

FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO



GAN-Based Methods For Whisper-To-Speech Conversion

José Diogo Pereira Tender Dias

Mestrado em Engenharia Eletrotécnica e de Computadores

Advisor: Dr. Aníbal João de Sousa Ferreira

July 29, 2022

Abstract

As a form of communication, speech conveys linguistic and paralinguistic information. In particular, whispered speech is characterized by the lack of vibration of the vocal folds. This oral mode of communication may be used to emphasize privacy or secrecy. However, a problem may arise when a condition of aphonia, temporary or permanent, restricts patients to speak in whispered speech only, which reduces speech projection and intelligibility. This dissertation is an extension of the DyNaVoiceR project, that focuses on developing an assistive, non-intrusive, real-time technology that allows efficient and practical communication to those with aphonia. In this context, the current dissertation explores the conversion of the spectral envelope from whispered speech to its normal speech counterpart. The solution presented relies on Generative Adversarial Networks, namely on the cycleGAN and discoGAN architectures, and a traditional vanilla GAN to act as baseline. For spectral envelope mapping, the DyNaVoiceR corpora was used. These architectures are trained with parallel and non-parallel corpora, and with gender-specific and non-gender-specific corpora, over the four most frequent vowels available. A self-supervised version that included an additional prediction loss was also considered for the parallel mode. Validation of the trained models included Mel Cepstral Distortion, as an objective metric, and Mean Opinion Score, as a subjective metric. The obtained objective results for the unsupervised cycleGAN trained with non-parallel corpora improved up to 23.5% compared to the baseline vanilla GAN. Moreover, the self-supervised discoGAN trained with parallel corpora improved up to 64.5%. Subjective results indicate that discoGAN is preferred to cycleGAN in terms of proximity to the original normal speech. Overall, the results obtained support the viability of the proposed architectures for on-the-fly cross-domain learning in the context of whisper-to-normal speech conversion. For the DyNaVoiceR project, self-supervised discoGAN with parallel training is highly effective, as speech corpora is manually annotated, simplifying the training and enabling higher quality results.

Resumo

Como forma de comunicação, a fala transmite informação linguística e paralinguística. Em particular, a fala sussurrada é caracterizada pela falta de vibração das cordas vocais. Este modo de comunicação oral pode ser usado para enfatizar privacidade ou sigilo. No entanto, um problema pode surgir quando afonia, temporária ou permanente, restringe os pacientes a utilizar apenas fala sussurrada, reduzindo a projeção e a inteligibilidade da fala. Esta dissertação é uma extensão do projeto DyNaVoiceR, que se foca no desenvolvimento de tecnologia que permita uma comunicação eficiente e prática aos pacientes com afonia. Neste contexto, esta dissertação explora a conversão da envelope espectral de fala sussurrada para sua contraparte de fala normal. Este trabalho baseia-se em *Generative Adversarial Networks*, nomeadamente cycleGAN e discoGAN. Além disso, uma GAN tradicional é implementada como *baseline*. As arquiteturas consideradas são treinadas com corpora paralela e não paralela, e corpora gênero-específico e não específico, considerando as quatro vogais mais frequentes disponíveis. Adicionalmente, uma versão auto-supervisionada que inclui uma perda de previsão adicional também foi considerada para o modo paralelo. A validação dos modelos treinados incluiu *Mel Cepstral Distortion*, como métrica objetiva, e *Mean Opinion Score*, como métrica subjetiva. Os resultados objetivos relativos à cycleGAN não supervisionada com corpora não paralela mostraram melhorias até 23.5% em comparação com a GAN tradicional. Além disso, a discoGAN auto-supervisionada com corpora paralela melhorou até 64.5%. Os resultados subjetivos indicam que a discoGAN é preferível em relação à cycleGAN, em termos de proximidade com a fala normal original. No geral, os resultados obtidos suportam a viabilidade das arquiteturas propostas para *cross-domain learning* em conversão de sussurro para fala normal. Para o projeto DyNaVoiceR, a DiscoGAN auto-supervisionada em treino paralelo é altamente eficaz. Isto porque os dados de fala são anotados manualmente, simplificando o processo de treino e maximizando os resultados obtidos.

Agradecimentos

Quero agradecer a todos aqueles que me ajudaram e apoiaram, direta ou indiretamente, durante esta dissertação de mestrado.

Em primeiro lugar, ao Professor Doutor Aníbal João de Sousa Ferreira, pela oportunidade de me envolver no projeto, pela excelente orientação e todo o saber que transmitiu, e pelas sugestões e correções do documento, e ao co-orientador Engenheiro Marco Oliveira, pelo apoio, companheirismo, pelas constantes sugestões e correções no documento, e pelo esforço em acolher-me como colega no projeto. Adicionalmente, à Faculdade de Engenharia da Universidade do Porto e a todo o seu corpo docente.

A todos os amigos e colegas que, de forma direta ou indireta, auxiliaram na elaboração do presente estudo. Além disso, pela paciência e atenção que prestaram nos bons e maus momentos. Em especial a Pedro, Filipa, Ana, Daniel e Carolina pelas sugestões e correções do documento.

Aos meus pais e irmão, por todo o apoio que me deram e continuam a dar. Em particular, à minha mãe, por todo o carinho, motivação, e a constante disponibilidade de ajudar. Ao meu irmão, por estar sempre presente ao meu lado, e pelo apoio que me deu.

Por fim, o meu profundo agradecimento a todas as pessoas que contribuíram para a concretização desta dissertação.

José Diogo Pereira Tender Dias

Contents

1	Introduction	1
1.1	Context	1
1.2	Motivation	1
1.3	Objectives	2
1.4	Document Structure	2
2	Basics of Speech Processing	3
2.1	Speech	3
2.1.1	Respiratory System	3
2.1.2	Larynx	4
2.1.3	Vocal Tract	5
2.1.4	Fundamental Frequency and Resonance	5
2.1.5	Voiced Speech and Whispered Speech	5
2.2	Auditory System	6
2.2.1	Peripheral Auditory System	7
2.2.2	Psychoacoustics	7
2.3	Digital Signal Processing of Speech	10
2.3.1	Source-Filter Model	10
2.3.2	Linear Predictive Coding	10
2.3.3	Cepstral Analysis	11
2.3.4	Mel-Frequency Cepstral Coefficients	11
2.3.5	Mel Cepstral Coefficients	12
3	Machine Learning	13
3.1	Basics of Machine Learning	13
3.1.1	Learning Systems	13
3.2	Machine Learning Categories	14
3.2.1	Supervised Learning	14
3.2.2	Unsupervised Learning	14
3.2.3	Semi-supervised Learning	15
3.3	Model Taxonomy	15
3.3.1	Discriminative Models	15
3.3.2	Generative Models	15
3.4	Neural Networks	15
3.4.1	Perceptron	16
3.4.2	Feed-Forward Neural Network	17
3.4.3	Convolutional Neural Network	18
3.5	Generative Models	19

3.5.1	Taxonomy of Generative Models	20
3.5.2	Maximum Likelihood Estimation	20
3.5.3	Fully Visible Belief Networks	21
3.5.4	Variational Autoencoders	21
3.5.5	Generative Adversarial Networks	22
4	State of the Art	24
4.1	Problem Formulation	24
4.2	GAN-Based Methods For Speech Processing	25
4.2.1	Fundamental Frequency Prediction	25
4.2.2	Spectral Envelope Mapping	25
4.3	Reference System Architectures	25
4.3.1	Cycle-GAN	25
4.3.2	CinC-GAN	26
4.3.3	Disco-GAN	27
4.3.4	MMSE-discoGAN	28
4.4	Other System Architectures	28
5	Methods	30
5.1	Dataset	31
5.1.1	Descriptive Analysis	32
5.1.2	Training Data	34
5.1.3	Validation Data	34
5.1.4	Testing Data	34
5.2	Spectral Envelope Modelling	35
5.3	Cepstral Features	37
5.3.1	Feature Extraction	37
5.3.2	Whispered Speech Spectral Envelope	39
5.3.3	Normal Speech Spectral Envelope	40
5.3.4	Feature Batching	42
5.4	GAN Topologies	42
5.4.1	Nomenclatures	44
5.4.2	Neural Network Layers	44
5.4.3	Graphical Overview	45
5.4.4	Optimizer	46
5.4.5	Loss Functions	47
5.4.6	Hyperparameters	50
5.5	Evaluation Metrics	51
5.5.1	Objective Criterion	51
5.5.2	Subjective Criterion	51
6	Results and Discussion	54
6.1	MCD Analysis	55
6.1.1	MCD for input speech	55
6.1.2	MCD analysis for generated speech	56
6.2	Generator Modal Effectiveness	59
6.2.1	Non-Parallel versus Parallel Training	59
6.2.2	Zero-Shot Learning	61
6.3	Subjective Test Results	63

6.3.1	Artifacts and Speech Quality	65
6.3.2	Proximity and Speech Naturality	67
6.4	Summary	69
7	Conclusions	71
7.1	Conclusion Synthesis	72
7.2	Future Work	73
A	DyNaVoiceR Dataset Tasks	74
B	Preliminary Study I: LPC-based Spectral Mapping	77
B.1	64 Point Representation	77
B.2	22 Levinson Coefficients	78
C	Results: Constant Learning Rate	80
D	Results: Batch Size	82
	References	83

List of Figures

2.1	Speech production system. Adapted from [1].	4
2.2	Phonemes and the different formant shapes. Adapted from [2]	6
2.3	Fletcher–Munson equal loudness curves. Adapted from [3][2].	8
2.4	Linear Predictive Coding system overview. Adapted from [2].	10
2.5	Log-spectrum of a signal (left) and the respective cepstrum (right). The quefrency coefficients represent the signal formants, and the isolated peak represents the fundamental period. Adapted from [2]	11
2.6	Block diagram for MFCC extraction. Adapted from [4]	12
3.1	Perceptron’s system processing. Inspired by [5]	16
3.2	Linear separability in Neural Network. Inspired by [5]	17
3.3	Adding layers to the perceptron model, building a multi-layer neural network. Adapted from [6]	18
3.4	CNN for an RGB image as input with one convolutional layer and one pooling layer. Adapted from [5][7]	19
3.5	Taxonomy of generative models from Ian Goodfellow’s NIPS tutorial, 2016 [8] .	20
3.6	Block diagram for GAN and respective framework. By [5][8]	22
4.1	Block diagram for Cycle-GAN, as proposed by [9]	26
4.2	Block diagram for CinC-GAN, as proposed by [9]	27
4.3	Block diagram for Disco-GAN, as proposed by [10]	28
4.4	Block diagram for MMSE-discoGAN, as proposed by [11]	28
5.1	Block diagram of the DyNaVoiceR platform and proposed GAN-based spectral envelope mapping	30
5.2	Block diagram of spectral envelope models.	35
5.3	Frame Window	35
5.4	Spectral envelope models for (a) whispered speech and (b) normal speech using Levinson-Durbin Recursion	36
5.5	Block diagram of feature retrieval and methodology	37
5.6	Block diagram of spectral envelope modelling	38
5.7	Cepstral Filterbanks: 22 versus 40 triangular filters	38
5.8	Whispered speech spectral envelope model	39
5.9	Spectral envelope from MFCCs: 22 versus 40 Coefficients	40
5.10	Normal speech spectral envelope model: 22 LPC coefficients	40
5.11	MFCC-derived spectral envelope: 22 versus 40 MFCCs	41
5.12	Normal speech spectral envelope model: 128 LPC coefficients and peak analysis .	41
5.13	Higher quality MFCC-derived spectral envelope: 22 versus 40 MFCCs	42
5.14	Block diagram of training and testing processes	43

