta \approx 6.31\%$; (f) $\mathbf{F} - \mathbf{C}_E\mathbf{B}_E$, 500×500, nnz $= 1498$, $\delta \approx 0.599\%$; (g) $\mathbf{A}$ or $\mathbf{A}'$, 1996×1996, nnz $= 1\,000\,490$, $\delta \approx 25.1\%$; and (h) $\mathbf{B}$ or $\mathbf{B}'$, 1996×500, and nnz $= 250\,000$, $\delta \approx 25.05\%$.

$$\ddot{\mathbf{w}}(t) = \mathbf{C}_E\dot{\mathbf{x}}(t) = \mathbf{C}_E[\mathbf{A}_E\mathbf{x}(t) + \mathbf{B}_E\mathbf{p}(t)], \tag{5}$$

$$\ddot{\mathbf{w}}(t) = \mathbf{F}\mathbf{p}(t) - \mathbf{q}(t). \tag{6}$$

Equating the right hand side of Eqs. (5) and (6) leads to the pressure difference vector as

$$\mathbf{p}(t) = (\mathbf{F} - \mathbf{C}_E\mathbf{B}_E)^{-1}[\mathbf{C}_E\mathbf{A}_E\mathbf{x}(t) + \mathbf{q}(t)]. \tag{7}$$

Substituting Eq. (7) back into Eq. (2), the modified state space equation is obtained as

$$\dot{\mathbf{x}}(t) = \mathbf{A}'\mathbf{x}(t) + \mathbf{B}'\mathbf{q}(t), \tag{8}$$

where the new system matrix becomes $\mathbf{A}' = \mathbf{A}_E + \mathbf{B}_E(\mathbf{F} - \mathbf{C}_E\mathbf{B}_E)^{-1}\mathbf{C}_E\mathbf{A}_E$; and new input matrix is $\mathbf{B}' = \mathbf{B}_E(\mathbf{F} - \mathbf{C}_E\mathbf{B}_E)^{-1}$. This new set of state space matrices $\mathbf{A}'$, $\mathbf{B}'$ can be shown to be analytically identical to the original ones $\mathbf{A}$, $\mathbf{B}$, using the Woodbury matrix identity for the inverse of a sum of matrices (Henderson and Searle, 1981). Applying this identity to the original system matrix $\mathbf{A}$, we have

$$\begin{aligned}\mathbf{A} &= [\mathbf{I} - \mathbf{B}_E\mathbf{F}^{-1}\mathbf{C}_E]^{-1}\mathbf{A}_E \\ &= [\mathbf{I} + \mathbf{B}_E(\mathbf{F} - \mathbf{C}_E\mathbf{B}_E)^{-1}\mathbf{C}_E]\mathbf{A}_E = \mathbf{A}'.\end{aligned} \tag{9}$$

Similarly, for the original input matrix $\mathbf{B}$, we have

$$\begin{aligned}\mathbf{B} &= [\mathbf{I} - \mathbf{B}_E\mathbf{F}^{-1}\mathbf{C}_E]^{-1}\mathbf{B}_E\mathbf{F}^{-1} \\ &= [\mathbf{I} - \mathbf{B}_E\mathbf{F}^{-1}\mathbf{C}_E]^{-1}\mathbf{B}_E\mathbf{F}^{-1}\mathbf{C}_E\mathbf{C}_E^{-1} \\ &= [\mathbf{I} - \mathbf{B}_E\mathbf{F}^{-1}\mathbf{C}_E]^{-1}[\mathbf{I} - (\mathbf{I} - \mathbf{B}_E\mathbf{F}^{-1}\mathbf{C}_E)]\mathbf{C}_E^{-1} \\ &= [\mathbf{I} - \mathbf{B}_E\mathbf{F}^{-1}\mathbf{C}_E]^{-1}\mathbf{C}_E^{-1} - \mathbf{C}_E^{-1} \\ &= [\mathbf{I} + \mathbf{B}_E(\mathbf{F} - \mathbf{C}_E\mathbf{B}_E)^{-1}\mathbf{C}_E]\mathbf{C}_E^{-1} - \mathbf{C}_E^{-1} \\ &= \mathbf{B}_E(\mathbf{F} - \mathbf{C}_E\mathbf{B}_E)^{-1} = \mathbf{B}'.\end{aligned} \tag{10}$$

IV. NUMERICAL ADAVANTAGE OF THE MODIFIED STATE SPACE METHOD

As shown in Eqs. (10), (21)–(23) of Elliott *et al.* (2007), $\mathbf{F}$ is a tri-diagonal matrix (i.e., having nontrivial elements only on the diagonal, first sub-diagonal and first super-diagonal), $\mathbf{A}_E$, $\mathbf{B}_E$, and $\mathbf{C}_E$ are block diagonal matrices (in which the diagonal elements are square matrices of any size and off-diagonal elements are zero). It is therefore expected that considerable memory and calculation savings can be achieved by employing sparse matrix storage and computation methods, especially when several hundred micro-elements are used. The sparsity patterns of the state space matrices are shown in Fig.

Redistribution subject to ASA license or copyright; see http://acousticalsociety.org/content/terms. Download to IP: 46.233.116.170 On: Sun, 23 Oct 2016 23:08:58

1, where the upper two rows show the individual matrices and the bottom two rows both those inverted and the final matrices in the state space formulation. Each square represents the structure of a matrix and each dot denotes a nonzero element at the corresponding location. Only the first 20×20 elements of each matrix are depicted for better visualization of their internal patterns. The size of each matrix for a 500-element model, the number of nonzero entries (nnz) and sparse matrix density, δ, defined as the ratio of nnz to the total number of entries, are also given in the figure caption. It can be seen that more than 99% of the matrices $\mathbf{A}_E$, $\mathbf{B}_E$, $\mathbf{C}_E$, $\mathbf{F}$, and $\mathbf{F}-\mathbf{C}_E\mathbf{B}_E$ are empty, whereas this number reduces to about 75% for the final set of state space matrices $\mathbf{A}$ and $\mathbf{B}$. In MATLAB, sparse algorithms (Gilbert *et al.*, 1992) can be employed when these matrices are defined to be sparse and the resultant sparse state space method is about twice as efficient as the original formulation. The modified state space formulation, however, can be arranged to be even faster with sparse algorithms, because the intermediate matrix $\mathbf{F}-\mathbf{C}_E\mathbf{B}_E$ is tri-diagonal and the calculation of the pressure difference vector $\mathbf{p}(t)$ in Eq. (7) is identical to solving a sparse tri-diagonal linear system of the form $Ax=b$. There exist very rapid algorithms for computing the solution of such linear system using either iterative or direct methods. Direct solvers based upon sparse matrix factorization are preferable in this case due to the dimension of the model and the nature of matrix $\mathbf{F}-\mathbf{C}_E\mathbf{B}_E$. In MATLAB, the matrix left divide operator, mldivde or backslash (\), encapsulates a host of algorithms for solving sparse linear systems and invokes the most suitable one according to the sparsity pattern of involved matrices (Davis, 2006). One solver is specifically designed for tri-diagonal systems, and direct use of this algorithm leads to a speed improvement of at least 20 times compared to the original formulation. But since the tri-diagonal matrix, $\mathbf{F}-\mathbf{C}_E\mathbf{B}_E$, is fixed during the entire simulation, the speed of solving this linear system can be further increased by performing a sparse LU decomposition beforehand. Essentially, this converts the tri-diagonal system into a combination of a lower and an upper triangular system which can be easily solved using forward and backward substitution. According to (Davis, 2004), this decomposition for the matrix $\mathbf{F}-\mathbf{C}_E\mathbf{B}_E$ takes the form of

$$\mathbf{F}-\mathbf{C}_E\mathbf{B}_E=\mathbf{R}\mathbf{P}^{-1}\mathbf{L}\mathbf{U}\mathbf{Q}^{-1}, \tag{11}$$

$$(\mathbf{F}-\mathbf{C}_E\mathbf{B}_E)^{-1}=\mathbf{Q}\mathbf{U}^{-1}\mathbf{L}^{-1}\mathbf{P}\mathbf{R}^{-1}, \tag{12}$$

where $\mathbf{L}$ is a lower triangular matrix, $\mathbf{U}$ is an upper triangular matrix, $\mathbf{P}$ and $\mathbf{Q}$ are row and column permutation matrices used to reduce the fill-in problem during the actual factorization process, and $\mathbf{R}$ is a diagonal row scaling matrix which can lead to a sparser and more stable factorization. Thus, in addition to using sparse matrix multiplication, the modified state space equation is implemented as the following:

$$\dot{\mathbf{x}}(t)=\mathbf{A}_E\mathbf{x}(t)+\bar{\mathbf{B}}\{\mathbf{U}^{-1}[\mathbf{L}^{-1}(\bar{\mathbf{C}}\mathbf{A}_E\mathbf{x}(t))]\}+\bar{\mathbf{q}}(t), \tag{13}$$

where $\bar{\mathbf{B}}=\mathbf{B}_E\mathbf{Q}$, $\bar{\mathbf{C}}=\mathbf{P}\mathbf{R}^{-1}\mathbf{C}_E$, and $\bar{\mathbf{q}}(t)=\mathbf{B}_E(\mathbf{F}-\mathbf{C}_E\mathbf{B}_E)^{-1}\mathbf{q}(t)$, all of which can be computed before calling the ODE solver. Such computational advantages are not possible with the original state space method, as its intermediate matrix, $\mathbf{I}-\mathbf{B}_E\mathbf{F}^{-1}\mathbf{C}_E$ contains significantly more nonzero entries (more than 167 times) and is of larger size (about 16 times) than those of $\mathbf{F}-\mathbf{C}_E\mathbf{B}_E$. The reduction of sparsity mainly results from $\mathbf{F}^{-1}$, which is a completely full matrix.