5.15	Network Activation Functions: Sigmoid and ReLU	45
5.16	Block diagram of the Generator	46
5.17	Block diagram of the Discriminator	46
5.18	Block diagram of the training process and respective losses for CycleGAN	48
5.19	Block diagram of the training process and respective losses for DiscoGAN	49
5.20	Adaptive learning rate overview given 500 epochs, introducing a linear decrease after epoch 250	51
6.1	DiscoGAN versus cycle GAN: parallel supervised in comparison to non-parallel unsupervised learning, respectively	60
6.2	DiscoGAN - Zero-Shot Learning spectral envelope compared to the ground truth.	62
6.3	CycleGAN - Zero-Shot Learning spectral envelope compared to the ground truth.	62
6.4	European Portuguese oral vowels positions in the vowel triangle (vowels used in the training phase are highlighted with a red circle).	63
6.5	MOS average results for tasks 1 and 2: synthetic voiced speech of vowels seen in training, with respect to the presence of undesired artifacts, with 95% confidence intervals.	66
6.6	MOS average results for task 3: synthetic voiced speech of one seen vowel and an unseen vowel for GAN-based models (models C and D), with respect to the presence of undesired artifacts, with 95% confidence intervals.	67
6.7	MOS average results for tasks 4 and 5: synthetic voiced speech of vowels seen in training, with respect to the proximity to the normal speech reference (ground truth), with 95% confidence intervals.	68
6.8	MOS average results for task 6: synthetic voiced speech of one seen vowel and an unseen vowel for GAN-based models (models C and D), with respect to the proximity to the normal speech reference (ground truth), with 95% confidence intervals.	69
A.1	Page 1 of the DyNaVoiceR tasks	74
A.2	Page 2 of the DyNaVoiceR tasks	75
A.3	Page 3 of the DyNaVoiceR tasks	76
B.1	CycleGAN - Parallel Training for 64 bins per frame	77
B.2	CycleGAN - Comparison from input for 64 bins per frame	78
B.3	CycleGAN - Comparison from input for 22 LPC coefficients per frame	78
B.4	CycleGAN - Loss given different inputs	79

List of Tables

2.1	Bark Scale, according to [12].	9
5.1	Oral vowels and respective examples, based in IPA for the European Portuguese Language.	31
5.2	Frequency table for used phonemes. In brackets, outliers are represented.	32
5.3	Ratio of occurrence for the nine vowels, considering whispered speech and normal speech. Information in brackets indicates the relation to the assumption of uniform distribution for a phoneme-balanced dataset.	33
5.4	Vowels and respective purposes: training dataset and zero-shot learning dataset	33
5.5	Phoneme occurrences based on gender for the four training vowels.	33
5.6	Phonemes used in the training dataset, respective examples and the total number of occurrences displayed as $W + S$, where W represents whispered speech frames, and S represents normal speech frames.	34
5.7	Vanilla GAN Topology.	43
5.8	DiscoGAN Topology.	43
5.9	Cycle-GAN Topology.	43
5.10	Hyperparameters obtained through experimentation for considered architectures	50
5.11	MOS tasks: specific words uttered in each task	52
5.12	MOS Tasks: scale and indications	52
5.13	MOS Tasks: scale and indications	53
6.1	Vowels and respective purposes: training dataset and zero-shot learning dataset, with respective examples.	54
6.2	MCD between input whispered speech and input normal speech for the training dataset consisting of four vowels.	56
6.3	MCD between input whispered speech and input normal speech for the zero-shot learning vowels.	56
6.4	Mel Cepstral Distortion (MCD) for the baseline model, Vanilla GAN, and proposed architectures. Training used 500 epochs. The results are separated into training type and gender-specific data. The percentage in brackets indicates the relative reduction in the MCD w.r.t the baseline model.	57
6.5	Mel Cepstral Distortion (MCD) for the proposed architectures considering self-supervised parallel training. Training utilized 500 epochs. The results are separated by training type and gender-specific data. The percentage in brackets indicates the relative reduction in the MCD w.r.t the baseline, which is the unsupervised parallel mode of training.	58

6.6	Mel Cepstral Distortion (MCD) for cycleGAN over specified number of epochs, considering the models are trained using 100, 250, and 500 epochs. The highlighted MCD scores reveal the best results obtained for the considered architecture.	60
6.7	Mel Cepstral Distortion (MCD) for discoGAN over a specified number of epochs, considering that the models are trained using 100, 250, and 500 epochs. The highlighted MCD scores reveal the best results obtained for the considered architecture.	61
6.8	Mel Cepstral Distortion (MCD) for training data (seen vowels) considering cycleGAN in non-parallel training and discoGAN in supervised parallel training. The models were trained using 100, 250, and 500 epochs.	61
6.9	Mel Cepstral Distortion (MCD) considering zero-shot learning (unseen vowels), for cycleGAN and discoGAN. Training used only seen vowels, and results are separated into training mode and gender-specific data. The information in brackets presents the relative variation of the zero-shot learning (unseen vowels) MCD compared to the training dataset (seen vowels). The models were trained using 500 epochs.	62
6.10	MOS models: model specifications, and respective nomenclatures used.	64
6.11	MOS questions, auditory task, and the uttered word for each task.	65
6.12	Mean Opinion Score (MOS) results obtained considering presence of artifacts, for cycleGAN and discoGAN, regarding seen and unseen vowels, i.e. training and zero-shot learning vowels	67
6.13	Mean Opinion Score (MOS) results obtained considering proximity to the normal speech reference, for cycleGAN and discoGAN, regarding seen and unseen vowels, i.e. training and zero-shot learning vowels	69
B.1	MCD analysis of LPC versus MFCC methods, for cycle-GAN with Parallel Training	79
C.1	Mel Cepstral Distortion (MCD) for the baseline model, Vanilla GAN, and utilized architectures. Training utilizes 500 epochs, and results are separated into training type and speaker gender. Results consider a constant learning rate.	80
C.2	Inconsistencies in Mel Cepstral Distortion (MCD) for the proposed training epochs, considering 400, 500, and 1000 epochs. Inconsistencies are defined as decreases in MCD for an increasing number of epochs, or increases in MCD for a decreasing number of epochs.	81
C.3	Increase in stability utilizing linear descent learning rate, measured in Mel Cepstral Distortion (MCD) for the proposed training epochs, considering 100, 250, and 500 epochs.	81
D.1	Mel Cepstral Distortion (MCD) for the proposed 500 training epochs. Results for batch size $B = 100$, compared to $B = 200$, for all speakers.	82

Abbreviations

AI	Artificial Intelligence
DCT	Discrete Cosine Transform
DFT	Discrete Fourier Transform
DyNaVoiceR	Dysphonic to Natural Voice Reconstruction
FEUP	Faculdade de Engenharia da Universidade do Porto
GAN	Generative Adversarial Network
GMM	Gaussian Mixture Models
HMM	Hidden Markov Models
IPA	International Phonetic Alphabet
LPC	Linear Predictive Coding
MCC	Mel Cepstral Coefficient
MCD	Mel Cepstral Distortion
MFCC	Mel Frequency Cepstral Coefficient
ML	Machine Learning
MOS	Mean Opinion Score
NN	Neural Network
ODFT	Odd-Discrete Fourier Transform
PSD	Power Spectral Density
STD	Standard Deviation
STFT	Short-Time Fourier Transform
VAE	Variational Autoencoder
WHSP2SPCH	Whisper to Speech

Chapter 1

Introduction

This section gives a brief introduction to the topic of whisper to normal speech conversion and the specific engineering area revolving around the DyNaVoiceR project.

1.1 Context

Human beings are a social species that communicate and interact with each other. Speech, the ability to express thoughts and feelings by articulating sounds, enables interactions and empowers individuals to give meaning to their surrounding environments. As humans perceive and understand the world, they also have the potential to articulate themselves and give meaning to their experiences. Whispered speech, as a mode of communication, may be considered an alternative to normal voiced speech, for privacy, secrecy, or other personal reasons. The problem arises when voice dysphonia, notably temporary or permanent aphonia, hurdles patients into being able to only perform whispered speech [2]. Whispered speech reduces speech projection and intelligibility, therefore, impacting people’s social and professional lives [9].

This dissertation is contextualized in the DyNaVoiceR Project [13]. This project focuses on advanced assistive technology that helps patients affected by voice dysphonia communicate effectively and comfortably. In this context, this project is currently exploring WHiSPer-to-normal SPeeCH (WHSP2SPCH) conversion.

1.2 Motivation

The motivation surrounding the present dissertation is to allow GAN-based technologies to be accessible as a countermeasure for patients with aphonia [13]. Fundamentally, this allows to further increase intelligibility of whispered information by humans and machines, and to allow the ability to perform voiced speech by aphonia patients.

1.3 Objectives

Two main approaches can be identified in the literature regarding WHSP2SPCH conversion: rule-based and statistical-based [14]. Rule-based approaches rely on modifying speech parameters according to empirical rules, but have been deemed unsatisfactory in the literature [14]. This dissertation focuses on machine learning-based approaches.

Early statistical conversion attempts have used Gaussian Mixed Models [15]. Recent studies have also introduced Deep Neural Networks (DNN) [16]. The current approach based on generative learning allows the mapping module to learn from a large amount of speech data, as it does not require parallel training corpora, that is, speech data temporally aligned in both domains (whispered and normal speech) [9]. This approach greatly improves voice quality [17]. Based on this perspective, Generative Adversarial Networks (GAN) methods have been proposed for WHSP2SPCH, such as CinC-GAN [9], CycleGAN [9], and Disco-GAN [10]. Experimental results show that, with non-parallel training data, CycleGAN achieves comparable performance to that of a GMM-based system that is trained on twice the amount of parallel data [17].

GAN-based methods show high objective and subjective evaluation results [18, 9], suggesting a natural-sounding WHSP2SPCH conversion [10]. The main goal for this dissertation is to develop a GAN-based method for spectral envelope remapping, where whispered speech spectral envelopes are converted to normal speech spectral envelopes, in the context of the DyNaVoiceR project and assess its results in comparison to the project rule-based conversion approach.

1.4 Document Structure

The structure of this document is separated by chapters in the following manner:

- Chapter 2 - Basics of Speech Processing - explains the theoretical aspects of speech processing relevant to the understanding of the scientific or engineering area where this dissertation is contextualized.
- Chapter 3 - Machine Learning - explains contextualized background knowledge in machine learning and briefly explains its history, to better understand the following chapters.
- Chapter 4 - State of the Art - attempts to contextualize and explain WHSP2SPCH conversion and relevant work done in the area, to the extent of the author's knowledge.
- Chapter 5 - Methods - explains the methods used in order to create the GAN-based models for spectral envelope mapping.
- Chapter 6 - Results and Discussion - illustrates results obtained and the comparison in different utilized architectures, as well as, a discussion of the obtained results.
- Chapter 7 - Conclusion - summarizes the conclusions derived from this dissertation. Besides, findings and obtained results are presented. In addition, future work is discussed.
- Lastly, appendices A, B, C and D highlight materials, preliminary studies and complementary results.

Chapter 2

Basics of Speech Processing

To understand the context of WHSP2SPCH conversion and the related speech processing concepts, a degree of background knowledge is necessary. This chapter explains the fundamental concepts of speech processing, allowing the reader to understand the background necessary for the current dissertation.

2.1 Speech

Speech holds transmission rates considerably high, up to 20 to 30 segments (symbols) per second [19]. Its study is divided into two sections:

1. Psycho linguistics, the study of the processes that occurs in the brain, the representation of the sound sequence and several commands that the speech organs will execute to produce the utterance [20]. This implies the need for a phonetic plan and a motor plan.
2. Physical production of sounds. Speech is produced by an air stream from the lungs, that propagates through the trachea and, lastly, the oral and nasal cavities. Speech production involves three stages – exhalation, phonation, and articulation - associated, at a peripheral level, respectively, with the subglottic respiratory system, larynx, and vocal tract. [21].

2.1.1 Respiratory System

The respiratory system has the responsibility to propagate air in and out of the body, that is, the intake of oxygen and the exhalation of carbon dioxide [22]. The air flow through the respiratory system is the element that allows speech production to be possible.

To initiate speech, the thorax is expanded by inspiration [21]. Then, while exhaling air, the volume of the chest cavity is reduced, causing the lung air pressure to increase. This increase in pressure causes air to flow through the trachea into the larynx [22].

Different views in the regulation of airflow have been studied. The main divergence is the role of the abdominal system in the regulation of airflow [23]. From a contemporary perspective, as air

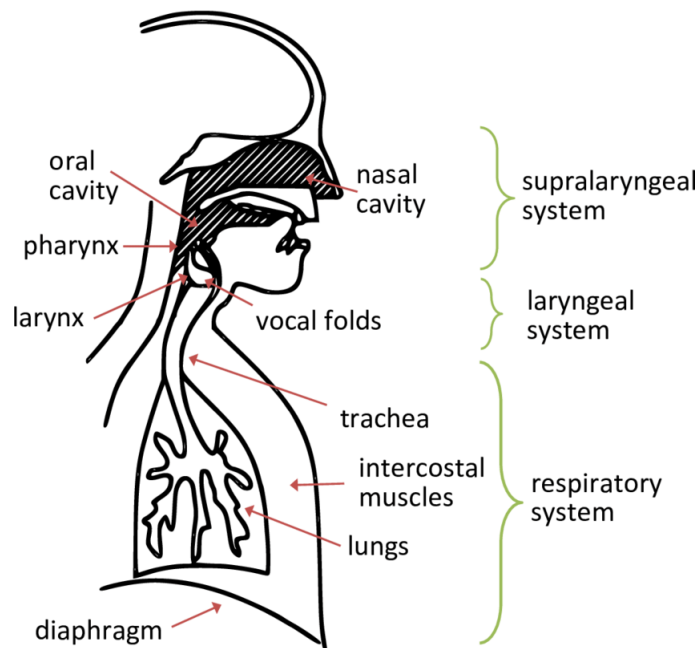


Figure 2.1: Speech production system. Adapted from [1].

travels to the larynx, co-activation of the inspiratory and expiratory muscles allows the necessary regulation to maintain the consistent pressure for speech to continue[21].

2.1.2 Larynx

The larynx is the area that connects the throat to the trachea. Its walls are made of connective tissue, muscles and cartilage, where cartilage provides support and also keeps the larynx elastic [24]. The larynx controls the vocal folds. These are the two masses of ligament and muscle stretching from the posterior to the anterior of the larynx. Also, the glottis is the opening between the two folds [22].

The vocal folds can assume three states - breathing, voiced and unvoiced [22]. In the breathing state, the glottis is wide open. Thus, no resistance is applied to the air pressure. In voiced speech, the glottis is narrow [21], and the vocal folds are close to each other. Considering the air pressure due to exhalation, the glottis will oscillate between states of closed positions and open positions [22]. As a property of the vocal folds when vibrating, its oscillation rate is measurable in terms of frequency [25]. Parameters that regulate this vibration are the transglottal pressure difference and glottal airflow [21].

The larynx holds essential elements for phonation, the process by which the vocal folds produce certain sounds through quasi-periodic vibration, allowing the sensation of pitch. Thus, phonation allows for the fundamental blocks of voiced speech, which consist of voiced phonemes [21].

2.1.3 Vocal Tract

The vocal tract contains two elements: the oral cavity that extends from the larynx to the lips [22] and the nasal cavity, which is coupled to the oral cavity via the velum [22]. As air from the lungs passes through the larynx, it then reaches the vocal tract. This part of the process of speech production is preparatory for articulation. Speech articulation is the most complex motor activity in humans, producing concatenations of phonemes into syllables and syllables into words [21].

When the input to the vocal tract is the periodic glottal source waveform, the resulting output speech spectrum can be obtained as the product of the glottal source spectrum and the vocal tract's frequency response [22].

2.1.4 Fundamental Frequency and Resonance

Fundamental Frequency (F_0) of the voice is the lowest harmonic component in voiced sounds [21] and it plays an important role in transmitting not only linguistic information, but also non-linguistic information such as naturalness, emotion, and speaker idiosyncrasy [26]. Thus, it is an essential element for the transmission of prosodic information. The typical range in adult speakers is about 80-400Hz in males and 120-800 Hz in females [21].

As the vocal tract is in constant shift in speech production, different resonances are produced, leading to a variance in magnitude for the frequency components of the excitation. This effect results in different sounds. The described resonance frequencies are usually referred to as formant frequencies [27, 22]. Each formant represents part of the vocal tract. The first formant F_1 is associated with changes in mouth opening [22], while the second formant F_2 is associated with changes in the oral cavity, such as tongue position and lip activity [22]. These elements - fundamental frequency and formants - are fundamental in digital speech processing, especially in order to comprehend and simulate vowels. Figure 2.2 illustrates typical changes according to the different speech units, phonemes.

2.1.5 Voiced Speech and Whispered Speech

Whispered speech is a naturally produced mode of communication. It distinguishes itself by the absence of fundamental frequency [28]. Other acoustic differences exist between whispered and voiced speech, such as speaking rate (measured as segment duration) and formant frequencies [28]. Whispered speech is generated by allowing the vocal folds to be in the unvoiced state [22]. Whispering means that all phonemes are spoken unvoiced or unphonated, although there are some variants referred to as "whispering", even though these are actually semiphonated [3]. In this state, the vocal folds do not vibrate, but instead become tense and tight. This leads to a turbulent air stream as it flows through the glottis. The air turbulence present in the process is called aspiration [22]. Despite this, the sensation of pitch in whispered speech has been reported and is encapsulated in an intricate way [29]. Fundamental frequency does not exist physically or phonetically due to the lack of vocal fold vibration. As there is no periodicity, the sensation of pitch is associated to the perception of the fundamental frequency countour [29].

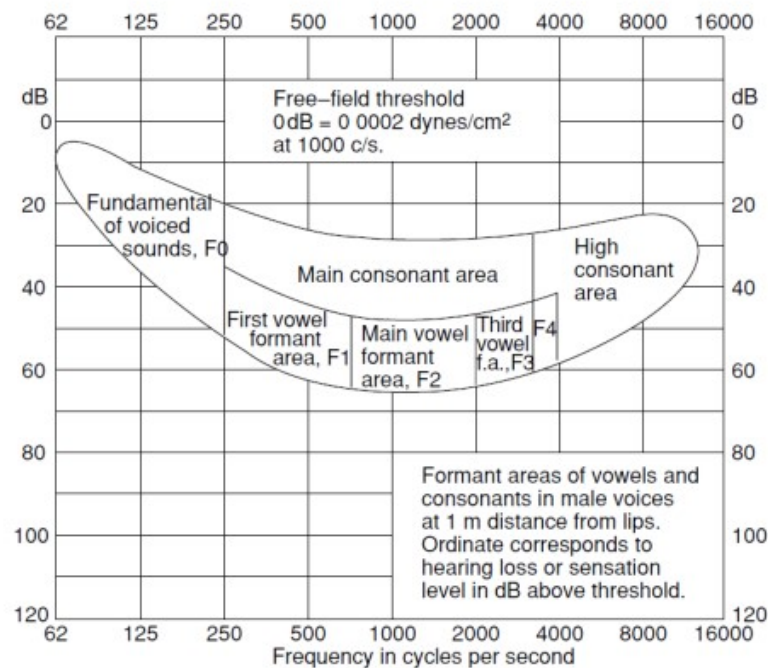


Figure 2.2: Phonemes and the different formant shapes. Adapted from [2]

Normal voiced speech is a natural mode of communication with fundamental frequency. It is generated by the exhalation of air, since the narrow glottis allows vibration in the vocal folds. Moreover, the vocal tract filters the sound according to transmitted information [22]. As the vocal folds oscillate in a quasi-periodic form, giving a perception of pitch in voiced speech, this denotes the existence of a fundamental frequency and harmonics. The respective periodic signal components are affected in different ways, enabling the generation of different phonemes. The vocal tract acts as a resonator for the corresponding harmonics and enables different forms of phonemes [22].

Whispered speech can be observed as an alternative to normal voiced speech. However, considering temporary or permanent aphonia, where patients can only perform whispered speech as their main means of oral communication, this creates hurdles with impact in people's social and professional lives [2].

2.2 Auditory System

The auditory system is an important part of the human sensory nervous system. It includes the ear and neuronal structure to process the sense of hearing. This system holds great importance for speech processing, as speech is perceived by listeners.

2.2.1 Peripheral Auditory System

The peripheral part of the auditory system can be divided into three sections: the outer, the middle, the inner ear, and the cochlear nerve to the point where it communicates with the central nervous system. The peripheral auditory system is designed to collect, filter, amplify, and convert the pressure of sound waves into a neural impulse pattern for transmission to the central nervous system for further processing [30].

The outer ear section consists of the auricle and the ear canal that leads to the tympanic membrane. It transports the sound wave to the middle section of the ear. The middle ear section is separated from the outer ear by the tympanic membrane. It consists of the tympanic cavity and epitympanic recess, and its function is to transform the energy of pressure waves into mechanical vibrations. The middle and outer ear act as a filter for the sound wave. Frequencies in the 1000 Hz to 2000 Hz range are essentially unfiltered, but transmitted energy at lower (<1000 Hz) frequencies is reduced at a rate of 12 dB per halving of frequency (i.e., per octave) and high frequencies (>4000 Hz) are attenuated at a rate of 6 dB per octave [30]. The inner ear is responsible for transforming the mechanical waves received by the middle ear into neurological messages sent by the cochlear nerves that communicate with the central nervous system. The cochlea is the spiral shaped part of the inner ear that is responsible for hearing, and the basilar membrane allows for frequency discrimination along the spiral [30].

The cochlea transmits the mechanical signals as waves in its fluid and membranes. Lastly, it converts them to nerve impulses, which are transmitted to the brain [31]. The basilar membrane (BM) acts as a logarithmic spectrum analyser, dividing and isolating the input vibration frequency components to different physical locations along its length [32]. In this manner, high frequencies vibrate near the base and low frequencies vibrate near the cochlear apex [30], with a frequency range of the order of 20 - 20 000 Hz in the case of young adults and children [30]. A fundamental point to this process is that each vibration resonates along the coil in different physical locations. This is the frequency-to-place transformation model [32]. The frequency selectivity of the basilar membrane may be considered similar to the functionality provided by a bank of band-pass filters, with a threshold of audibility in each filter being dependent upon the noise energy falling within its passband [3].

Auditory critical bands can be defined as the range of audio frequencies within which a masking tone will interfere with the perception of another tone through various mechanisms of auditory masking [32]. Auditory masking is the process by which the audibility threshold for one sound is raised by the presence of another sound [3]. The effect of the critical-band filters is that, for a given tone that has fixed frequency and amplitude, the sensitivity of the ear to tones of similar frequency is reduced [3].

2.2.2 Psychoacoustics

Psychoacoustics is the scientific study of sound perception and audiology, which includes the perception of speech. Audio signals have inherent subjective components. The most common

subjective characteristics of a sound are loudness, pitch and timbre [33, 32]. These characteristics correspond to the measurable counterparts: sound intensity, fundamental frequency and spectral structure, respectively [32].

Loudness has been characterized in multiple ways, but a standard has been adopted. The Fletcher–Munson equal loudness curves have been taken for many years as the standard for representing the variations in perceptual loudness [32] using the x-axis to represent the frequency and the y-axis to represent the sound pressure level (SPL).

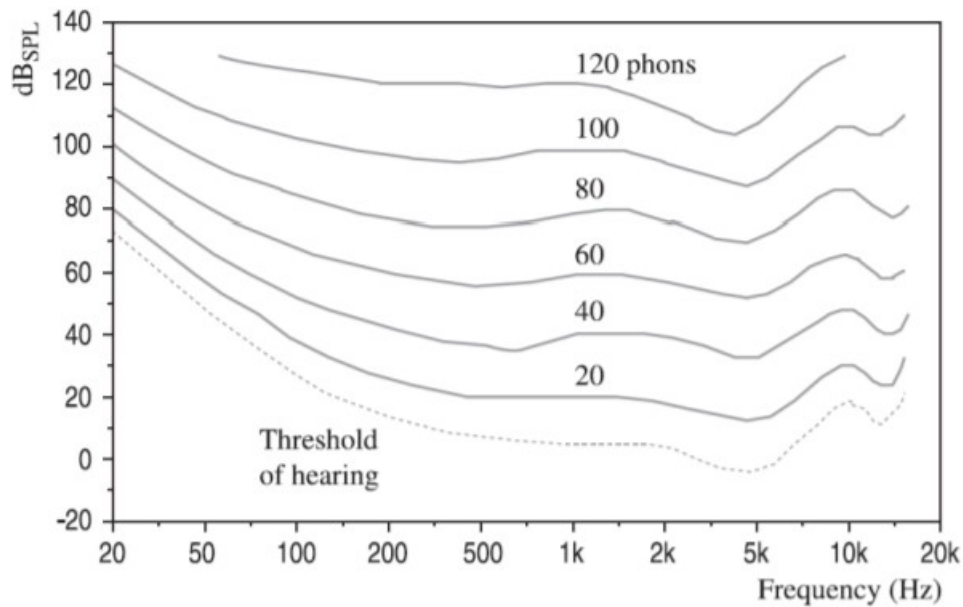


Figure 2.3: Fletcher–Munson equal loudness curves. Adapted from [3][2].

The focal point of this study is to allow for a basic understanding of the auditory system. As quoted by [3] "If there is one topic that has most deeply impacted audio and speech research over the past two decades or so, it is psychoacoustics [...]". The attempt to explain the basics of psychoacoustics in this dissertation is justified by the need to understand the lack of linearity in the auditory system. Due to the non-linearity, perceptual frequency scales have been devised and, in particular, the Bark [32, 12], the Mel Scale [32] and the Equivalent Rectangular Bandwidth (ERB)[32] have been deeply studied and used in speech processing.