V. INCLUSION OF NONLINEARITY IN THE STATE SPACE EQUATIONS

Only the active feedback force needs to be compressed with stimulus level, so the matrix $\mathbf{A}_E$ can be decomposed into a time-invariant passive part, $\mathbf{A}_{E_{\text{pas}}}$ and a time-varying active part, $\mathbf{A}_{E_{\text{act}}}(\gamma)$, which consists of all the components that are functions of γ. One way of implementing the level-dependent nonlinearity in the time domain is to update $\mathbf{A}_{E_{\text{act}}}(\gamma)$ every time γ changes. This involves several modifications to a 1996×1996 matrix for every time step when using the ODE solver and can be extremely time-consuming. By observing that each individual block matrix inside $\mathbf{A}_{E_{\text{act}}}(\gamma)$ only contains nonzero entries on its first row, which also have a common factor γ, an equivalent implementation is to scale all the components of the state vector for each cochlear partition element by a factor $\gamma(n, t)$, which is given by

$$\gamma(n,t)=\left|\frac{f[x_d(n,t)]}{x_d(n,t)}\right|, \tag{14}$$

where n is the element index; $x_d(n, t)$ is the relative displacement between the BM and TM; and $f(x)$ is the Boltzmann function. The active part of the state space equation, $\mathbf{A}_{E_{\text{act}}}(\gamma)\mathbf{x}(t)$, can now be written as $\mathbf{A}_{E_{\text{full}}}\mathbf{x}_{\text{scaled}}(t)$, where $\mathbf{A}_{E_{\text{full}}}=\mathbf{A}_{E_{\text{act}}}(\gamma=1)$, is a constant matrix, $\mathbf{x}_{\text{scaled}}(t)=\boldsymbol{\gamma}(t)\odot\mathbf{x}(t)$, $\boldsymbol{\gamma}(t)$ is a column vector including all of the scaling factors, $\gamma(n, t)$, arranged in the same order as are the elements of the complete state vector $\mathbf{x}(t)$ and the symbol $\odot$ denotes element-by-element multiplication of the two column vectors $\boldsymbol{\gamma}(t)$ and $\mathbf{x}(t)$. Therefore, the final fluid-coupled state space equation for the nonlinear cochlear model is realized as the following:

$$\begin{aligned}\dot{\mathbf{x}}(t)=&\mathbf{A}_{E_{\text{pas}}}\mathbf{x}(t)+\mathbf{A}_{E_{\text{full}}}\mathbf{x}_{\text{scaled}}(t)+\bar{\mathbf{q}}(t)\\&+\bar{\mathbf{B}}\{\mathbf{U}^{-1}[\mathbf{L}^{-1}(\bar{\mathbf{C}}(\mathbf{A}_{E_{\text{pas}}}\mathbf{x}(t)+\mathbf{A}_{E_{\text{full}}}\mathbf{x}_{\text{scaled}}(t)))]\}.\end{aligned} \tag{15}$$

This method is considerably more effective than the first one, as none of the state space matrices is changed during the numerical integration. The original SS method for the nonlinear model is implemented in a similar way for comparison of numerical efficiency.

VI. RESULTS

This section presents a comparison of the computational efficiency of four time domain numerical algorithms: the SS, sparse state space (SSS), modified state space (MSS), and Diependaal's two-stage method after extending it to allow two DOF micromechanics following the Appendix of Diependaal *et al.* (1987). All of these were programmed and simulated in MATLAB R2015a using a desktop computer with a 3.40 GHz, quad-core Intel Core i5-3570 processor and 4 GB DDR3 RAM. The final component of all four algorithms is numerical integration. We implemented an explicit adaptive solver, which is a modified version of the MATLAB ode45 function,

Redistribution subject to ASA license or copyright; see http://acousticalsociety.org/content/terms. Download to IP: 46.233.116.170 On: Sun, 23 Oct 2016 23:08:58

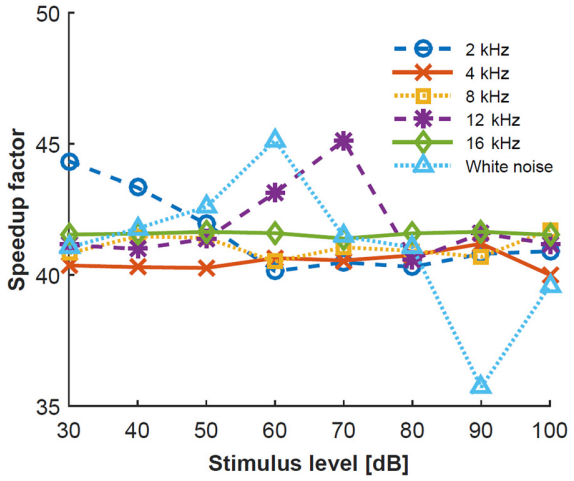


FIG. 2. (Color online) Dependence of speedup factor for the modified state space method on stimulus SPL for white noise and sinusoidal stimuli with frequencies of 2, 4, 8, 12, and 16 kHz.

removing unnecessary functionalities such as event function, mass matrix and non-negative solutions. The relative error tolerance was set as 10^{-5} in all experiments, whereas the absolute error tolerance was determined individually for each stimulus to be one order of magnitude lower than the corresponding scale of the model displacement response. This reflects the flexibility of the adaptive solver, which can solve the model with the required accuracy and without unnecessary effort. The effect of the stimulus spectrum and level on program runtime was investigated by using white noise and sinusoids having frequencies of 2, 4, 8, 12, and 16 kHz, each of which was sampled at 100 kHz, and had a duration of 30 ms with a 10 ms half-Hanning window onset ramp and levels varying from 30 to 100 dB sound pressure level (SPL) in step of 10 dB.

For each stimulus, the simulation time gradually rises with increasing input level and frequency for sinusoidal stimuli. But the speedup factor, defined as the ratio of the SS simulation time to that of any other method, is roughly constant across most stimulus types and levels, as shown in Fig. 2 for the MSS. Similarly, the speedup factor is found to be relatively independent of the number of microelements in the model (not shown). Table I shows the average runtime across different signal levels and frequencies of sinusoidal excitations for each type of stimulus and algorithm. The overall average runtime of all stimuli are taken as the final metric for comparison of the computational efficiency of each method and the overall speedup factor is shown in the last row of Table I. It can be seen that the sparse algorithms alone are able reduce the runtime by almost a half, but the MSS yields a speed improvement of more than a factor of 40 and is also about 40% quicker than the two-stage method.

TABLE I. Comparison of average run-time in seconds, used by four algorithms to solve the nonlinear 1D cochlear model with 500 discrete elements in response to white noise and sinusoidal stimuli with frequencies of 2, 4, 8, 12, and 16 kHz, each of which was sampled at 100 kHz, had a duration of 30 ms with a 10 ms half-Hanning window onset ramp and levels varying from 30 to 100 dB SPL in step of 10 dB. SS: original dense state space; SSS: sparse state space; MSS: modified state space; two-stage: Diependaal's method.

	SS	SSS	MSS	Two-stage
Sinusoids	777.02	427.50	18.80	25.52
White noise	630.54	361.85	15.52	22.27
Average runtime	703.78	394.68	17.16	23.90
Speedup factor	1	1.78	41.01	29.45

VII. CONCLUSIONS

The main contribution of this paper is the description of the modified state space method, which has been applied here to the nonlinear and active cochlear model developed in Ku *et al.* (2009). Although analytically identical to the original state space equation, the sparsity pattern of the constituting matrix of this alternative formulation offers the opportunity for considerably more efficient numerical algorithms, producing a speedup factor of more than 40. Its computational efficiency is on a similar scale to that of the Diependaal's method. The approach presented here can be readily applied to various other 1D cochlear models, such as the nonlinear and active ones based on an array of one DOF lumped-parameter oscillators (Sisto *et al.*, 2010).

ACKNOWLEDGMENT

The Ph.D. work of S.P. is supported by Cirrus Logic.

Ayat, M., Teal, P. D., and McGuinness, M. (**2014**). "An integrated electro-mechanical model for the cochlear microphonic," Biocybern. Biomed. Eng. **34**, 206–219.
Bertaccini, D., and Sisto, R. (**2011**). "Fast numerical solution of nonlinear nonlocal cochlear models," J. Comput. Phys. **230**, 2575–2587.
Davis, T. A. (**2004**). "A column pre-ordering strategy for the unsymmetric-pattern multifrontal method," ACM Trans. Math. Software **30**, 165–195.
Davis, T. A. (**2006**). *Direct Methods for Sparse Linear Systems* (Society for Industrial and Applied Mathematics, Philadelphia, PA), Chap. 8, pp. 135–144).
Diependaal, R. J., Duifhuis, H., Hoogstraten, H., and Viergever, M. A. (**1987**). "Numerical methods for solving one-dimensional cochlear models in the time domain," J. Acoust. Am. Soc. **82**, 1655–1666.
Elliott, S. J., Ku, E. M., and Lineton, B. (**2007**). "A state space model for cochlear mechanics," J. Acoust. Am. Soc. **122**, 2759–2771.
Gilbert, J. R., Moler, C., and Schreiber, R. (**1992**). "Sparse matrices in MATLAB: Design and implementation," SIAM J. Matrix Anal. Appl. **13**, 333–356.
Henderson, H. V., and Searle, S. R. (**1981**). "On deriving the inverse of a sum of matrices," SIAM Rev. **23**, 53–60.
Kringlebotn, M. (**1988**). "Network model for the human middle ear," Scand. Audiol. **17**, 75–85.
Ku, E. M. (**2008**). "Modelling the human cochlea," Ph.D. thesis, University of Southampton, Southampton, UK.
Ku, E. M., Elliott, S. J., and Lineton, B. (**2009**). "Limit cycle oscillations in a nonlinear state space model of the human cochlea," J. Acoust. Am. Soc. **126**, 739–750.
Liu, Y.-W., and Neely, S. T. (**2010**). "Distortion product emissions from a cochlear model with nonlinear mechanoelectrical transduction in outer hair cells," J. Acoust. Am. Soc. **127**, 2420–2432.
Neely, S. T., and Kim, D. (**1986**). "A model for active elements in cochlear biomechanics," J. Acoust. Soc. Am. **79**, 1472–1480.
Rapson, M. J., Tapson, J. C., and Karpul, D. (**2012**). "Unification and extension of monolithic state space and iterative cochlear models," J. Acoust. Soc. Am. **131**, 3935–3952.
Sisto, R., Moleti, A., Paternoster, N., Botti, T., and Bertaccini, D. (**2010**). "Different models of the active cochlea, and how to implement them in the state-space formalism," J. Acoust. Soc. Am. **128**, 1191–1202.

Redistribution subject to ASA license or copyright; see http://acousticalsociety.org/content/terms. Download to IP: 46.233.116.170 On: Sun, 23 Oct 2016 23:08:58

COMPARISON OF THE NONLINEAR RESPONSES OF A TRANSMISSION-LINE AND A FILTER CASCADE MODEL OF THE HUMAN COCHLEA

Shuokai Pan, Stephen J. Elliott, Dario Vignali*

Institute of Sound and Vibration, University of Southampton, UK

ABSTRACT

Models of the mammalian cochlea have been proposed in a number of ways and they have varying degree of realism and complexity. The transmission-line (TL) models are faithful to the physiology, particularly in terms of cochlear nonlinearity, but are computationally demanding. The pole-zero filter cascade (PZFC) model, however, is much more efficient to implement, but the nonlinearity is included implicitly, using an automatic gain control network. In this study, the connection between the linear responses of these two models is first discussed, followed by a comparison of their nonlinear responses in terms of self-suppression and two-tone suppression on the level of the basilar membrane. Both models are capable of simulating dynamic range compression as measured on the cochlear partition, but the TL model is more reasonable in representing two-tone suppression, with the PZFC having lower suppression thresholds and over-predicting the suppression due to high-side suppressors. Further tuning of its parameters and structure, especially the automatic gain control (AGC) network may be possible to make it more compatible with these experimental observations. After adapting the PZFC model to have a more realistic nonlinear behavior, its use for investigating auditory signal processing such as masking effects, and hence as a front-end processor for acoustic signals can be enhanced, while retaining its computational efficiency.

Index Terms— Transmission-line cochlear model, pole-zero filter cascade, self-suppression, two-tone suppression

1. INTRODUCTION

Models of the mammalian cochlea can be very beneficial, not only in understanding various aspects of cochlear physiology and auditory perception, but also in inspiring effective and robust algorithms for processing complex stimuli like speech and music. The transmission-line (TL) type of model [1] is more physiologically based than filterbank ones, and thus more accurate in simulating cochlear mechanics, but it is also more computationally challenging when implemented in the time domain. Parallel filterbank models, on the other hand, are only concerned with reproducing measured cochlear responses and pay little attention to the underlying biophysics, although they can be much more efficient to implement. The pole-zero filter cascade (PZFC) model [2–4] sits between these two classes, since it is directly inspired by the cochlear traveling waves, with each serial section being a simple digital filter modeling the frequency response of basilar membrane (BM) vibrations from one point to the next. It thus enjoys both the physiological plausibility of the TL models and a comparable level of numerical efficiency of parallel filterbank-based ones, achieving a good compromise between model capacity and complexity.

*Shuokai Pan is supported by Cirrus Logic.

The normal cochlea is known to possess an active process, which amplifies the BM responses to low-level single-tone stimuli, but gradually decreases its gain with increasing stimulus intensity. This phenomenon is known as self-suppression or dynamic-range-compression. Such nonlinearity residing in the cochlear amplifier can cause various nonlinear interactions between different spectral components of more complex stimuli, such as two tones. It has been demonstrated that the mechanical vibrations of the BM in response to a probe tone can be reduced by a simultaneous presence of a second tone, which is referred to as two-tone suppression. Suppressor with frequency higher than the probe frequency is termed high-side suppressor, while the opposite is called low-side suppressor. Similar suppression effect has also been observed in the receptor potentials of the inner hair cells (IHCs) [5], the neural firing rate of auditory nerve fibers [6] and psychophysical experiments with human subjects [7]. Both of these nonlinear phenomena are particularly important in the study of many areas of auditory signal processing, such as the peripheral encoding of speech and music stimuli and the determination of masking effects.

In this paper, after reviewing the formal connections between the linear responses of a TL and the PZFC models, a comparison of their nonlinear responses in terms of self-suppression and two-tone suppression on the level of BM is presented. As shown later, both models are reasonably accurate in simulating self-suppression, whereas the PZFC is less realistic than the TL model in predicting two-tone suppression.

2. MODEL DESCRIPTION

2.1. The Transmission-line Model

The transmission-line model adopted here is a one-dimensional, active and nonlinear one, based on the model originally proposed by Kanis and de Boer [8], but with parameters as those used by Young [9]. It simplifies the biological cochlea into a symmetric fluid-filled rectangular box, separated by a flexible partition representing the BM, as shown in Fig. 5.1 in [1]. An array of about 500 discrete micro-mechanical elements are used to represent the responses of the BM and organ of Corti (OC) along the 35 mm length of the human cochlea, each of which is a two degree-of-freedom (DOF) lumped-parameter oscillators (see Fig. 1 of [10]), coupled solely by the cochlear fluid. The saturating cochlear amplifier is simulated explicitly with a first-order Boltzmann function placed before the active impedance in the feedback loop of each oscillator, as illustrated in Fig. 1(b) of [11]. A two-port network model of the ear canal and middle ear, based on the model in [12] and programmed in [13], is also included. Model responses are computed in the time domain using a fast modified state-space method [14], which is about 40 times more efficient than the original state-space method [10].

978-1-4799-7450-4/15/$31.00 ©2015 IEEE

2.2. The Filter Cascade

Assuming that only a long-wave propagates inside the cochlea and the model parameters vary slowly in the longitudinal direction such that the relative change of wavenumber over one wavelength is negligible [1], the cochlear partition velocity in the above described TL model can be derived from the WKB approximation [15] as,

$$v(x) = Bk^{3/2}(x)e^{-i\Phi(x)} \tag{1}$$

where B is an amplitude factor determined by the excitation. $k(x)$ and $\Phi(x)$ are the wavenumber and phase integral respectively, both of which are also complex functions of frequency, although this dependence is suppressed for notational convenience. Their analytical expressions are given by,

$$k^2(x) = \frac{-2i\omega\rho}{HZ_{cp}(x)}, \Phi(x) = \int_0^x k(x')\,\mathrm{d}x' \tag{2}$$

where ρ is fluid density, H is the height of cochlear chamber and Z_{cp} is cochlear partition impedance. The frequency response for the BM motion of a longitudinal segment of the cochlea, extending from x_0 to $x_0 + \Delta$, can thus be written as,

$$H_\Delta = \frac{v(x_0+\Delta)}{v(x_0)} = \left(\frac{k(x_0+\Delta)}{k(x_0)}\right)^{3/2} e^{-i\int_{x_0}^{x_0+\Delta} k(x')\,\mathrm{d}x'} \tag{3}$$

The responses of the whole cochlea can be regarded as a cascade of filters with frequency responses given by (3). Other methods of deriving the cascaded filters from TL models have also been described by Kates [16], for instance. Lyon [17] experimented with a number of different filters and chose one with a single pair of poles and single pair of zeros in the Laplace plane, to reduce the group delay while maintaining a sharp enough high-side roll-off. Only the exponential of the phase integral in (3) was considered in that paper, however, and the wavenumber ratio (WNR) factor, $(k(x_0+\Delta)/k(x_0))^{3/2}$, was ignored. Since the complex wavenumber changes along the length of the cochlea, this term contributes to variations in both magnitude and phase. The influence of neglecting this WNR term is further investigated in Section 3.

To account for the observed cochlear nonlinearity, the dampings of pole and zero pairs are dynamically adjusted in the PZFC model in proportion to the filter outputs to turn up or down the gain of each stage using a combination of level detector and AGC network [18]. Early version use a simple half-wave-rectifier (HWR) as an energy estimator, while the more recent one employed a dedicated inner hair cell (IHC) model adapted from that proposed in [19]. The AGC filter of each stage is a set of four first-order low-pass filters with increasing time constant, which provides temporal smoothing to the outputs of the IHC model. By coupling each of them with their nearest neighbors, it also allows the activity in one channel to influence the gains of nearby channels, making it possible to simulate two-tone suppression and lateral inhibition. The PZFC generates outputs of both BM displacement and neural activity pattern, but only the mechanical vibrations are considered here in order to compare with the TL model. Even though the scale of its BM output is arbitrary, we refer to it as BM motion in later sections. All of the simulations with the PZFC are performed using the default parameter values specified in [20], except that the model sample rate is increased from 22.05 kHz to 100 kHz, to be the same as that of the time domain simulation of the TL model and a total number of 85 channels are used covering the frequency range from around 20 Hz to 20 kHz.

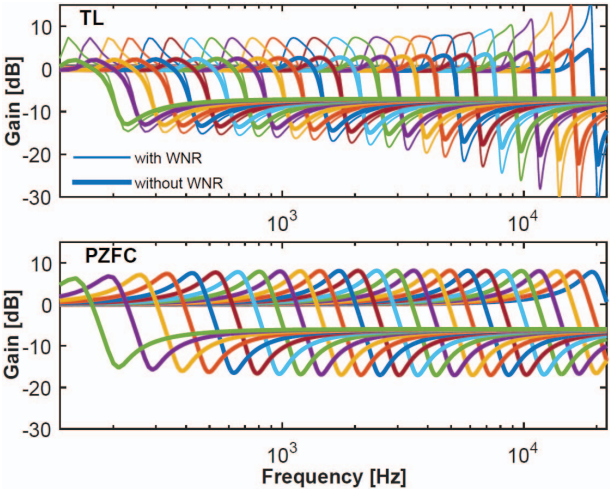


Figure 1: Frequency responses of the individual filters in the cascade computed from the TL model (top panel) with (thin) and without (thick) the WNR and PZFC (bottom panel).

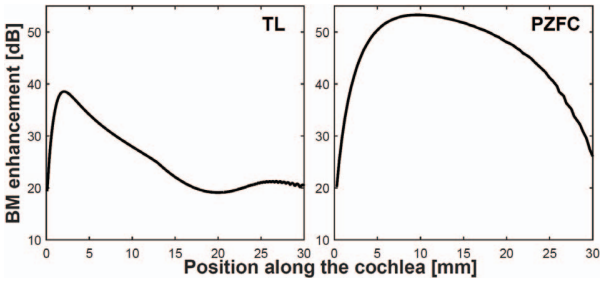


Figure 2: BM vibration enhancement provided at different positions along the fully active cochlea for both the TL model (left) and PZFC (right). Each channel of the PZFC is transformed to equivalent position along the cochlea according to its best frequency in the low-level limit using the Greenwood cochlear map for humans [21].

3. MODEL COMPARISONS

3.1. Stage Frequency Response

To examine the effect of ignoring the WNR in (3), we compute two sets of cascade filter frequency responses from the TL model, one with and one without the WNR, as shown in Fig. 1 (top panel) as thin and thick solid lines respectively. The 35 mm length of the cochlea has been uniformly divided into 85 sections, which is the number of channels in the PZFC and only the frequency responses of every third of the first 76 sections are shown in Fig. 1 for better visualization. It can be seen that the gain in the pass-band is significantly reduced, especially at basal sections, if the WNR is neglected. The bottom panel of Fig. 1 shows the section frequency responses of the cascade filters calculated from the PZFC, which are more similar to those of the TL model with WNR at low frequencies and those without the WNR at high frequencies. The error introduced by ignoring the WNR can, however, be largely compensated for by turning up the gain of various sections in the PZFC. Both models are set to be linear and fully active in these simulations, corresponding to the low-level limit.

Although the peak gain of each section is very similar between the two models, the decrease towards unity in the low frequency side of each stage frequency response is more gradual in the PZFC than in the TL model. The area of overlap in the passband region between different channels is therefore larger in the PZFC. This causes the cumulative gain at each section of the PZFC to be higher than that at the corresponding place along the BM of the TL model, except at the first a few stages, as shown in Fig. 2. The BM vibration enhancement along the cochlea is computed at each position as the difference in dB in the peak magnitude responses between the fully active and fully passive models. Each channel of the PZFC is converted to the equivalent position along the BM according to its best frequency in the low-level limit using the Greenwood map for humans [21]. Both models are in reasonable agreement with experimental observations [22], in that the active gain is greater at the base than at the apex. But the active gain decrease in the TL model is more rapid and the PZFC probably takes a too long section of the BM to fully build up the maximum gain, due to lower amplification provided by initial sections of the filter cascade.

3.2. Single Tone Stimulation

Fig. 3 shows the comparison of model responses to 2 kHz pure tone with levels ranging from 0 to 100 dB in steps of 10 dB. All stimuli had a duration of one second, with a 10 ms half-Hanning window onset ramp and the maximum amplitude of each section of each model during the whole length of the simulation time was taken for evaluation. Panels (a) and (b) show the simulated BM vibration patterns along the length of the cochlea in the TL model and PZFC, respectively. Both response patterns become broader and their maxima move basally as the stimulus level increases, showing qualitative agreement with measured longitudinal BM-velocity patterns in [23]. But the characteristic place (CP) for 2 kHz pure tone is different in the two models and the basal shift of response peak in the TL model appears at higher stimulus SPL (70 dB) than that (30 dB) in the PZFC.