Mel scale (short for *melody*) is derived from the original work by Stevens et al.[32] in 1937. As its frequency represents a logarithmic approximation, different formulae have been used to convert the Mel scale $Mel(f)$ into linear frequencies, f . As used by Paul Hill [32], a popular formula for frequency conversion to Mel is

$$Mel(f) = 2595 \cdot \log \left(1 + \frac{f}{700} \right). \quad (2.1)$$

The Bark scale is the scale presented after the discovery of Critical Bands in 1940. The most

commonly used Bark Scale is the one defined by Zwicker [12]. Table 2.1 clarifies the proposed Bark scale adapted by Paul Hill [32], created by [12].

Table 2.1: Bark Scale, according to [12].

Bark	Central Freq. (Hz)	Freq. Range (Hz)
1	50	20–100
2	150	100–200
3	250	200–300
4	350	300–400
5	450	400–510
6	570	510–630
7	700	630–770
8	840	770–920
9	1000	920–1080
10	1170	1080–1270
11	1370	1270–1480
12	1600	1480–1720
13	1850	1720–2000
14	2150	2000–2320
15	2500	2320–2700
16	2900	2700–3150
17	3400	3150–3700
18	4000	3700–4400
19	4800	4400–5300
20	5800	5300–6400
21	7000	6400–7700
22	8500	7700–9500
23	10500	9500–12,000
24	13500	12,000–15,500

Lastly, another common approach for critical band modelling is the Equivalent Rectangular Bandwidth (ERB). This methodology, where $ERB(f)$ is the ERB bandwidth at frequency f was updated in 1990 by Glasberg and Moore [34], becoming the standard formula used for conversion [32], given by:

$$ERB(f) = 24.7 \cdot \left(4.37 \cdot \frac{f}{1000} + 1 \right). \quad (2.2)$$

2.3 Digital Signal Processing of Speech

The physiology and anatomy of the auditory system and the system associated with speech production allow a deeper understanding of the reasoning behind speech processing. This section describes basic concepts, algorithms, and breakthroughs in the field of speech processing, with digital signal processing techniques. The first concept to be presented is the Source-Filter Model.

2.3.1 Source-Filter Model

The Source-Filter Model was theorized by Gunnar Fant in 1960 [35]. Source-Filter theory describes speech production as a two-stage process. First, we generate sound with our sound source – emulating our phonation process, with its own spectral shape and structure. Second, the source is filtered – emulating our articulation system. These two components are considered independent of each other [35]. According to this theory, the sound source can be either periodic or non-periodic, in accordance with the type of speech trying to emulate. Despite this model being a simple approximation, its simplicity allowed for a convenient representation of speech production in general. This paradigm also promotes a fundamental basis for important voice signal processing tools: Linear Predictive Coding (LPC) [36] and Cepstral Analysis [37].

2.3.2 Linear Predictive Coding

LPC is a method used in audio digital processing and speech processing for representing the spectral envelope of digital signal of speech in a compact way, using the information of a linear predictive model. It is a compression method that models the process of speech production. LPC models this process as a linear sum of earlier samples using a digital filter, where the input is an excitement signal. As seen in Fig. 2.4, LPC acts as a filter for the input $x(n)$, capturing the signal residue $e(n)$. The residue may be used for the filter reconstruction $P(z)$, to reconstruct the original signal [2].

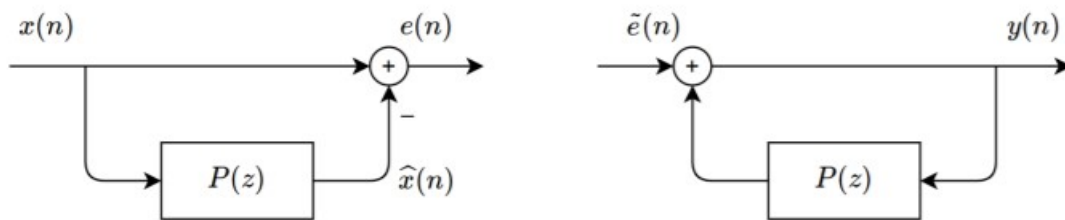


Figure 2.4: Linear Predictive Coding system overview. Adapted from [2].

LPC has had extended versions, such as Code Excited Linear Predictive (CELP) and Vector Selectable Excited Linear Predictive (VSELP) algorithms, which were developed in the mid 1980s and used commercially for audio and music coding in the later part of that decade [2].

2.3.3 Cepstral Analysis

Cepstrum analysis is a tool for the detection of periodicity in a frequency spectrum, and thus far appears to have been used in speech analysis to determine the pitch of the voice and related questions [37, 3]. It is also used to derive the spectral envelope. Cepstral analysis deals with a signal in the frequency domain as if it were a time signal. Using the log magnitude scale for a frequency signal, the inverse Fourier transform is applied, and, later, the spectral domain is converted to the new domain - the cepstrum. The power cepstrum is particularly utilized in signal processing [2], and it is declared according to Eq. 2.3, where C represents the power cepstrum, $x(t)$ represents a signal in the time domain, and $\mathcal{F}$ and $\mathcal{F}^{-1}$ represent, respectively, the Fourier transform and its inverse [2].

$$C = |\mathcal{F}^{-1} \log(|\mathcal{F}x(t)|^2)|^2 \quad (2.3)$$

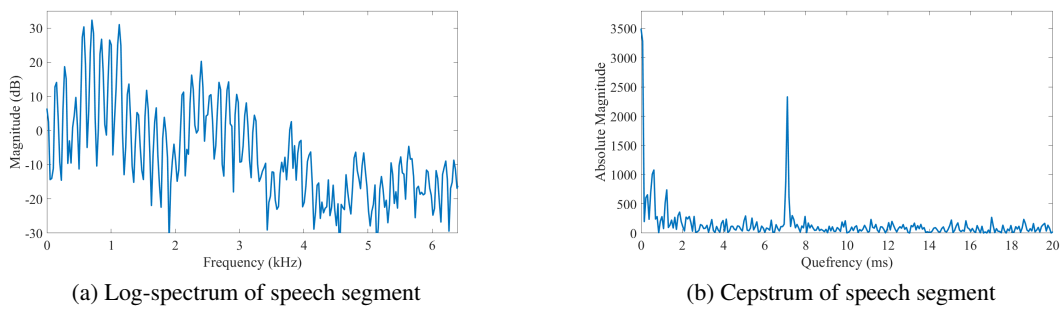


Figure 2.5: Log-spectrum of a signal (left) and the respective cepstrum (right). The quefrequency coefficients represent the signal formants, and the isolated peak represents the fundamental period. Adapted from [2]

The creation of the cepstrum allows the decomposition of the spectral envelope structure. The repetitive fundamental frequency and the respective harmonics create a pattern in the frequency domain. The use of the cepstrum domain, as seen in Fig. 2.6, converts the frequency pattern to a peak. However, the signal formant shape is converted in multiple quefrequency coefficients [2].

2.3.4 Mel-Frequency Cepstral Coefficients

Mel Frequency Cepstral Coefficients (MFCCs) represent a set of features of a signal, which describe the spectral envelope given a frame of specified size. It is often used to find timber, and are a widely used technique in speech signal processing. They were introduced by Davis and Mermelstein in the 1980s.

MFCC's computation can be summarized in a seven-step process [38]. Assuming that the signal is already in digital format, first, but not necessarily, the signal may be pre-emphasized [38], altering the slope of the spectrum to increase the energy at high frequencies. The following step is a common process in speech processing: to apply a window function to a frame of a given

size. This window function is usually a Hamming window [38]. This window reduces the effects of strong edges applied to the edges of the frame, resulting in a smooth spectral representation. The third step is to apply the Discrete Fourier Transform (DFT) and, later, a Mel filterbank is applied to measure energy in different frequency bands [4]. The fifth step is the logarithm computation of the obtained energy in the corresponding domain. Later, the Inverse DFT (IDFT) is applied to the log-power spectrum, creating the so called cepstral domain [38]. At first sight, the computation of the cepstral domain features may seem pointless. Despite this, the idea behind this technique is that the features are less correlated [38], being fit for statistical-based systems [9].

The process of computation allows less-correlated features to be extracted, allowing different levels of information to be managed. This process can be over-viewed in a simpler fashion in figure 2.6.

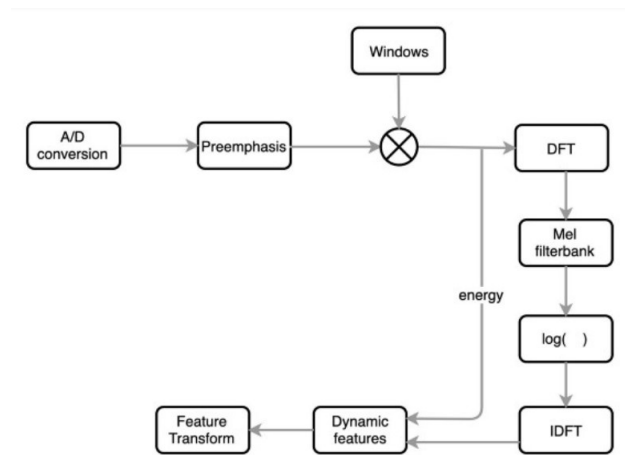


Figure 2.6: Block diagram for MFCC extraction. Adapted from [4]

2.3.5 Mel Cepstral Coefficients

Similar to the MFCC features, Mel Cepstral Coefficients have a frequency resolution that is similar to that of the human ear [39], given the derived implementation of the perceptual Mel scale. As stated by Toshiaki Fukada [39], to approximate the Mel scale, an all-pass transform is applied. In contrast, MFCCs use filterbanks to adjust the scale, as already discussed in the previous section [38].

Chapter 3

Machine Learning

This chapter overviews the fundamental knowledge of machine learning, as speech processing techniques have been explained in the previous chapter. This chapter attempts to overview machine learning models and relevant techniques in order to understand Generative Adversarial Networks and their application in WHSP2SPCH conversion.

3.1 Basics of Machine Learning

Firstly, to understand the context of this chapter, a clarification is needed for Machine Learning (ML), as it is not a synonym of Artificial Intelligence (AI). [6].

AI is the concept of building an intelligent machine, realized on software and hardware, that can possess human-like intelligence [6]. This broad term allows us to assert that AI is concerned with the goal of attaining cognitive intelligence in machines and not on low-level information processing.

ML is a subset of AI, and a scientific field rooted in statistics, computer science, information theory, signal processing, and automatic control [6]. This area embodies the study of computer algorithms that can be automatically improved through experience and using data.

3.1.1 Learning Systems

The learning systems of a machine learning algorithms can be divided in three main parts [6]:

- Decision process: algorithms are used, in a simplified and generalized manner, to make a prediction or solve a classification problem. Based on input data, labelled or unlabeled, algorithms produce estimates as output.
- Error function: a function that evaluates the prediction of the model. In the case of known examples being fed into the system, this function is utilized to measure the accuracy of the model.
- Model optimization process: weights are adjusted to reduce the discrepancy between the known example and the model estimate. The algorithm will repeat this evaluation and

optimize this process, updating weights autonomously until a threshold of accuracy has been met.

3.2 Machine Learning Categories

3.2.1 Supervised Learning

Supervised learning is an approach that attempts to solve problems that involve using a model to learn a mapping function between input examples and target variables, like $(y_n, x_n), n = 1, 2, \dots, N$, where $\bar{x} = [x_1 \ x_2 \ \dots \ x_N]^T$ usually is a multidimensional vector of size N . Considering these variables to analyze the data, we will represent the available data in pairs of observations [6]. These observations relay information about features - input variables to the system, x_n , and the variable y_n , known as the output, target or label variable [6].

Two standard problems can be solved in supervised learning, according to the available data.

- **Classification:** a supervised learning problem that involves predicting a class label, assigning a pattern to one of the set of possible classes.
- **Regression:** a supervised learning problem that involves predicting non-discrete values in an interval in the real axis or in a region of the complex numbers' plane.

3.2.2 Unsupervised Learning

Unlike supervised learning, where labels are provided in the observations, unsupervised learning describes the class of problems where only input variables are available. Thus, in this scenario, only x_n is given, and no y_n labels exist. As such, unsupervised learning does not have a "teacher" correcting the model. The goal in unsupervised learning is to find the structure that underlies the given set of data. One of the most important types of unsupervised learning is clustering [6].