Fig. 3 (c) and (d) show the derived BM input/output (I/O) level curves at the 2 kHz CP. In the TL model, the active mechanism offers an active gain of about 25 dB up to around 30 dB input SPL, and then compresses the response with a slope of 0.5 dB/dB until an input SPL of about 80 dB, after which the model behaves linearly again. The PZFC, however, provides an active gain of almost 50 dB at low stimulus level, and the compression starts to take effect at 10 dB SPL and does not finish until about 100 dB SPL, again with a growth rate of about 0.5 dB/dB inbetween. Both models show almost linear I/O functions at locations with characteristic frequencies (CFs) one octave above and below 2 kHz, i.e. about 4.5 mm to the left or right of the CP in Fig. 3 (a) and (b). Measurements of mammalian cochleae from different studies and across various species show a high degree of variation in terms of compression range and rate. Some experiment [24] measured compressive behavior at as low as 0 dB input SPL in the chinchilla, while others [25, 26] did not observe it until 50 dB SPL in the guinea pig. The BM response growth rate in the compressive region also varies from about 0.2 to 0.5 dB/dB [22]. Thus, although the level curves of two models are different, both of them are consistent with the range of experimental data.

3.3. Two-tone Suppression

Two-tone suppression on the level of the mechanical vibration of the BM has been demonstrated in a number of studies [27–31], but the

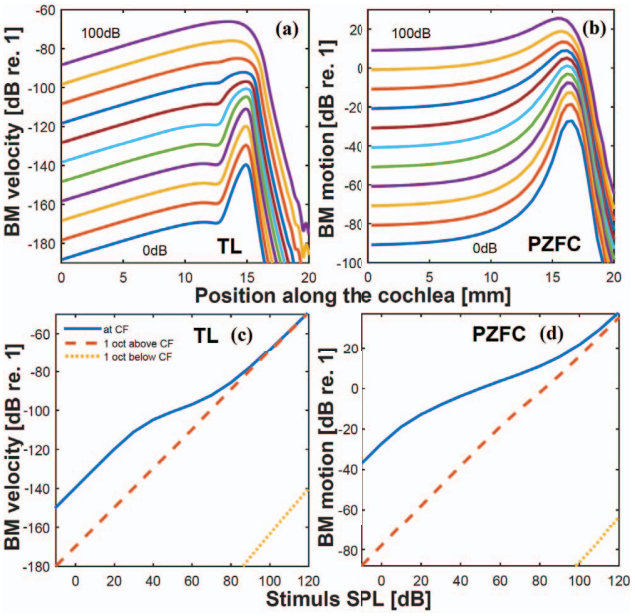


Figure 3: Comparison of model responses to 2 kHz pure tone with levels ranging from 0 dB to 100 dB SPL in steps of 10 dB. (a) Maximum BM velocity amplitudes of the TL model; (b) Maximum BM motion amplitudes of the PZFC. Derived input/output functions of BM vibrations at the 2 kHz CP and positions with CFs one octave above and below 2 kHz of the TL model (c) and PZFC (d).

measurements are not always consistent with each other. Ruggero *et al* [27] measured suppression to the overall velocity response of the BM with both high-side and low-side suppressors, while later experiments [29–31] only observed reductions in the probe frequency component. In this paper, we adopt the later definition which is also used in [29–31], defining the two-tone suppression as a reduction in the magnitude of the spectral component at the probe frequency during the presentation of a suppressor tone.

Fig. 4 shows the summary of model comparisons in two-tone suppression simulations. The stimulus also had a length of one second, but only the last 0.5 second of model outputs were used for Fourier analysis to reduce the transient effects. Panels (a) and (b) show the input/output level curves for a 2 kHz probe tone in the absence and presence of a 2.4 kHz suppressor tone at 5 different SPLs of the TL and PZFC models respectively. Both of them display reduced low-level linear and mid-range compressive regions of the probe tone growth curve as the intensity of the suppressor tone is increased. In contrast, for high probe tone levels, the level curves are almost linear and hardly affected by the suppressor. These results are in good qualitative agreement with those measured experimentally and shown in Fig. 2A of [30].

The level dependence of suppression to a 2 kHz probe tone, fixed at 45 dB SPL, on different suppressor frequencies in both models is then investigated and a small subset of results is displayed in panels (c) and (d) of Fig. 4. The degree of suppression was quantified as the ratio of the magnitude responses at the probe frequency in the presence and absence of the suppressor. Both models are capable of reproducing the general trend that low-side suppressors produce higher growth rates of suppression than high-side suppressors [29, 30], but the PZFC generates greater suppression at almost

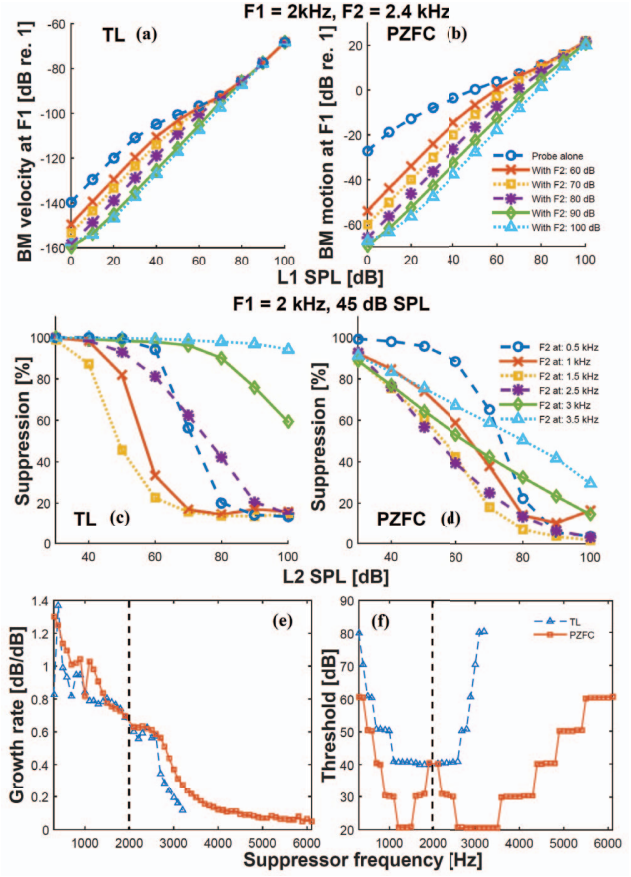


Figure 4: Comparison of model responses in two-tone suppression simulation. Input/output functions for a probe tone (F1=2 kHz) in the absence and presence of a suppressor tone (F2=2.4 kHz) at 5 different sound pressure levels of the TL model (a) and PZFC (b). Level dependence of two-tone suppression to a 2 kHz probe tone, at 45 dB SPL, on suppressor tone frequency in the TL model (c) and PZFC (d). Dependence of two-tone suppression growth rate (e) and threshold (f) on suppressor frequency in both models.

all levels of each suppressor and the suppression effect appears in it at lower suppressor level. In addition, in the TL model, the influence of high-side suppressors drops off considerably quicker with increasing separation of two tone frequencies, similar to Fig. 5A of [29], but the extent of suppression tends to reach a limit after a certain suppressor level. The PZFC continues to suppress the probe tone even at the highest suppressor level, closer to the real measurements [29, 30], which show no sign of suppression saturation. Differences in low- and high-side suppression have also been observed in another filter cascade cochlear model by Kates [32].

To quantify these differences between the two models, Fig. 4 (e) and (f) show the dependence of two-tone suppression growth rates and thresholds on suppressor frequency, computed from the complete simulation data set. Following the practice adopted in [29], suppression threshold is taken as the suppressor SPL which produces 10 % suppression of the probe tone component and suppression growth rate is determined as the steepest slope of the suppression growth curves shown in Fig. 4 (c) and (d) above the suppression threshold. Suppressors with thresholds higher than 100 dB SPL (e.g. F2 = 3.5 kHz in Fig. 4 (c)) are excluded from this analysis. The frequency of the probe tone is indicated by a dashed line in both panels. In the TL model, below-CF suppressors cause an average of ~0.85 dB of suppression for every dB increase in SPL and the above-CF suppressors cause a suppression growth rate of ~0.41 dB/dB. In the PZFC, the average suppression growth rates induced by low- and high-side suppressors are ~0.95 dB/dB and ~0.23 dB/dB, respectively. Both models are reasonably close to experimentally measured growth rate values, which are ~1 dB/dB for low-side and ~0.5 dB/dB for high-side suppressors [30], except that high-side suppression growth rate in the PZFC is too low. This is because the PZFC allows a significant wider range of above-CF suppressors to reduce the probe tone, the suppression growth rates of which are rather low as shown in Fig. 4 (e). Such difference can also be observed in Fig. 4 (f), which shows that suppressors with frequencies higher than about 3300 Hz cease to affect the probe tone in the TL model (i.e. suppression threshold higher than 100 dB), while in the PZFC, two-tone suppression is still measurable for suppressors with frequencies that are three times that of the probe. Furthermore, almost all of the suppression thresholds in the PZFC are smaller than those in the TL model, but the threshold pattern is better predicted by the TL model in comparison to physiological measurements, e.g. Fig. 6 of [29], which displays roughly constant values around the probe tone frequency.

4. CONCLUSIONS

In this paper, we have firstly derived the frequency response of each stage of the filter cascade from a linear TL model of the human cochlea. The error involved in ignoring the wavenumber ratio term in the original PZFC formulation can be largely compensated for by enhancing the amplification provided by each serial section, but the BM vibration enhancement at the basal sections is still smaller than that in the TL model.

The BM outputs of these two models in response to single-tone and two-tone stimuli have then been computed and compared with experimental measurements on laboratory animals, although the latter can be quite variable. Both models can account for the general features of these measurements, but differ in their response details. The TL model produces a reasonably realistic representation of cochlear mechanics in most simulations, but is more computationally intensive. The PZFC shares a similar performance in modeling self-suppression or dynamic-range-compression, but it overpredicts two-tone suppression due to high-side suppressor tones. Moreover, its two-tone suppression thresholds are too low and the threshold pattern across frequency is not consistent with experimental observations. But the major benefit of the PZFC is its computational efficiency and preliminary tests show that it is about 40 times faster than the TL model when running in the time domain. It would be possible to further adjust the parameters and structure of the PZFC, especially the AGC network, based on either physiological data from animals or psychoacoustical experiments with human subjects [2, 7], to allow better representation of these nonlinear phenomena that are important in peripheral auditory signal encoding and masking effects and hence enhance its suitability for either modeling cochlear mechanics or auditory based processing of acoustic signals.

5. REFERENCES

[1] E. De Boer, "Mechanics of the cochlea: modeling efforts," in *The cochlea*. Springer, 1996, pp. 258–317.

[2] R. F. Lyon, "Cascades of two-pole–two-zero asymmetric resonators are good models of peripheral auditory function," *The Journal of the Acoustical Society of America*, vol. 130, no. 6, pp. 3893–3904, 2011.

[3] R. F. Lyon, M. Rehn, S. Bengio, T. C. Walters, and G. Chechik, "Sound retrieval and ranking using sparse auditory representations," *Neural computation*, vol. 22, no. 9, pp. 2390–2416, 2010.

[4] T. C. Walters, D. A. Ross, and R. F. Lyon, "The intervalgram: An audio feature for large-scale cover-song recognition," in *From Sounds to Music and Emotions*. Springer, 2013, pp. 197–213.

[5] M. Cheatham and P. Dallos, "Two-tone suppression in inner hair cell responses," *Hearing research*, vol. 40, no. 3, pp. 187–196, 1989.

[6] B. Delgutte, "Two-tone rate suppression in auditory-nerve fibers: Dependence on suppressor frequency and level," *Hearing research*, vol. 49, no. 1, pp. 225–246, 1990.

[7] H. Duifhuis, "Level effects in psychophysical two-tone suppression," *The Journal of the Acoustical Society of America*, vol. 67, no. 3, pp. 914–927, 1980.

[8] L. J. Kanis and E. de Boer, "Self-suppression in a locally active nonlinear model of the cochlea: A quasilinear approach," *The Journal of the Acoustical Society of America*, vol. 94, no. 6, pp. 3199–3206, 1993.

[9] J. A. Young, S. J. Elliott, and B. Lineton, "Investigating the wave-fixed and place-fixed origins of the 2f1-f2 distortion product otoacoustic emission within a micromechanical cochlear model," *The Journal of the Acoustical Society of America*, vol. 131, no. 6, pp. 4699–4709, 2012.

[10] S. J. Elliott, E. M. Ku, and B. Lineton, "A state space model for cochlear mechanics," *The Journal of the Acoustical Society of America*, vol. 122, no. 5, pp. 2759–2771, 2007.

[11] J. A. How, S. J. Elliott, and B. Lineton, "The influence on predicted harmonic and distortion product generation of the position of the nonlinearity within cochlear micromechanical models," *The Journal of the Acoustical Society of America*, vol. 127, no. 2, pp. 652–655, 2010.

[12] M. Kringlebotn, "Network model for the human middle ear," *Scandinavian audiology*, vol. 17, no. 2, pp. 75–85, 1988.

[13] E. M. Ku, "Modelling the human cochlea," Ph.D. dissertation, University of Southampton, 2008.

[14] S. Pan, S. J. Elliott, P. D. Teal, and B. Lineton, "Efficient time-domain simulation of nonlinear, state-space, transmission-line models of the cochlea," *The Journal of the Acoustical Society of America*, vol. 137, no. 6, pp. 3559–3562, 2015.

[15] G. Zweig, R. Lipes, and J. Pierce, "The cochlear compromise," *The Journal of the Acoustical Society of America*, vol. 59, no. 4, pp. 975–982, 1976.

[16] J. M. Kates, "Accurate tuning curves in a cochlear model," *Speech and Audio Processing, IEEE Transactions on*, vol. 1, no. 4, pp. 453–462, 1993.

[17] R. F. Lyon, "Filter cascades as analogs of the cochlea," in *Neuromorphic systems engineering*. Springer, 1998, pp. 3–18.

[18] R. F. Lyon *et al.*, "Using a cascade of asymmetric resonators with fast-acting compression as a cochlear model for machine-hearing applications," in *Autumn Meeting of the Acoustical Society of Japan*, 2011, pp. 509–512.

[19] J. Allen, "A hair cell model of neural response," in *Mechanics of Hearing*. Springer, 1983, pp. 193–202.

[20] "Cascade of asymmetric resonators with fast-acting compression cochlear model," https://github.com/google/carfac, last accessed: April, 2015.

[21] D. D. Greenwood, "A cochlear frequency-position function for several species29 years later," *The Journal of the Acoustical Society of America*, vol. 87, no. 6, pp. 2592–2605, 1990.

[22] L. Robles and M. A. Ruggero, "Mechanics of the mammalian cochlea," *Physiological reviews*, vol. 81, no. 3, pp. 1305–1352, 2001.

[23] T. Ren, "Longitudinal pattern of basilar membrane vibration in the sensitive cochlea," *Proceedings of the National Academy of Sciences*, vol. 99, no. 26, pp. 17 101–17 106, 2002.

[24] W. S. Rhode and A. Recio, "Study of mechanical motions in the basal region of the chinchilla cochlea," *The Journal of the Acoustical Society of America*, vol. 107, no. 6, pp. 3317–3332, 2000.

[25] E. Murugasu and I. J. Russell, "The effect of efferent stimulation on basilar membrane displacement in the basal turn of the guinea pig cochlea," *The Journal of neuroscience*, vol. 16, no. 1, pp. 325–332, 1996.

[26] E. Murugasu and I. Russell, "Salicylate ototoxicity: the effects on basilar membrane displacement, cochlear microphonics, and neural responses in the basal turn of the guinea pig cochlea," *Aud. Neurosci*, vol. 1, pp. 139–150, 1995.

[27] M. A. Ruggero, L. Robles, and N. C. Rich, "Two-tone suppression in the basilar membrane of the cochlea: Mechanical basis of auditory-nerve rate suppression," *Journal of neurophysiology*, vol. 68, no. 4, pp. 1087–1099, 1992.

[28] A. Nuttall and D. Dolan, "Two-tone suppression of inner hair cell and basilar membrane responses in the guinea pig," *The Journal of the Acoustical Society of America*, vol. 93, no. 1, pp. 390–400, 1993.

[29] N. Cooper and W. Rhode, "Two-tone suppression in apical cochlear mechanics," *Auditory Neuroscience*, vol. 3, no. 2, pp. 123–134, 1996.

[30] N. P. Cooper, "Two-tone suppression in cochlear mechanics," *The Journal of the Acoustical Society of America*, vol. 99, no. 5, pp. 3087–3098, 1996.

[31] C. D. Geisler and A. L. Nuttall, "Two-tone suppression of basilar membrane vibrations in the base of the guinea pig cochlea using low-side suppressors," *The Journal of the Acoustical Society of America*, vol. 102, no. 1, pp. 430–440, 1997.

[32] J. M. Kates, "Two-tone suppression in a cochlear model," *Speech and Audio Processing, IEEE Transactions on*, vol. 3, no. 5, pp. 396–406, 1995.

References

[1] Stephen J. Elliott, Emery M. Ku, and Ben Lineton. "A state space model for cochlear mechanics". In: *The Journal of the Acoustical Society of America* 122.5 (2007), pp. 2759–2771.

[2] DeLiang Wang and Guy J. Brown. *Computational auditory scene analysis: Principles, algorithms, and applications*. Wiley-IEEE press, 2006.

[3] Amin Saremi et al. "A comparative study of seven human cochlear filter models". In: *The Journal of the Acoustical Society of America* 140.3 (2016), pp. 1618–1634.

[4] Richard M. Stern. "Applying physiologically-motivated models of auditory processing to automatic speech recognition". In: *International Symposium on Auditory and Audiological Research*. 2011.

[5] Yuxuan Wang. *Supervised speech separation using deep neural networks*. The Ohio State University, 2015.

[6] Xiao-Lei Zhang and DeLiang Wang. "Boosting contextual information for deep neural network based voice activity detection". In: *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 24.2 (2016), pp. 252–264.

[7] Manfred R. Schroeder, Bishnu S. Atal, and J. L. Hall. "Optimizing digital speech coders by exploiting masking properties of the human ear". In: *The Journal of the Acoustical Society of America* 66.6 (1979), pp. 1647–1652.

[8] M. Schroeder and B. S. Atal. "Code-excited linear prediction (CELP): High-quality speech at very low bit rates". In: *IEEE International Conference on Acoustics, Speech, and Signal Processing*. Vol. 10. IEEE, 1985, pp. 937–940.

[9] Stephen T. Neely and D. O. Kim. "A model for active elements in cochlear biomechanics". In: *The Journal of the Acoustical Society of America* 79.5 (1986), pp. 1472–1480.

[10] Emery M. Ku. "Modelling the human cochlea". PhD thesis. University of Southampton, 2008.

[11] Jacqueline Ann Young. "Modelling the cochlear origins of distortion product otoacoustic emissions". PhD thesis. University of Southampton, 2011.

[12] Richard F. Lyon. *Human and Machine Hearing: Extracting Meaning from Sound*. Cambridge University Press, 2017.

[13] Nathalie Virag. "Single channel speech enhancement based on masking properties of the human auditory system". In: *IEEE Transactions on Speech and Audio Processing* 7.2 (1999), pp. 126–137.

[14] Stanley S. Stevens and John Volkmann. "The relation of pitch to frequency: A revised scale". In: *The American Journal of Psychology* 53.3 (1940), pp. 329–353.

[15] Hynek Hermansky. "Perceptual linear predictive (PLP) analysis of speech". In: *the Journal of the Acoustical Society of America* 87.4 (1990), pp. 1738–1752.

[16] R. D. Patterson et al. "An efficient auditory filterbank based on the gammatone function". In: *a meeting of the IOC Speech Group on Auditory Modelling at RSRE*. Vol. 2. 7. 1987.

[17] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. "Deep learning". In: *Nature* 521.7553 (2015), pp. 436–444.

[18] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. `http://www.deeplearningbook.org`. MIT Press, 2016.

[19] *CS231n: Convolutional Neural Networks for Visual Recognition*. URL: `http://cs231n.stanford.edu/`.

[20] Dong Yu and Li Deng. *Automatic speech recognition: A deep learning approach*. Springer, 2014.

[21] Stanley A. Gelfand. *Hearing: An introduction to psychological and physiological acoustics*. CRC Press, 2016.

[22] *Anatomy of the Human Ear*. Nov. 2017. URL: `https://commons.wikimedia.org/wiki/File:Anatomy_of_the_Human_Ear_en.svg`.

[23] E. M. Relkin. "Introduction to the analysis of middle-ear function". In: *Physiology of the Ear (AF Jahn and J. Santos-Sacchi, eds.). Raven Press, New York* (1988), pp. 103–123.

[24] Jerome Pascal et al. "Linear and nonlinear model of the human middle ear". In: *The Journal of the Acoustical Society of America* 104.3 (1998), pp. 1509–1516.

[25] Peter Dallos and Richard R. Fay. *The cochlea*. Vol. 8. Springer-Verlag New York, 1996.

[26] A. Wright et al. "Hair cell distributions in the normal human cochlea." In: *Acta oto-laryngologica. Supplementum* 444 (1987), pp. 1–48.

[27] Hans Engström and Jan Wersäll. "Structure and innervation of the inner ear sensory epithelia". In: *International Review of Cytology* 7 (1958), pp. 535–585.

[28] Eric R. Kandel et al. *Principles of neural science*. Vol. 4. McGraw-hill New York, 2000.

[29] *Tempel lab, Courses, Lectures and Handouts*. Nov. 2017. URL: `https://depts.washington.edu/tempelab/index.html`.

[30] William E. Brownell et al. "Evoked mechanical responses of isolated cochlear outer hair cells". In: *Science* 227 (1985), pp. 194–197.

[31] Donald D. Greenwood. "A cochlear frequency-position function for several species—29 years later". In: *The Journal of the Acoustical Society of America* 87.6 (1990), pp. 2592–2605.

[32] *Cochlea anatomy*. Nov. 2017. URL: `https://www.britannica.com/science/cochlea`.

[33] Luis Robles and Mario A. Ruggero. "Mechanics of the mammalian cochlea". In: *Physiological Reviews* 81.3 (2001), pp. 1305–1352.

[34] J. Ashmore et al. "The remarkable cochlear amplifier". In: *Hearing Research* 266.1 (2010), pp. 1–17.