Data Clustering [6] organizes a collection of data into a number of clusters or data segments. Referencing to a specific algorithm, as an example, K-means is used practically to organize observations into clusters by minimizing the distance between observations within a cluster and maximizing the distance between observations across clusters. The following steps showcase the logical approach to solve clustering problems:

1. Creating a model by making an assumption on the input data.
2. Selecting the objective function or goal of the clustering.
3. Evaluating one or more algorithms to optimize the objective function.

Another set of unsupervised learning problems are dimensional reduction, using algorithms such as Linear Discriminant Analysis (LDA) or Principal Component Analysis (PCA) to transform a D -dimensional feature space to a D' -dimensional feature space, where D is higher than D' .

3.2.3 Semi-supervised Learning

Semi-supervised learning lies in between supervised and unsupervised learning. It is used to build models in a context where a dataset has incomplete labels y_n . The existence of unlabeled patterns can assist in a task, since it can reveal additional structure of the input data that can be efficiently utilized [6]. There is no need for this dissertation's purpose to dive into detail for semi-supervised learning, only a brief explanation is needed to understand typical scenarios in machine learning problems.

3.3 Model Taxonomy

Two different approaches need to be distinguished in machine learning. This section attempts to explain the differences between discriminative and generative models, as well as their respective necessity in order to solve different problems.

3.3.1 Discriminative Models

In a supervised learning context, we may be interested in developing a model to predict a class label, C_k , given an example of input variables, x - a classification task. This path of modeling is also known as discriminative. Discriminative models directly infer the *a posteriori* probability of $p(C_k|x)$, where $C_k, k = 0, 1, \dots, C$ represent k -th class for the C classes in the classification task. This method then uses this probability to make the class assignment. The detail to retain from this explanation is that discriminative models bypass the need to explicitly model the input data distribution [6].

3.3.2 Generative Models

Generative models retain information about the distribution of input data, unlike discriminative models. These models attempt to sufficiently summarize the data distribution, and then use this information to generate new variables that plausibly fit into the distribution of the input variable. Thus, for a given domain, the generative model attempts to replicate indistinguishable outputs [8]. Later in the chapter, this topic will be explained in detail.

3.4 Neural Networks

Neural Networks (NNs) have a long history that combines multiple areas of study. This history goes back to the first attempts to understand how the human brain works [6].

Frank Rosenblatt introduced, in 1957, the perceptron [40]. He used a single neuron and adopted a rule that can learn to separate data that belongs to one of two linearly separable classes. Thus, he built a pattern recognition system [6]. This model was influenced and inspired by Warren McCulloch and Walter Pitts and their respective work in a computational model for the basic neuron and its results that tie neurophysiology with mathematical logic [6]. This model introduced

the first computational representation of a neuron. This gave birth to what is commonly known as neural networks.

Neural networks are learning machines consisting of neurons. Neurons are the fundamental building blocks for the NN. They are connected in a layered fashion, and their learning is achieved by adjusting weights to minimize a preselected cost function [6].

3.4.1 Perceptron

The perceptron deals with the linearly separable two-class classification problems. Consider (y_n, x_n) , $n = 1, 2, \dots, N$, where $\bar{x} = [x_1 \ x_2 \ \dots \ x_N]^T$ is our feature vector, and the definition for our learnable weights is $\bar{w} = [w_1 \ w_2 \ \dots \ w_N]^T$. Also, a bias b is included. The bias is applied to the linear combination of the input.

In the perceptron, illustrated in Fig. 3.1, each feature is multiplied by the respective weights $w_1, w_2, \dots, w_N$, and then the bias term b is added. The linear outcome of this operation then goes through a nonlinear function, f , known as the activation function.

$$y = f\left(b + \sum_{i=1}^N w_i x_i\right) = f\left(b + \bar{w}^T \bar{x}\right) \quad (3.1)$$

Depending on the form of the non-linearity of the activation function, different system outputs are possible. For simplicity, we will introduce $\text{sign}(x)$, the sign function, where

$$\text{sign}(x) = \begin{cases} 1 & \text{if } x \geq 0 \\ -1 & \text{if } x < 0 \end{cases} \quad (3.2)$$

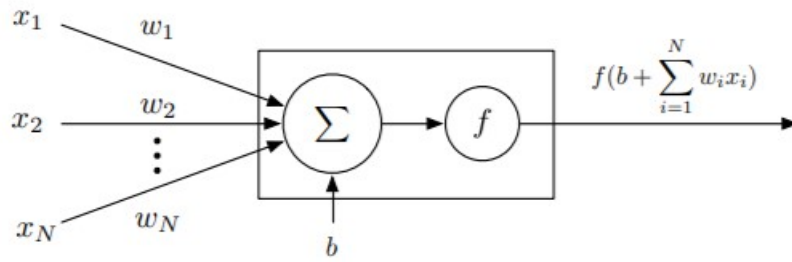


Figure 3.1: Perceptron's system processing. Inspired by [5]

The convergence for this model is guaranteed only for the problems of linearly separable two-class classification. Linear separability is defined by the existence of at least one hyperplane in the Euclidean space, where the points are defined that separate the two sets [5, 6].

Fig. 3.1 illustrates the perceptron's system from a high-level perspective of the system.

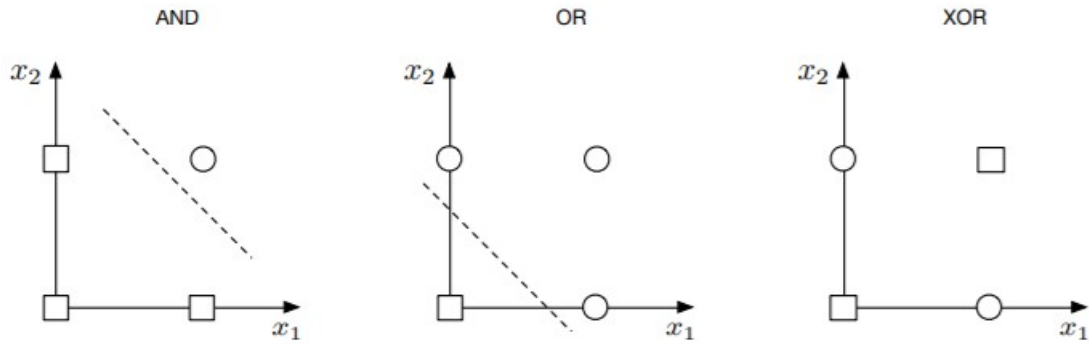


Figure 3.2: Linear separability in Neural Network. Inspired by [5]

As Fig. 3.2 describes, the AND/OR functions can be resolved by the use of the perceptron for a two-class classification. This is due to linear separability, that does not occur in the XOR functions. To classify non-linearly separable data, the perceptron (neuron) needs improvement.

3.4.2 Feed-Forward Neural Network

Representing the perceptron model as a neuron, the development of Neural Network techniques allows a solution to two-class nonlinear separability. A generalization from two class classification can be made for a multi-class problem [6].

In order to classify non-linearly separable data, neural network models can be implemented. Given a certain feature space, a single neuron is associated with a hyperplane:

$$H : w_1x_1 + w_2x_2 + \dots + w_lx_l + w_0 = 0, \quad (3.3)$$

Given this context, the previously used bias term b has changed, where $b = w_0$, where $\bar{w} = [w_0 \ w_1 \ w_2 \ \dots \ w_l]^T$ and feature space of dimension $L = l + 1$, given by $\bar{x} = [x_0 \ x_1 \ x_2 \ \dots \ x_l]^T$. As the previous bias term is multiplied by 1, $x_0 = 1$, thus $w_0x_0 = w_0 = b$. This is the standard notation used in machine learning, to the extent of the author's knowledge.

Classification is performed via the nonlinearity, which fires a one or stays at -1 (or zero, dependent on the activation function), depending on which side of H a point lies [6].

Building a multi-layer neural network displays neurons in layers, connecting parallel neurons - nodes - to a branch of new neurons - represented as a layer. Referencing to case B in Fig. 3.3, outputs of the neurons of the first layer - first hidden layer - become inputs which feed the neurons of a second layer - second hidden layer.

The feed-forward neural network was the first and simplest type of neural network. In this network, the information moves only from the input layer directly through a number of hidden layers to the output layer without looping.

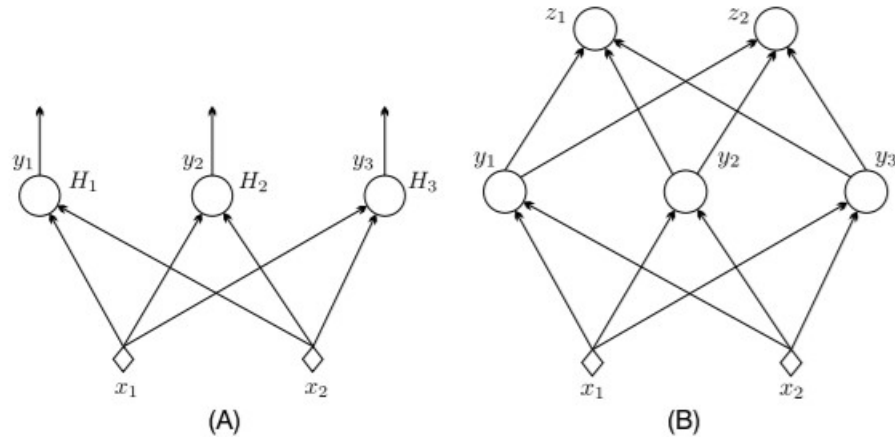


Figure 3.3: Adding layers to the perceptron model, building a multi-layer neural network. Adapted from [6]

3.4.3 Convolutional Neural Network

Feed-forward neural networks with only one output allow a solution for two-class nonlinear separability. A generalization can be made for multi-class problems with several outputs, one for each class. Despite this, simple feed-forward neural networks are not able to represent every mapping functions. Multi-layer feed-forward neural networks are a class of universal approximators [41]. However, the mapping function may require an high number of parameters, making the training of the network hard. Understanding this hurdle is necessary. We cannot, in practice, feed the input of a neural network directly with raw data as the tasks at hand tend to become unmanageable.

Convolutional Neural Networks (CNN) were invented in the late 1980s, when the feature generation phase was integrated as part of the training of a neural network [6]. Their purpose is to combine local features in order to obtain features with a higher level of abstraction, learning the features from the data together with the parameters of the neural network and not independently. They extract information that quantifies correlations or other statistical dependencies that relate information to the raw-data. As stated by Theodoridis, "In this way, one can efficiently "encode" learning-related information that resides in the raw data [...]" [6]. Findings indicate that convolutional networks are able to learn generic feature extractors that can be used for different tasks [7].

The basic steps involved in convolutional networks are:

- The convolution step.
- The nonlinearity step.
- The pooling step.

The convolution step is given by the sharing concept, where, considering only a one-layer system, the first layer hidden nodes do not connect to all input nodes, but only to a neighborhood

of input nodes and the connection weights do not change when a different neighborhood region of the input is considered [5]. After this process of weight sharing, the nonlinearity step is introduced by the activation function, a non-linear function [6]. Currently, the rectified linear activation function, ReLU, seems to be the most popular [6]. Lastly, the pooling step takes place to reduce the dimensionality of each feature map array [6]. Fig. 3.4 illustrates the convolutional step and the pooling step. The nonlinearity step is represented by the connections in the hidden-layers.

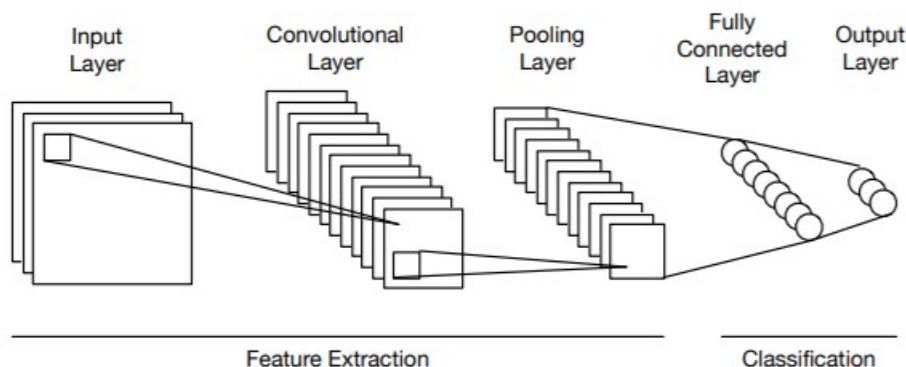


Figure 3.4: CNN for an RGB image as input with one convolutional layer and one pooling layer. Adapted from [5][7]

3.5 Generative Models

As already introduced, generative models attempt to sufficiently summarize the data distribution, and then use this information to generate new variables that plausibly fit into the distribution of the input variable. Despite this simple definition, generative model is a term used in many contexts [8]. In order to simplify the use of the term, a contextualization is needed. Generative models will be referred in this context as models that take a training set, consisting of samples drawn from a certain distribution - p_{data} - and learn to represent an estimate of that distribution somehow. This results in a model probability distribution - p_{model} - as used in "NIPS 2016 Tutorial: Generative Adversarial Networks", by Ian Goodfellow [8]. The attempt to model a probability distribution, given a training set, is known as the problem of density estimation, one of the most fundamental problems in statistics. It is a notoriously difficult problem due to the curse of dimensionality [42]. The curse of dimensionality is the problem described as the phenomena that arise in high-dimensional spaces that do not occur in low-dimensional ones. To contextualize, given two hyperplanes of different sizes, H_1 and H_2 , where $H_1 \in \mathbb{R}^{l_1} \cap H_2 \in \mathbb{R}^{l_2}, l_1 \gg l_2$, for H_1 the amount of data needed is exponentially higher in comparison to H_2 , in order to obtain approximately the same results.