[35] Luc J. Kanis and Egbert de Boer. "Self-suppression in a locally active nonlinear model of the cochlea: A quasilinear approach". In: *The Journal of the Acoustical Society of America* 94.6 (1993), pp. 3199–3206.

[36] Sid P. Bacon, Richard R. Fay, and Arthur N. Popper. *Compression: from cochlea to cochlear implants*. Springer, 2004.

[37] B. M. Johnstone, R. Patuzzi, and G. K. Yates. "Basilar membrane measurements and the travelling wave". In: *Hearing Research* 22.1 (1986), pp. 147–153.

[38] Mario A. Ruggero et al. "Basilar-membrane responses to tones at the base of the chinchilla cochlea". In: *The Journal of the Acoustical Society of America* 101.4 (1997), pp. 2151–2163.

[39] Bertrand Delgutte. "Two-tone rate suppression in auditory-nerve fibers: Dependence on suppressor frequency and level". In: *Hearing Research* 49.1 (1990), pp. 225–246.

[40] Mario A. Ruggero, Luis Robles, and Nola C. Rich. "Two-tone suppression in the basilar membrane of the cochlea: Mechanical basis of auditory-nerve rate suppression". In: *Journal of Neurophysiology* 68.4 (1992), pp. 1087–1099.

[41] N. P. Cooper and W. S. Rhode. "Two-tone suppression in apical cochlear mechanics". In: *Auditory Neuroscience* 3.2 (1996), pp. 123–134.

[42] C. Daniel Geisler and Alfred L. Nuttall. "Two-tone suppression of basilar membrane vibrations in the base of the guinea pig cochlea using "low-side" suppressors". In: *The Journal of the Acoustical Society of America* 102.1 (1997), pp. 430–440.

[43] A. L. Nuttall and D. F. Dolan. "Two-tone suppression of inner hair cell and basilar membrane responses in the guinea pig". In: *The Journal of the Acoustical Society of America* 93.1 (1993), pp. 390–400.

[44] Hendrikus Duifhuis. "Level effects in psychophysical two-tone suppression". In: *The Journal of the Acoustical Society of America* 67.3 (1980), pp. 914–927.

[45] William S. Rhode. "Mutual suppression in the 6kHz region of sensitive chinchilla cochleae". In: *The Journal of the Acoustical Society of America* 121.5 (2007), pp. 2805–2818.

[46] Mohamed Farouk Abdel Hady and Friedhelm Schwenker. "Semi-supervised learning". In: *Handbook on Neural Information Processing*. Springer, 2013, pp. 215–239.

[47] Sinno Jialin Pan and Qiang Yang. "A survey on transfer learning". In: *IEEE Transactions on Knowledge and Data Engineering* 22.10 (2010), pp. 1345–1359.

[48] Dimitri P. Bertsekas and John N. Tsitsiklis. "Neuro-dynamic programming: an overview". In: *IEEE Conference on Decision and Control*. Vol. 1. IEEE. 1995, pp. 560–564.

[49] Christopher M. Bishop. *Pattern recognition and machine learning*. Springer, 2006.

[50] Sepp Hochreiter et al. *Gradient flow in recurrent nets: the difficulty of learning long-term dependencies*. 2001.

[51] Xavier Glorot, Antoine Bordes, and Yoshua Bengio. "Deep sparse rectifier neural networks". In: *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*. 2011, pp. 315–323.

[52] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. "Imagenet classification with deep convolutional neural networks". In: *Advances in Neural Information Processing Systems*. 2012, pp. 1097–1105.

[53] Kaiming He et al. "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification". In: *Proceedings of the IEEE International Conference on Computer Vision*. 2015, pp. 1026–1034.

[54] Ossama Abdel-Hamid et al. "Convolutional neural networks for speech recognition". In: *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 22.10 (2014), pp. 1533–1545.

[55] David E. Rumelhart, Geoffrey E. Hinton, Ronald J. Williams, et al. "Learning representations by back-propagating errors". In: *Cognitive modeling* 5.3 (1988), p. 1.

[56] Michael A. Nielsen. *Neural networks and deep learning*. `http://neuralnetworksanddeeplearning.com/chap2.html`. Determination Press, 2015.

[57] Boris T. Polyak. "Some methods of speeding up the convergence of iteration methods". In: *USSR Computational Mathematics and Mathematical Physics* 4.5 (1964), pp. 1–17.

[58] John Duchi, Elad Hazan, and Yoram Singer. "Adaptive subgradient methods for online learning and stochastic optimization". In: *Journal of Machine Learning Research* 12.Jul (2011), pp. 2121–2159.

[59] Geoffrey Hinton. *Neural Networks for Machine Learning. Coursera, video lectures*. Nov. 2017. URL: `https://www.coursera.org/learn/neural-networks`.

[60] Matthew D. Zeiler. "ADADELTA: an adaptive learning rate method". In: *arXiv preprint arXiv:1212.5701* (2012).

[61] Diederik Kingma and Jimmy Ba. "Adam: A method for stochastic optimization". In: *arXiv preprint arXiv:1412.6980* (2014).

[62] Geoffrey E. Hinton, Simon Osindero, and Yee-Whye Teh. "A fast learning algorithm for deep belief nets". In: *Neural Computation* 18.7 (2006), pp. 1527–1554.

[63] Nitish Srivastava et al. "Dropout: a simple way to prevent neural networks from overfitting." In: *Journal of Machine Learning Research* 15.1 (2014), pp. 1929–1958.

[64] Sripriya Ramamoorthy, Niranjan V. Deo, and Karl Grosh. "A mechano-electro-acoustical model for the cochlea: response to acoustic stimuli". In: *The Journal of the Acoustical Society of America* 121.5 (2007), pp. 2758–2773.

[65] Renata Sisto et al. "Different models of the active cochlea, and how to implement them in the state-space formalism". In: *The Journal of the Acoustical Society of America* 128.3 (2010), pp. 1191–1202.

[66] Julien Meaud and Karl Grosh. "Response to a pure tone in a nonlinear mechanical-electrical-acoustical model of the cochlea". In: *Biophysical Journal* 102.6 (2012), pp. 1237–1246.

[67] Julius L. Goldstein. "Modeling rapid waveform compression on the basilar membrane as multiple-bandpass-nonlinearity filtering". In: *Hearing research* 49.1 (1990), pp. 39–60.

[68] Enrique A Lopez-Poveda and Ray Meddis. "A human nonlinear cochlear filterbank". In: *The Journal of the Acoustical Society of America* 110.6 (2001), pp. 3107–3118.

[69] Xuedong Zhang et al. "A phenomenological model for the responses of auditory-nerve fibers: I. Nonlinear tuning with compression and suppression". In: *The Journal of the Acoustical Society of America* 109.2 (2001), pp. 648–670.

[70] Luc-J. Kanis and Egbert De Boer. "Two-tone suppression in a locally active nonlinear model of the cochlea". In: *The Journal of the Acoustical Society of America* 96.4 (1994), pp. 2156–2165.

[71] Julius L. Goldstein. "Relations among compression, suppression, and combination tones in mechanical responses of the basilar membrane: data and MBPNL model". In: *Hearing Research* 89.1 (1995), pp. 52–68.

[72] R. Stoop and A. Kern. "Two-tone suppression and combination tone generation as computations performed by the Hopf cochlea". In: *Physical Review Letters* 93.26 (2004), p. 268103.

[73] James M. Kates. "Two-tone suppression in a cochlear model". In: *IEEE Transactions on Speech and Audio Processing* 3.5 (1995), pp. 396–406.

[74] Guangjian Ni et al. "Modelling cochlear mechanics". In: *BioMed research international* 2014 (2014).

[75] Hendrikus Duifhuis. "Comment on "An approximate transfer function for the dual-resonance nonlinear filter model of auditory frequency selectivity"[J. Acoust. Soc. Am. 114, 2112–2117](L)". In: *The Journal of the Acoustical Society of America* 115.5 (2004), pp. 1889–1890.

[76] Robert Duncan Luce, Robert R. Bush, and Galanter Ed Eugene. "Handbook of Mathematical Psychology." In: (1963).

[77] Egbert De Boer. "Mechanics of the cochlea: modeling efforts". In: *The cochlea*. Springer, 1996, pp. 258–317.

[78] Toshio Irino and Roy D. Patterson. "A time-domain, level-dependent auditory filter: The gammachirp". In: *The Journal of the Acoustical Society of America* 101.1 (1997), pp. 412–419.

[79] Richard F. Lyon. "Filter cascades as analogs of the cochlea". In: *Neuromorphic systems engineering*. Springer, 1998, pp. 3–18.

[80] Richard F. Lyon. "Cascades of two-pole–two-zero asymmetric resonators are good models of peripheral auditory function". In: *The Journal of the Acoustical Society of America* 130.6 (2011), pp. 3893–3904.

[81] Richard F. Lyon et al. "Using a Cascade of Asymmetric Resonators with Fast-Acting Compression as a Cochlear Model for Machine-Hearing Applications". In: *Autumn Meeting of the Acoustical Society of Japan*. 2011, pp. 509–512.

[82] Christopher A. Shera and John J. Guinan Jr. "Evoked otoacoustic emissions arise by two fundamentally different mechanisms: a taxonomy for mammalian OAEs". In: *The Journal of the Acoustical Society of America* 105.2 (1999), pp. 782–798.

[83] Emery M. Ku, Stephen J. Elliott, and Ben Lineton. "Limit cycle oscillations in a nonlinear state space model of the human cochlea". In: *The Journal of the Acoustical Society of America* 126.2 (2009), pp. 739–750.

[84] Egbert De Boer and Alfred L. Nuttall. "The mechanical waveform of the basilar membrane. I. Frequency modulations ("glides") in impulse responses and cross-correlation functions". In: *The Journal of the Acoustical Society of America* 101.6 (1997), pp. 3583–3592.

[85] Richard F. Lyon et al. "Sound retrieval and ranking using sparse auditory representations". In: *Neural Computation* 22.9 (2010), pp. 2390–2416.

[86] Thomas C. Walters, David A. Ross, and Richard F. Lyon. "The Intervalgram: An Audio Feature for Large-Scale Cover-Song Recognition". In: *From Sounds to Music and Emotions*. Springer, 2013, pp. 197–213.

[87] Jacqueline A. Young, Stephen J. Elliott, and Ben Lineton. "Investigating the wave-fixed and place-fixed origins of the 2f1-f2 distortion product otoacoustic emission within a micromechanical cochlear model". In: *The Journal of the Acoustical Society of America* 131.6 (2012), pp. 4699–4709.

[88] Stephen J. Elliott and Christopher A. Shera. "The cochlea as a smart structure". In: *Smart Materials and Structures* 21.6 (2012), p. 064001.

[89] Emery M. Ku, Stephen J. Elliott, and Ben Lineton. "Statistics of instabilities in a state space model of the human cochlea". In: *The Journal of the Acoustical Society of America* 124.2 (2008), pp. 1068–1079.

[90] Jacqueline A. How, Stephen J. Elliott, and Ben Lineton. "The influence on predicted harmonic and distortion product generation of the position of the nonlinearity within cochlear micromechanical models". In: *The Journal of the Acoustical Society of America* 127.2 (2010), pp. 652–655.