3.5.1 Taxonomy of Generative Models

As noted, density estimation is one fundamental problem regarding generative learning. Implicit of explicit declarations of the probability distribution p_{model} is a notorious difference in different generative approaches, as observed in the Fig. 3.5. As proposed in the literature [8][43][44], we will focus on generative models that work via the principle of maximum likelihood and the models that do not by default can be made to do so [8].

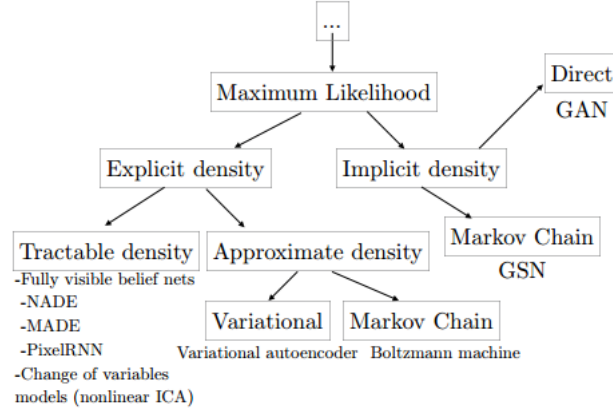


Figure 3.5: Taxonomy of generative models from Ian Goodfellow’s NIPS tutorial, 2016 [8]

3.5.2 Maximum Likelihood Estimation

The basic idea of maximum likelihood is to define a model that provides an estimate of a probability distribution, with respect to θ , where θ represents the matrix of adaptable weights to a function given by the form $y = \theta x$.

We are given a set of N observation, $X = [x_1 \ x_2 \ \dots \ x_N]^T$, drawn from a probability distribution. First, we assume that the joint probability density function of the observations is of a known parametric functional type, denoted as $p(X; \theta)$, where $\theta \in \mathbb{R}^K$ is unknown. Thus, the task is to estimate θ . This joint PDF is known as the likelihood function of θ with respect to the given set of observations X [6]. The maximum likelihood can be defined as

$$\theta_{ML} = \underset{\theta \in \beta \subset \mathbb{R}^K}{\operatorname{argmax}} p(X; \theta) \quad (3.4)$$

where, for simplification, $\beta \subset \mathbb{R}^K$ has no restriction. Thus, $\beta = \mathbb{R}^K$. Also, let us consider that the function enjoys a unique maximizer with respect to the parameter θ .

For the N statistically independent observations, the joint likelihood function is given by

$$\theta_{ML} = \underset{\theta}{\operatorname{argmax}} \prod_i^N p(x_i; \theta) \quad (3.5)$$

Thus, in this space, parameter estimation is simpler to do in the logarithmic space, where we have a sum rather than a product over examples. This sum simplifies the algebraic expressions for the

derivatives of the likelihood with respect to the models [8]. The maximum log-likelihood is given

$$\theta_{ML} = \operatorname{argmax}_{\theta} \sum_i^N \log p(x_i; \theta) \quad (3.6)$$

Finally, considering that generative models work by maximum likelihood, we can compare several models by contrasting the ways that they compute either the likelihood and its gradients, or approximations to these quantities [8]. Therefore, every model can be parametrized by θ_{ML}^* , given by

$$\theta_{ML}^* = \operatorname{argmax}_{\theta} E_{X \sim p_{data}} \log p(X; \theta) \quad (3.7)$$

where $E_{X \sim p_{data}}$ represents the expected value of X assuming a probability density function described by p_{data} .

3.5.3 Fully Visible Belief Networks

Fully Visible Belief Networks (FVBN) are models that define $p_{model}(X; \theta)$ explicitly, making the maximization of the likelihood easy to perform [5]. The main issue is related to the design of a model that is computationally tractable. This method uses the chain rule of probability to decompose a probability distribution over an n -dimensional vector x into a product of one-dimensional probability distributions [8], by

$$p_{model}(X) = \prod_i^N p_{model}(x_i | x_1, \dots, x_{i-1}; \theta) \quad (3.8)$$

3.5.4 Variational Autoencoders

Autoencoders consist of two parts, the encoder and the decoder [6], and they can be used to generate data [45]. A variational autoencoder (VAE) is an artificial neural network (ANN) architecture introduced by Diederik P. Kingma and Max Welling, that belongs to the families of probabilistic graphical models and variational Bayesian methods. VAEs are characterized by one loss function, unlike generative adversarial networks (GANs). VAEs belong to a class of generative models that provide a computationally intractable density function, but the corresponding likelihood is maximized using deterministic approximations, in contrast with FVBNs [8]. The VAEs loss function is characterized by

$$L(X; w) \leq \log p_{model}(X; w) \quad (3.9)$$

The main disadvantage in VAEs is that the gap between L and the true likelihood can be such that the model learns a distribution different from p_{model} [8].

3.5.5 Generative Adversarial Networks

Machine Learning has had multiple breakthroughs in research throughout the years. This field is rooted in statistics, computer science, information theory, signal processing, and automatic control. Large data sets were gradually created, and the computer power became affordable to allow the use of more complex models [6].

A Generative Adversarial Network (GAN) is a class of machine learning frameworks designed by Ian Goodfellow and his colleagues in 2014 [43]. Two neural networks contest with each other in the form of a zero-sum game, where one agent's gain is another agent's loss. Originally proposed as a form of generative model for unsupervised learning, GANs have also proven useful for semi-supervised learning, fully supervised learning, and reinforcement learning [6].

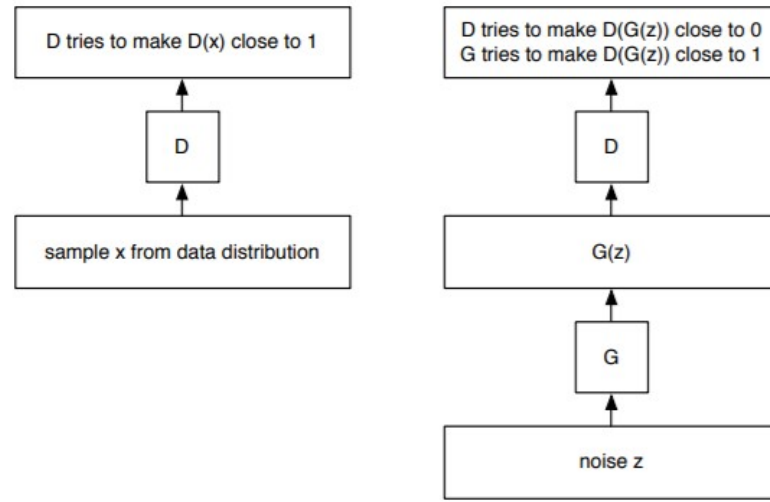


Figure 3.6: Block diagram for GAN and respective framework. By [5][8]

GANs are unsupervised learning algorithms that use a supervised loss as part of the training. The data comes in with no labels, and we are not trying to generalize any kind of prediction to new data. The GAN sets up a supervised learning problem in order to do unsupervised learning. It generates fake / random looking data, and tries to determine if a sample is generated fake data or real data.

As we can see from Fig. 3.6, G and D are the functional blocks in a GAN architecture, and generally they are neural networks [8]. Different approaches have been used, but in particular we are interested in neural networks or deep neural networks. These approaches have become widely spread for high resolution images and have come in contact with Voice Conversion (VC).

The loss function of D [8] is described as

$$J^{(D)}(\theta^{(D)}, \theta^{(G)}) = -\frac{1}{2}E_{x \sim p_{data}} \log(D(x)) - \frac{1}{2}E_z \log(1 - D(G(z))) \quad (3.10)$$

where this function represents the standard cross-entropy cost that is minimized when training a standard binary classifier with a sigmoid output. The only difference is that the classifier is trained

on two separate batches of data; one coming from the dataset, where the label is 1 for all examples, and one coming from the generator, where the label is 0 for all examples [8].

The loss function of G can be described, considering the same notation [8], by

$$J^{(G)}(\theta^{(D)}, \theta^{(G)}) = -J^{(D)}(\theta^{(D)}, \theta^{(G)}) \quad (3.11)$$

In this scenario, we have the case of a zero-sum game, also called *minimax*, where the solution involves minimization of the generator loss function and maximization of the discriminator loss function. This scenario is described by

$$\theta^{(G)} = \operatorname{argmin}_{\theta^{(G)}} \max_{\theta^{(D)}} V(\theta^{(D)}, \theta^{(G)}) \quad (3.12)$$

where $V(\theta^{(D)}, \theta^{(G)}) = -J^{(D)}(\theta^{(D)}, \theta^{(G)})$. This *minimax* game is mostly of interest because it is easily amenable to theoretical analysis [8][43] but does not perform especially well in practice. In this approach, the discriminator minimizes a cross-entropy, but the generator maximizes the same cross-entropy. This is unfortunate for the generator, because when the discriminator successfully rejects generator samples with high confidence, the generator's gradient vanishes. To solve this problem, one approach is to continue to use cross-entropy minimization for the generator. The cost for the generator then becomes a non-saturating game [8], given by

$$J^{(G)}(\theta^{(D)}, \theta^{(G)}) = -\frac{1}{2} E_z \log(D(G(z))) \quad (3.13)$$

Chapter 4

State of the Art

This chapter discusses the state of the art in WHSP2SPCH conversion, more specifically, in generative adversarial networks for spectral envelope mapping and fundamental frequency (F_0) prediction.

4.1 Problem Formulation

WHSP2SPCH conversion has been an area of study with a long history. The traditional rule-based approaches use speech processing techniques [21, 32] that rely on modifying speech parameters according to empirical rules. These methods can obtain high quality spectral details and lead to good naturalness in the speech synthesis [46]. However, rule-based methods have been deemed unsatisfactory in the literature [14], when compared to statistical models. Functions that represent speech and its spectral information are usually difficult to be accurately extracted [46]. As such, in voice conversion, rule-based models are reported to vary in performance, as mapping functions may be accurately extracted between some speaker pairs, but also to show unsatisfactory results between other speaker pairs [46]. For this reason, statistical-based models have gained popularity [46]. Moreover, for voice conversion, the improvement of statistical models with the introduction of deep learning have had major impact on many real-life applications, such as personalized speech synthesis and communication aids for the speech-impaired [47].

In regards to the statistical-based models, the recent use of GAN-based models in WHSP2SPCH conversion has shown great results [9, 11, 10], and recently developed techniques demonstrate their abilities to discover cross-domain relationships [11, 48] between whispered speech and normal speech. The prominent problems and limitations of GAN-based models for WHSP2SPCH conversion in the literature are as follows:

- Fundamental Frequency (F_0) Prediction [9].
- Spectral Envelope Mapping (using the standard MFCCs or MCCs) [49].
- GAN optimization and stability [11].

4.2 GAN-Based Methods For Speech Processing

In order to obtain more natural-sounding speech, a valid option is to implement GAN-based methods, in an attempt to predict the fundamental frequency for normal speech and map the cross-domain relationship between whisper and normal speech. This machine learning methodology has been proven to predict the fundamental frequency with high quality [9] and to learn cross-domain relationships without a pair-labelling dataset [10, 9].

4.2.1 Fundamental Frequency Prediction

Fundamental frequency prediction using GANs can be described as an attempt to emulate the probability distribution for F_0 in natural speech[10]. Given this scenario, F_0 is modeled, and its prediction affects spectral feature mapping, adding non-linear noise to the features [9]. This affects the overall performance. Thus, solutions are proposed using state of the art architectures [9][49]. Using state of the art GAN techniques, the distributions of the generated F_0 closely follow the distribution of the F_0 corresponding to the natural speech signal [10, 9].