[91] Peter Dallos. "Overview: cochlear neurobiology". In: *The cochlea*. Springer, 1996, pp. 1–43.

[92] Corné J. Kros. "Physiology of mammalian cochlear hair cells". In: *The cochlea*. Springer, 1996, pp. 318–385.

[93] Stuart L. Johnson et al. "Prestin-driven cochlear amplification is not limited by the outer hair cell membrane time constant". In: *Neuron* 70.6 (2011), pp. 1143–1154.

[94] M. Kringlebotn. "Network model for the human middle ear". In: *Scandinavian Audiology* 17.2 (1988), pp. 75–85.

[95] Shuokai Pan et al. "Efficient time-domain simulation of nonlinear, state-space, transmission-line models of the cochlea". In: *The Journal of the Acoustical Society of America* 137.6 (2015), pp. 3559–3562.

[96] James M. Kates. "A time-domain digital cochlear model". In: *IEEE Transactions on signal processing* 39.12 (1991), pp. 2573–2592.

[97] M. Holmes and J. D. Cole. "Pseudo-resonance in the cochlea". In: *Mechanics of Hearing*. Springer, 1983, pp. 45–52.

[98] *Cascade of asymmetric resonators with fast-acting compression cochlear model*. URL: `https://github.com/google/carfac`.

[99] J. B. Allen. "A hair cell model of neural response". In: *Mechanics of Hearing*. Springer, 1983, pp. 193–202.

[100] G. von Bekesy. *Sensory inhibition*. Princeton University Press, 1967.

[101] George Zweig, R. Lipes, and J. R. Pierce. "The cochlear compromise". In: *The Journal of the Acoustical Society of America* 59.4 (1976), pp. 975–982.

[102] William S. Rhode and Alberto Recio. "Study of mechanical motions in the basal region of the chinchilla cochlea". In: *The Journal of the Acoustical Society of America* 107.6 (2000), pp. 3317–3332.

[103] Euan Murugasu and Ian J. Russell. "The effect of efferent stimulation on basilar membrane displacement in the basal turn of the guinea pig cochlea". In: *The Journal of neuroscience* 16.1 (1996), pp. 325–332.

[104] E. Murugasu and I. J. Russell. "Salicylate ototoxicity: the effects on basilar membrane displacement, cochlear microphonics, and neural responses in the basal turn of the guinea pig cochlea". In: *Auditory Neuroscience* 1 (1995), pp. 139–150.

[105] Nigel P. Cooper. "Two-tone suppression in cochlear mechanics". In: *The Journal of the Acoustical Society of America* 99.5 (1996), pp. 3087–3098.

[106] William S. Rhode and Alberto Recio. "Multicomponent stimulus interactions observed in basilar-membrane vibration in the basal region of the chinchilla cochlea". In: *The Journal of the Acoustical Society of America* 110.6 (2001), pp. 3140–3154.

[107] Corstiaen P. C. Versteegh and Marcel van der Heijden. "The spatial buildup of compression and suppression in the mammalian cochlea". In: *Journal of the Association for Research in Otolaryngology* 14.4 (2013), pp. 523–545.

[108] Javier Ramirez, Juan Manuel Gorriz, and Jose Carlos Segura. "Voice activity detection. fundamentals and speech recognition system robustness". In: *Robust Speech Recognition and Understanding*. InTech, 2007.

[109] Lawrence R. Rabiner and Marvin R. Sambur. "An algorithm for determining the endpoints of isolated utterances". In: *Bell Labs Technical Journal* 54.2 (1975), pp. 297–315.

[110] Thomas Drugman et al. "Voice activity detection: merging source and filter-based information". In: *IEEE Signal Processing Letters* 23.2 (2016), pp. 252–256.

[111] Thomas Drugman et al. "Glottal source processing: From analysis to applications". In: *Computer Speech & Language* 28.5 (2014), pp. 1117–1138.

[112] Brian Kingsbury et al. "Robust speech recognition in noisy environments: The 2001 IBM SPINE evaluation system". In: *Acoustics, Speech, and Signal Processing (ICASSP), 2002 IEEE International Conference on*. Vol. 1. IEEE. 2002, pp. I–53.

[113] Sumit Basu. "A linked-HMM model for robust voicing and speech detection". In: *Acoustics, Speech, and Signal Processing, 2003. Proceedings.(ICASSP'03). 2003 IEEE International Conference on*. Vol. 1. IEEE. 2003, pp. I–I.

[114] Sassan Ahmadi and Andreas S. Spanias. "Cepstrum-based pitch detection using a new statistical V/UV classification algorithm". In: *IEEE Transactions on speech and audio processing* 7.3 (1999), pp. 333–338.

[115] Simon Graf et al. "Features for voice activity detection: a comparative analysis". In: *EURASIP Journal on Advances in Signal Processing* 2015.1 (2015), p. 91.

[116] Adil Benyassine et al. "ITU-T Recommendation G. 729 Annex B: a silence compression scheme for use with G. 729 optimized for V. 70 digital simultaneous voice and data applications". In: *IEEE Communications Magazine* 35.9 (1997), pp. 64–73.

[117] Steven Davis and Paul Mermelstein. "Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences". In: *IEEE Transactions on Acoustics, Speech, and Signal Processing* 28.4 (1980), pp. 357–366.

[118] Chanwoo Kim and Richard M. Stern. "Power-normalized cepstral coefficients (PNCC) for robust speech recognition". In: *IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP)* 24.7 (2016), pp. 1315–1329.

[119] Jitong Chen, Yuxuan Wang, and DeLiang Wang. "A feature study for classification-based speech separation at low signal-to-noise ratios". In: *IEEE/ACM Transactions on Audio, Speech and Language Processing* 22.12 (2014), pp. 1993–2002.

[120] Xiao-Lei Zhang and DeLiang Wang. "Boosted deep neural networks and multi-resolution cochleagram features for voice activity detection". In: *Fifteenth Annual Conference of the International Speech Communication Association*. 2014.

[121] Pham Chau Khoa. "Noise robust voice activity detection". In: *Master thesis, Nanyang Technological University* 24 (2012).

[122] Javier Ramırez et al. "Efficient voice activity detection algorithms using long-term speech information". In: *Speech Communication* 42.3-4 (2004), pp. 271–287.

[123] Georgios Evangelopoulos and Petros Maragos. "Speech event detection using multiband modulation energy". In: *Ninth European Conference on Speech Communication and Technology*. 2005.

[124] Yariv Ephraim and David Malah. "Speech enhancement using a minimum-mean square error short-time spectral amplitude estimator". In: *IEEE Transactions on Acoustics, Speech, and Signal Processing* 32.6 (1984), pp. 1109–1121.

[125] Rainer Martin. "Noise power spectral density estimation based on optimal smoothing and minimum statistics". In: *IEEE Transactions on Speech and Audio Processing* 9.5 (2001), pp. 504–512.

[126] Israel Cohen. "Noise spectrum estimation in adverse environments: Improved minima controlled recursive averaging". In: *IEEE Transactions on Speech and Audio Processing* 11.5 (2003), pp. 466–475.

[127] Richard C Hendriks, Richard Heusdens, and Jesper Jensen. "MMSE based noise PSD tracking with low complexity". In: *IEEE International Conference on Acoustics Speech and Signal Processing*. IEEE. 2010, pp. 4266–4269.

[128] Timo Gerkmann and Richard C. Hendriks. "Unbiased MMSE-based noise power estimation with low complexity and low tracking delay". In: *IEEE Transactions on Audio, Speech, and Language Processing* 20.4 (2012), pp. 1383–1393.

[129] Rainer Martin. "Speech enhancement using MMSE short time spectral estimation with gamma distributed speech priors". In: *IEEE International Conference on Acoustics, Speech, and Signal Processing*. Vol. 1. IEEE. 2002, pp. I–253.

[130] J. H. Chang, J. W. Shin, and N. S. Kim. "Voice activity detector employing generalised Gaussian distribution". In: *Electronics Letters* 40.24 (2004), pp. 1561–1563.

[131] Joon-Hyuk Chang, Nam Soo Kim, and Sanjit K. Mitra. "Voice activity detection based on multiple statistical models". In: *IEEE Transactions on Signal Processing* 54.6 (2006), pp. 1965–1976.

[132] R. Reininger and J. Gibson. "Distributions of the two-dimensional DCT coefficients for images". In: *IEEE Transactions on Communications* 31.6 (1983), pp. 835–839.

[133] Theodoros Petsatodis et al. "Convex combination of multiple statistical models with application to VAD". In: *IEEE Transactions on Audio, Speech, and Language Processing* 19.8 (2011), pp. 2314–2327.

[134] Xiao-Lei Zhang and Ji Wu. "Deep belief networks based voice activity detection". In: *IEEE Transactions on Audio, Speech, and Language Processing* 21.4 (2013), pp. 697–710.

[135] Speech Processing. "Speech processing, transmission and quality Aspects (STQ); Distributed speech recognition; Advanced front-end feature extraction algorithm; Compression algorithms". In: *ETSI ES* 202.050 (2002), p. V1.

[136] Jongseo Sohn, Nam Soo Kim, and Wonyong Sung. "A statistical model-based voice activity detection". In: *IEEE Signal Processing Letters* 6.1 (1999), pp. 1–3.

[137] Javier Ramirez et al. "Statistical voice activity detection using a multiple observation likelihood ratio test". In: *IEEE Signal Processing Letters* 12.10 (2005), pp. 689–692.

[138] Javier Ramirez et al. "Improved voice activity detection using contextual multiple hypothesis testing for robust speech recognition". In: *IEEE Transactions on Audio, Speech, and Language Processing* 15.8 (2007), pp. 2177–2189.

[139] Ji Wu and Xiao-Lei Zhang. "Efficient multiple kernel support vector machine based voice activity detection". In: *IEEE Signal Processing Letters* 18.8 (2011), pp. 466–469.

[140] Dongwen Ying et al. "Voice activity detection based on an unsupervised learning framework". In: *IEEE Transactions on Audio, Speech, and Language Processing* 19.8 (2011), pp. 2624–2633.

[141] Jeff Ma. "Improving the speech activity detection for the DARPA RATS phase-3 evaluation". In: *Fifteenth Annual Conference of the International Speech Communication Association*. 2014.

[142] Samuel Thomas et al. "Improvements to the IBM speech activity detection system for the DARPA RATS program". In: *IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE. 2015, pp. 4500–4504.

[143] John S Garofolo et al. "DARPA TIMIT acoustic-phonetic continous speech corpus CD-ROM. NIST speech disc 1-1.1". In: *NASA STI/Recon technical report n* 93 (1993).

[144] Andrew Varga and Herman J. M. Steeneken. "Assessment for automatic speech recognition: II. NOISEX-92: A database and an experiment to study the effect of additive noise on speech recognition systems". In: *Speech communication* 12.3 (1993), pp. 247–251.

[145] David B. Dean et al. "The QUT-NOISE-TIMIT corpus for the evaluation of voice activity detection algorithms". In: *Proceedings of Interspeech 2010* (2010).

[146] Philipos C Loizou. *Speech enhancement: theory and practice*. CRC press, 2013.