4.2.2 Spectral Envelope Mapping

Spectral envelope is mapped using MFCC [49] or MCC [9], depending on the context. Also, the spectrogram has been used to represent the spectral envelope [18]. A further study needs to be made to assert differences in evaluation based on different architectures. Despite this, the spectral envelope mapping in current architectures tends to demonstrate good results [9, 10].

4.3 Reference System Architectures

4.3.1 Cycle-GAN

Cycle-GAN implements two different GANs, each constituted by a generator and a discriminator [9]. One GAN model is tasked with the Mel Cepstral Coefficients (MCC) mapping between whisper and normal speech. The other model is tasked with F_0 prediction. The proposed architecture trains the first model to map the spectral envelope of whisper to normal speech, in order to generate synthetic normal speech, based on the input whispered speech. Later, the synthetic normal speech is used to train the second model for F_0 prediction. In the proposed method, F_0 is highly dependent on the MCC model. This dependency generates additional non-linear noise in F_0 prediction.

Using Cycle-GAN for MCC mapping, we have two generators. One generator maps the cepstral features of whisper to normal speech, and another maps the inverse cepstral features. Two Discriminators are also considered. Figure 4.1 illustrates the proposed architecture in the referenced paper [9].

The nomenclatures used for the generators and discriminators are based in domain X , representing whispered speech MCCs and Y , representing normal speech MCCs. $G_{X \rightarrow Y}$ is the

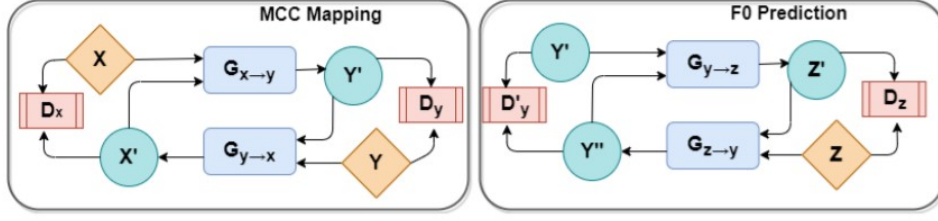


Figure 4.1: Block diagram for Cycle-GAN, as proposed by [9]

generator that converts X domain MCCs to the Y domain, represented by Y' . The same logic is applied to $G_{Y \rightarrow X}$.

The discriminators attempt to distinguish real domain MCCs from fake generated MCCs. As such, D_X discriminates real X domain MCCs from the generated by X' . The same logic applies to D_Y .

The model is trained for MCC mapping by learning to generate high quality Y' domain (normal speech) MCCs, using $G_{X \rightarrow Y}$, in an attempt to fool the discriminator D_Y into considering that the speech is real instead of synthetic.

For fundamental frequency prediction, Z domain is introduced. The trained MCC model generates Y' by converting whispered speech into normal speech, using $G_{X \rightarrow Y}$. The synthetic MCCs Y' are now converted to a one-dimensional feature, the fundamental frequency, with the generator $G_{Y \rightarrow Z}$. This generator attempts to map MCCs to a single feature. The same logic applies to the $G_{Z \rightarrow Y}$, that maps a fundamental frequency to the respective normal speech MCCs. The discriminator D_Z attempts to distinguish real fundamental frequency samples, Z , from fake generated ones, Z' . Additionally, D_Y attempts to distinguish the Y' domain from the Y'' domain. The Y'' domain represents the synthetic MCCs derived from the generator $G_{Z \rightarrow Y}$. The sequence of operation relies on the previous synthesis of normal speech Y' , as training is done separately for F_0 and MCC remapping. As such, after training $G_{X \rightarrow Y}$, this generator is used to generate Y' , that acts as input for fundamental frequency prediction [9].

4.3.2 CinC-GAN

The referenced paper [9] proposes a different training method, namely, CinC-GAN, an advanced version of cycle-GAN, for WHSP2SPCH conversion. This creates an inner cycle for acoustic mapping of cepstral features, and an outer cycle for F_0 prediction. The outer cycle relies on cepstral features of converted normal speech and input whisper speech as well (i.e., joint training). The results obtained demonstrate that CinC-GAN significantly outperforms cycle-GAN for seen and unseen speakers, respectively, on gender-specific tasks [9]. Figure 4.2 illustrates the proposed architecture in the referenced paper, according to the same variables as the cycle-GAN [9]. The same nomenclatures are applied in CinC-GAN, as in cycleGAN 4.1. The sequence of operations rely on in the idea that, each iteration, whispered speech X is converted into the synthetic normal speech Y' . For each Y' , the generator $G_{Y \rightarrow Z}$ converts the synthetic normal speech MCCs into a

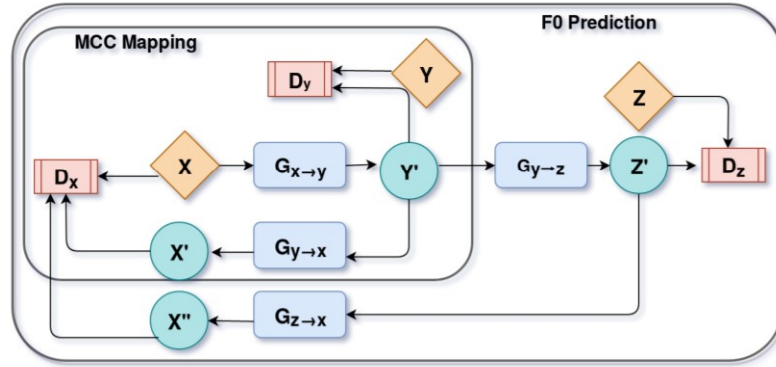


Figure 4.2: Block diagram for CinC-GAN, as proposed by [9]

one-dimensional fundamental frequency feature, Z' . As both systems are jointly trained, the total loss of the system is dependent on both MCC mapping and fundamental frequency prediction. This allows CinC-GAN to function similarly to cycleGAN in MCC mapping, but substantially improve results in fundamental frequency prediction [9].

4.3.3 Disco-GAN

The training method of cycle-GAN and disco-GAN is very similar. These architectures only differ in terms of their loss functions [10]. In the cycleGAN, the least mean square loss (i.e., MSE) plays the role of adversarial loss. However, Disco-GAN has binary cross-entropy loss (BCE) as the adversarial loss [10]. Moreover, the DiscoGAN has two different reconstruction losses, which are regularized by the MSE loss [10].

Given two domains, namely, whispered speech (W) and the normal speech (S), features are mapped (X_s and X_w) for the two different speech types. The model uses two generators (G_{ws} and G_{sw}) along with two discriminators (D_w and D_s). G_{ws} converts X_w into X_{ws} (converted features of the normal speech) such that X_{ws} is indistinguishable from the true samples X_s , and similarly for G_{sw} . The discriminator D_s attempts to distinguish between X_s and X_{ws} . D_w performs an analogous operation for the X_w .

Figure 4.3 illustrates the proposed architecture in the referenced paper [10].

DiscoGAN behaves similarly to the cycleGAN. DiscoGAN relies on the reconstruction losses to convert whispered speech MCCs into synthetic normal speech MCCs, X_{ws} , by using the generator G_{ws} . Moreover, the synthetic normal speech X_{ws} is reconverted into the whispered speech MCCs domain, X_{wsW} . The main idea in the reconstruction loss is to maximize similarity between X_w and X_{wsW} . This loss is similar to the cycle-consistency loss function of the cycleGAN [10]. Despite this, there is only one cycle-consistency loss, while discoGAN presents two reconstruction losses (one for each domain).

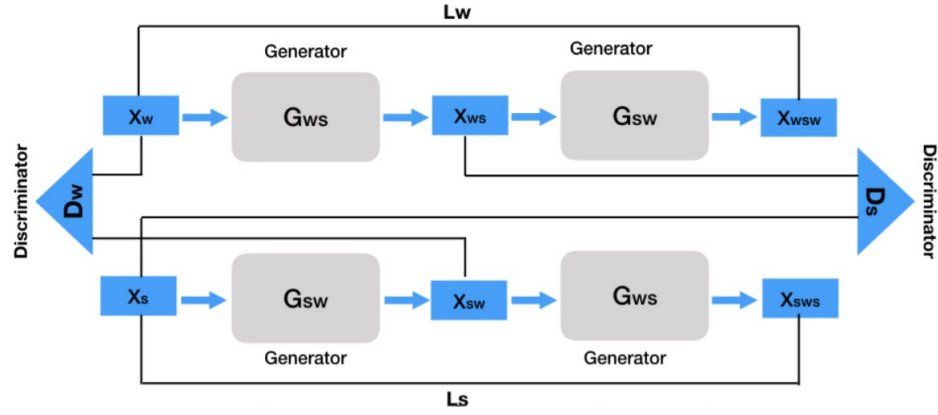


Figure 4.3: Block diagram for Disco-GAN, as proposed by [10]

4.3.4 MMSE-discoGAN

MMSE-discoGAN is used to learn the mapping between the cepstral features corresponding to the whispered and the normal speech, and the mapping between the converted cepstral features and the corresponding F_0 of the normal speech. Figure 4.4 illustrates the proposed architecture in the referenced paper [11].

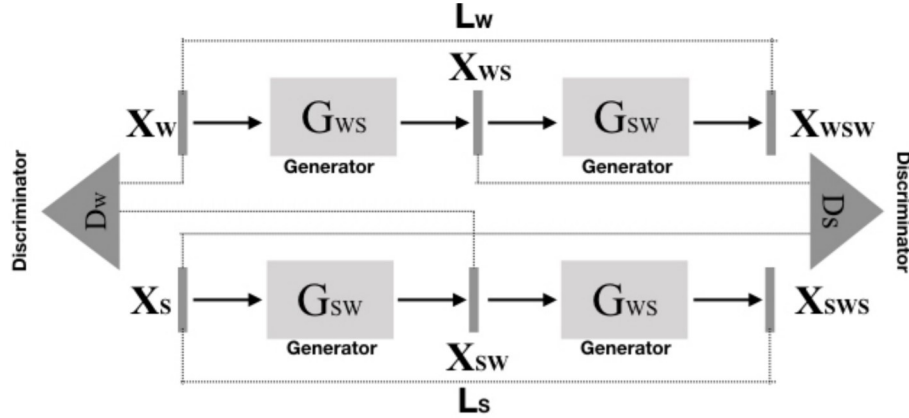


Figure 4.4: Block diagram for MMSE-discoGAN, as proposed by [11]

4.4 Other System Architectures

DNN-based systems are also relevant, as these systems can be converted into GAN-based models. According to SEQ2SEQ-DNN [50], this architecture extracts (a) the spectral envelope, the aperiodic component and F_0 for normal speech, and (b) the spectral envelope of whispered speech [50]. Later, both the spectral envelope are transformed in MFCCs. The sequence-to-sequence framework (SEQ2SEQ) is trained for mapping the relationship between the MFCCs of a whisper

and those of normal speech [50]. After training, these MFCCs are used to train two DBLSTM models for estimating F0 and the aperiodic component, respectively [51].

According to the DNN-MCC-F0[51], most of the WHSP2SPCH algorithms based on DNN directly use the whisper spectral envelope as the mapping object for the voiced speech version. The main problem with this method is that it can not exploit the whisper characteristic, as the spectral envelope of the whisper is very smooth due to the lack of vocal-cord vibrating [51]. As such, DNN-MCC-F0 compresses the spectral envelope to obtain low dimensional features. The relationship of the whisper and the normal is thereafter modeled in the Mel Cepstral Coefficients (MCC) feature space [51].

Lastly, MSpeC-Net [52] tackles Non-Audible Murmur (NAM) and corresponding conversions NAM-to-whisper (NAM2WHSP) and NAM-to-speech (NAM2SPCH). The difference in the acoustic characteristics between NAM and natural speech is very high [52]. MSpeC-Net solves the cross-domain speech conversion task using simple domain-specific autoencoders. The autoencoder is used to get an internal representation of features, which is known as latent representation [52].

Chapter 5

Methods

The DyNaVoiceR speech processing and synthesis framework currently depend on the following procedures:

- Spectral envelope mapping, that is being currently tackled in this dissertation.
- Fundamental frequency F_0 and relative phase delay (NRD) synthesis [53], based in template models for synthesis.

The present dissertation focuses on spectral envelope mapping, according to GAN-based methods for WHSP2SPCH conversion. This is the process to convert an input whispered speech spectral envelope into the domain-correspondent normal speech spectral envelope. F_0 and NRDs are approached by parametric modeling, based on template models that were previously retrieved and stored. Fig. 5.1 illustrates the proposed system.