[147] Xavier Glorot and Yoshua Bengio. "Understanding the difficulty of training deep feedforward neural networks". In: *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*. 2010, pp. 249–256.

[148] François Chollet et al. *Keras*. `https://github.com/fchollet/keras`. 2015.

[149] Theano Development Team. "Theano: A Python framework for fast computation of mathematical expressions". In: *arXiv e-prints* abs/1605.02688 (May 2016). URL: `http://arxiv.org/abs/1605.02688`.

[150] Maarten Van Segbroeck, Andreas Tsiartas, and Shrikanth Narayanan. "A robust frontend for VAD: exploiting contextual, discriminative and spectral cues of human voice." In: *Interspeech*. 2013, pp. 704–708.

[151] Yang Shao, Soundararajan Srinivasan, and DeLiang Wang. "Incorporating auditory feature uncertainties in robust speaker identification". In: *IEEE International Conference on Acoustics, Speech and Signal Processing*. Vol. 4. IEEE. 2007, pp. IV–277.

[152] Bernd T. Meyer et al. "Comparing different flavors of spectro-temporal features for ASR". In: *Twelfth Annual Conference of the International Speech Communication Association*. 2011.

[153] Svante Granqvist and Britta Hammarberg. "The correlogram: a visual display of periodicity". In: *The Journal of the Acoustical Society of America* 114.5 (2003), pp. 2934–2945.

[154] Prasanta Kumar Ghosh, Andreas Tsiartas, and Shrikanth Narayanan. "Robust voice activity detection using long-term signal variability". In: *IEEE Transactions on Audio, Speech, and Language Processing* 19.3 (2011), pp. 600–613.

[155] Hynek Hermansky and Sangita Sharma. "TRAPS-classifiers of temporal patterns." In: *ICSLP*. 1998, pp. 1003–1006.

[156] Thomas Drugman et al. "Voice activity detection: merging source and filter-based information". In: *IEEE Signal Processing Letters* 23.2 (2016), pp. 252–256.

[157] Gilles Degottex et al. "COVAREP—A collaborative voice analysis repository for speech technologies". In: *IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE. 2014, pp. 960–964.

[158] Baris Bozkurt, Laurent Couvreur, and Thierry Dutoit. "Chirp group delay analysis of speech signals". In: *Speech Communication* 49.3 (2007), pp. 159–176.

[159] Seyed Omid Sadjadi and John HL Hansen. "Unsupervised speech activity detection using voicing measures and perceptual spectral flux". In: *IEEE Signal Processing Letters* 20.3 (2013), pp. 197–200.

[160] James Hillenbrand and Robert A. Houde. "Acoustic correlates of breathy vocal quality: dysphonic voices and continuous speech". In: *Journal of Speech, Language, and Hearing Research* 39.2 (1996), pp. 311–321.

[161] Thomas Drugman and Abeer Alwan. "Joint robust voicing detection and pitch estimation based on residual harmonics". In: *Twelfth Annual Conference of the International Speech Communication Association*. 2011.

[162] C. l. Doire et al. "Single-channel enhancement of speech corrupted by reverberation and noise". PhD thesis. 2017. URL: `http://hdl.handle.net/10044/1/43932`.

[163] Kuldip K. Paliwal and Leigh D. Alsteris. "On the usefulness of STFT phase spectrum in human listening tests". In: *Speech Communication* 45.2 (2005), pp. 153–170.

[164] Michael Berouti, Richard Schwartz, and John Makhoul. "Enhancement of speech corrupted by acoustic noise". In: *IEEE International Conference on Acoustics, Speech, and Signal Processing*. Vol. 4. IEEE. 1979, pp. 208–211.

[165] Steven Boll. "Suppression of acoustic noise in speech using spectral subtraction". In: *IEEE Transactions on Acoustics, Speech, and Signal Processing* 27.2 (1979), pp. 113–120.

[166] Philip Lockwood and Jerome Boudy. "Experiments with a nonlinear spectral subtractor (NSS), hidden Markov models and the projection, for robust speech recognition in cars". In: *Speech Communication* 11.2-3 (1992), pp. 215–228.

[167] Harald Gustafsson, Sven E. Nordholm, and Ingvar Claesson. "Spectral subtraction using reduced delay convolution and adaptive averaging". In: *IEEE Transactions on Speech and Audio Processing* 9.8 (2001), pp. 799–807.

[168] Israel Cohen. "Speech spectral modeling and enhancement based on autoregressive conditional heteroscedasticity models". In: *Signal Processing* 86.4 (2006), pp. 698–709.

[169] Yariv Ephraim and David Malah. "Speech enhancement using a minimum mean-square error log-spectral amplitude estimator". In: *IEEE Transactions on Acoustics, Speech, and Signal Processing* 33.2 (1985), pp. 443–445.

[170] Timo Gerkmann and Richard C. Hendriks. "Unbiased MMSE-based noise power estimation with low complexity and low tracking delay". In: *IEEE Transactions on Audio, Speech, and Language Processing* 20.4 (2012), pp. 1383–1393.

[171] Seyedmahdad Mirsamadi and Ivan Tashev. "Causal Speech Enhancement Combining Data-Driven Learning and Suppression Rule Estimation." In: *Interspeech*. 2016, pp. 2870–2874.

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

[173] Michael C. Anzalone et al. "Determination of the potential benefit of time-frequency gain manipulation". In: *Ear and Hearing* 27.5 (2006), p. 480.

[174] Douglas S. Brungart et al. "Isolating the energetic component of speech-on-speech masking with ideal time-frequency segregation". In: *The Journal of the Acoustical Society of America* 120.6 (2006), pp. 4007–4018.

[175] Ning Li and Philipos C. Loizou. "Factors influencing intelligibility of ideal binary-masked speech: Implications for noise reduction". In: *The Journal of the Acoustical Society of America* 123.3 (2008), pp. 1673–1682.

[176] Yuxuan Wang, Arun Narayanan, and DeLiang Wang. "On training targets for supervised speech separation". In: *IEEE/ACM Transactions on Audio, Speech and Language Processing* 22.12 (2014), pp. 1849–1858.

[177] Xiao-Lei Zhang and DeLiang Wang. "A deep ensemble learning method for monaural speech separation". In: *IEEE/ACM Transactions on Audio, Speech and Language Processing* 24.5 (2016), pp. 967–977.

[178] Donald Williamson and DeLiang Wang. "Time-Frequency Masking in the Complex Domain for Speech Dereverberation and Denoising". In: *IEEE/ACM Transactions on Audio, Speech, and Language Processing* (2017).

[179] Yong Xu et al. "A regression approach to speech enhancement based on deep neural networks". In: *IEEE/ACM Transactions on Audio, Speech and Language Processing* 23.1 (2015), pp. 7–19.

[180] Yong Xu et al. "Dynamic noise aware training for speech enhancement based on deep neural networks". In: *Fifteenth Annual Conference of the International Speech Communication Association*. 2014.

[181] Yong Xu et al. "Global variance equalization for improving deep neural network based speech enhancement". In: *IEEE China Summit & International Conference on Signal and Information Processing*. IEEE. 2014, pp. 71–75.

[182] Guoning Hu. "Monaural speech organization and segregation". PhD thesis. The Ohio State University, 2006.

[183] Zhaozhang Jin and DeLiang Wang. "A supervised learning approach to monaural segregation of reverberant speech". In: *IEEE Transactions on Audio, Speech, and Language Processing* 17.4 (2009), pp. 625–638.

[184] Ron J. Weiss and Daniel P. W. Ellis. "Estimating single-channel source separation masks: relevance vector machine classifiers vs. pitch-based masking." In: *SAPA@ INTERSPEECH*. 2006, pp. 31–36.

[185] Kun Han and DeLiang Wang. "A classification based approach to speech segregation". In: *The Journal of the Acoustical Society of America* 132.5 (2012), pp. 3475–3483.

[186] Gibak Kim and Philipos C. Loizou. "Improving speech intelligibility in noise using environment-optimized algorithms". In: *IEEE/ACM transactions on audio, speech, and language processing* 18.8 (2010), pp. 2080–2090.

[187] Gibak Kim et al. "An algorithm that improves speech intelligibility in noise for normal-hearing listeners". In: *The Journal of the Acoustical Society of America* 126.3 (2009), pp. 1486–1494.

[188] Yuxuan Wang, Kun Han, and DeLiang Wang. "Exploring monaural features for classification-based speech segregation". In: *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 21.2 (2013), pp. 270–279.

[189] Antony W. Rix et al. "Perceptual evaluation of speech quality (PESQ)-a new method for speech quality assessment of telephone networks and codecs". In: *IEEE International Conference on Acoustics, Speech, and Signal Processing*. Vol. 2. IEEE. 2001, pp. 749–752.

[190] Cees H. Taal et al. "A short-time objective intelligibility measure for time-frequency weighted noisy speech". In: *IEEE International Conference on Acoustics Speech and Signal Processing*. IEEE. 2010, pp. 4214–4217.

[191] ITU-T. "Perceptual evaluation of speech quality (PESQ): An objective method for end-to-end speech quality assessment of narrow-band telephone networks and speech codecs, ITU-T Recommendation P.862". In: (2000).

[192] ITU-T. "Wideband extension to recommendation P. 862 for the assessment of wideband telephone networks and speech codecs, ITU-T Recommendation P.862.2". In: (2007).

[193] ITU-T. "862.1: Mapping function for transforming P. 862 raw result scores to MOS-LQO, ITU-T, Recommendation P.862.1". In: (2003).

[194] Ulrik Kjems et al. “Role of mask pattern in intelligibility of ideal binary-masked noisy speech”. In: *The Journal of the Acoustical Society of America* 126.3 (2009), pp. 1415–1426.

[195] Torsten Dau, Dirk Püschel, and Armin Kohlrausch. “A quantitative model of the “effective”signal processing in the auditory system. I. Model structure”. In: *The Journal of the Acoustical Society of America* 99.6 (1996), pp. 3615–3622.

[196] Jesper B. Boldt and Daniel P. W. Ellis. “A simple correlation-based model of intelligibility for nonlinear speech enhancement and separation”. In: *Signal Processing Conference, 2009 17th European*. IEEE. 2009, pp. 1849–1853.

[197] Ray L. Goldsworthy and Julie E. Greenberg. “Analysis of speech-based speech transmission index methods with implications for nonlinear operations”. In: *The Journal of the Acoustical Society of America* 116.6 (2004), pp. 3679–3689.

[198] Yariv Ephraim and David Malah. “Speech enhancement using a minimum mean-square error log-spectral amplitude estimator”. In: *IEEE Transactions on Acoustics, Speech, and Signal Processing* 33.2 (1985), pp. 443–445.

[199] Dionysis E. Tsoukalas, John N. Mourjopoulos, and George Kokkinakis. “Speech enhancement based on audible noise suppression”. In: *IEEE Transactions on Speech and Audio Processing* 5.6 (1997), pp. 497–514.

[200] Sepp Hochreiter and Jürgen Schmidhuber. “Long short-term memory”. In: *Neural Computation* 9.8 (1997), pp. 1735–1780.

[201] Kyunghyun Cho et al. “On the properties of neural machine translation: Encoder-decoder approaches”. In: *arXiv preprint arXiv:1409.1259* (2014).