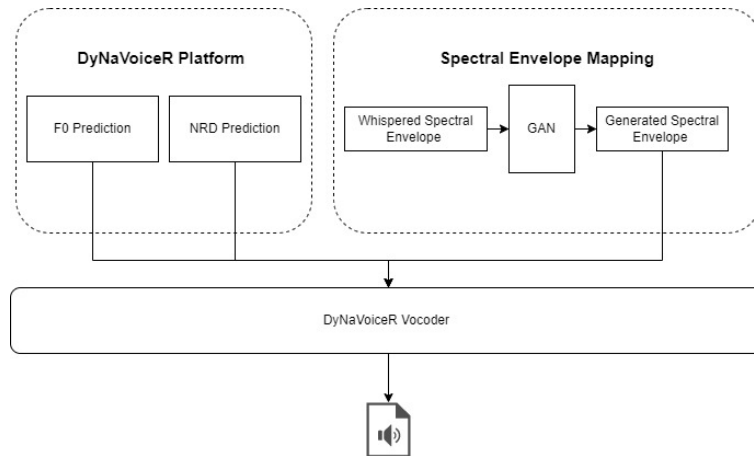


Figure 5.1: Block diagram of the DyNaVoiceR platform and proposed GAN-based spectral envelope mapping

This chapter details the used dataset, the process to retrieve spectral envelope models and obtain cepstral features. Moreover, the process to train and test the neural networks with cepstral features as inputs is explained. Lastly, the implemented architectures and validation metrics are discussed.

5.1 Dataset

The original dataset is the DyNaVoiceR dataset [13]. Original data contains 20 speakers, 10 feminine and 10 masculine. Speakers were instructed to reproduce specific tasks, as detailed in Appendix A. Each task is recorded and saved in WAV format, sampled at 44100 Hz, for two modes of communication: normal speech and whispered speech. From the obtained data, audio files are down-sampled to 22050 Hz. Moreover, the database includes manual phonetic annotations based in IPA for the European Portuguese Language. The phonetic annotations are an aggregation of formatted files utilizing TextGrid [54] format, that identify phonemes based on timestamps for a specific audio file.

From the DyNaVoiceR dataset, the data retrieved for the current project is as follows:

- Nine oral vowels uttered in the context of sustained speech.
- The same nine vowels uttered in European Portuguese words.

The retrieved oral vowels are indicated in table 5.1, with respective examples for phonetic clarification, as vowels constituted the main focus of the current project.

Table 5.1: Oral vowels and respective examples, based in IPA for the European Portuguese Language.

Vowels	Examples
/i/	<u>i</u> lha
/ê/	p <u>e</u> so
/é/	<u>e</u> la
/á/	<u>a</u> gua
/â/	<u>a</u> marelo
/ó/	<u>o</u> culos
/ô/	<u>o</u> vo
/u/	<u>u</u> va
/e/	sed <u>e</u>

In the next section a statistical analysis is made, considering the retrieved vowels, in order to distinguish phonemes according to frequency and ratio of occurrence. An occurrence is defined as a frame of data, according to the short-time Fourier transform (STFT) applied to the signal. Considering frequency of occurrence, two classes of vowels are later distinguished: vowels used to train the deep neural network and vowels used to explore zero-shot learning in the GAN-based system.

5.1.1 Descriptive Analysis

Firstly, the dataset was evaluated according to the frequency of each vowel, instanced according to number of frames per vowel, with no restrictions applied. Restrictions were defined as constrictions applied in data from the original dataset. Table 5.2 displays frequency for specified phoneme and in brackets the number of outliers is presented. The definition for outliers, in the given scenario, is as follows:

- An outlier is framed data that is classified as normal speech in the whispered dataset, or whispered speech in the normal dataset.

This approach was applied in order to avoid manual errors in the phonetic annotations of the dataset.

Table 5.2: Frequency table for used phonemes. In brackets, outliers are represented.

Vowels	Frequency in Whispered Speech	Frequency in Normal Speech
/â/	14808 (+5)	12421 (+53)
/á/	10651	9312 (+166)
/i/	7146 (+14)	6921 (+29)
/u/	6864 (+5)	6710 (+1)
/ó/	4890 (+4)	4505
/é/	4546	4396
/e/	4166 (+6)	4161
/ê/	4010	3937
/ô/	3927	3850

The outliers displayed in brackets, in table 5.2, constituted a maximum of 2.1% of occurrences for phoneme /á/. Considering their low representativeness, the outliers were thus excluded from the dataset.

From the retrieved data, balance with respect to the considered vowels was analyzed, assuming that a phoneme-balanced dataset has uniformly distributed phonemes. In table 5.3, the ratio of occurrence is presented.

Table 5.3: Ratio of occurrence for the nine vowels, considering whispered speech and normal speech. Information in brackets indicates the relation to the assumption of uniform distribution for a phoneme-balanced dataset.

Vowels	Ratio in Whispered Speech	Ratio in Normal Speech
/â/	24.27% (+12.0%)	20.10% (+10.99%)
/á/	17.46% (+7.2%)	16.57% (+5.45%)
/i/	11.71% (+1.2%)	12.31% (+1.20%)
/u/	11.25% (+0.02%)	11.94% (+0.83%)
/ó/	8.02% (-2.1%)	8.01% (-3.10%)
/ê/	7.45% (-4.0%)	7.82% (-3.29%)
/e/	6.83% (-4.6%)	7.40% (-3.70%)
/ê/	6.57% (-4.8%)	7.00% (-4.11%)
/ô/	6.44% (-5.0%)	6.85% (-4.26%)

The adopted structure separated vowels into two categories, as specified in Table 5.4 - zero-shot learning vowels and training vowels. Zero-shot learning is a problem rooted in machine learning-based systems, where in test scenarios, the system observes features from classes which were not observed during the training phase. This allowed the project to explore the architectures' capacity to generalize, empirically test model overfitting and a subjective listening evaluation for zero-shot learning.

Table 5.4: Vowels and respective purposes: training dataset and zero-shot learning dataset

Training Dataset	Example	Zero-Shot Dataset	Example
/i/	<u>i</u> lha	/ê/	p <u>e</u> so
/á/	<u>á</u> gua	/é/	<u>e</u> la
/â/	<u>a</u> marelo	/ó/	<u>o</u> culos
/u/	<u>u</u> va	/ô/	<u>o</u> vo
		/e/	sed <u>e</u>

Lastly, table 5.5 displays the number of gender-specific frames, with a ratio of approximately 50% for the four training vowels. The same applied for the case considering the nine initial vowels.

Table 5.5: Phoneme occurrences based on gender for the four training vowels.

Gender	Whispered Speech	Normal Speech	Total Frames
Feminine	29812	28102	57914
Masculine	31196	28111	59307

5.1.2 Training Data

Training data acted as input to update weights in the deep learning training step. As such, precautions were taken and vowels recorded in a read aloud context were explicitly disregarded for this step. Table 5.6 highlights respective training vowels and the number of occurrences, where occurrences are represented as the number of frames.

Table 5.6: Phonemes used in the training dataset, respective examples and the total number of occurrences displayed as $W + S$, where W represents whispered speech frames, and S represents normal speech frames.

Vowels	Examples	Occurrences
/i/	<u>i</u> lha	7146 + 6921
/á/	á <u>g</u> ua	10651 + 9312
/â/	<u>a</u> marelo	14808 + 12421
/u/	<u>u</u> va	6864 + 6710

5.1.3 Validation Data

Training disregarded vowels uttered in a real aloud context. This task, present for all speakers, was utilized for validation purposes. Thus, from the original dataset, the data retrieved for validation was as follows: the same four vowels utilized in training, described in table 5.6, but recorded in a read aloud context.

These frames were discarded in training due to lower number of frames per vowel, and previous phoneme residues being more likely to affect the current phoneme's frame. As such, they did not interfere in the training steps, but were still used to check the system capacity to generalize and obtain objective results.

5.1.4 Testing Data

Lastly, the utilized data for testing scenarios was as follows:

- The training dataset was utilized to analyze the non-linear transformations of whispered speech into normal speech.
- The validation dataset was utilized to obtain evaluation metrics and evaluate the system.
- Remaining oral vowels are utilized for zero shot learning.

5.2 Spectral Envelope Modelling

Spectral envelope models were derived from LPC coefficients, based on the Levinson-Durbin Recursion. These respective models were posteriorly utilized to derive cepstral features (MFCCs), where MFCCs act as input to the GAN-based system. To obtain the input spectral envelopes, short-time Fourier transform (STFT) was applied to the input signal in time-domain, dividing speech into frames - $x[n]$. In order to convert the signal to the frequency domain, the Odd-Discrete Fourier Transform (ODFT) [55] was applied to the temporal signal. Lastly, the power spectral density (PSD) of the signal was obtained and the Levinson-Durbin recursion was used to obtain the spectral envelope models - $c[k]$. In Fig. 5.2 the process to retrieve spectral envelope models is illustrated.

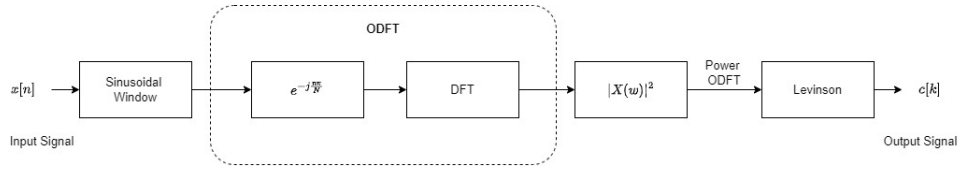


Figure 5.2: Block diagram of spectral envelope models.

In order to obtain the spectral envelopes, the STFT was applied to the input signal, windowing the temporal signal using a sinusoidal window w_n defined in Eq. 5.1, where frame length is specified by the variable $N = 1024$, with 50% overlap, leading to a window step of $N_{hop} = 512$.

$$w_n = \sin\left(\frac{\pi(n+0.5)}{N}\right), n = 0, 1, \dots, N-1 \quad (5.1)$$

The sinusoidal window is illustrated in Fig. 5.3.

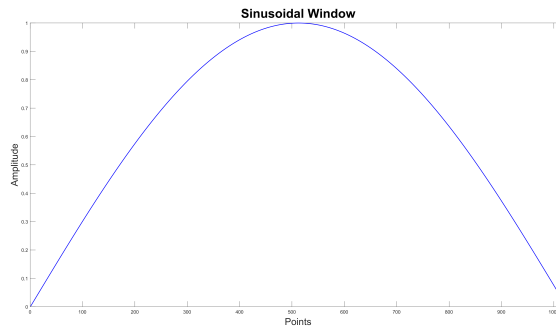


Figure 5.3: Frame Window

Given that the framed data was in the time-domain, ODFT was utilized to convert the signals to the frequency domain. As such, firstly, the framed data was multiplied by

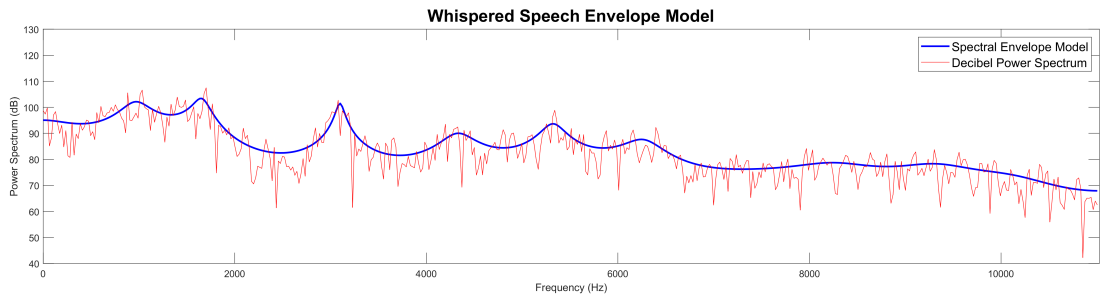
$$exp_n = e\left(-j\frac{(n\pi)}{N}\right) \quad (5.2)$$

where n corresponds to the data index in the respective frame of length N . Later, the Discrete Fourier Transform (DFT) was applied to the signal. The advantage of utilizing ODFT is that it induces a $\frac{\pi}{N}$ shift in the frequency domain, for each frequency sample, i.e. each frequency bin. This leads to a higher degree of precision characterizing the minimum and maximum frequency bins. Lastly, using this technique, a real signal is characterized in frequency domain by $\frac{N}{2}$ coefficients, instead of $\frac{N}{2} + 1$.

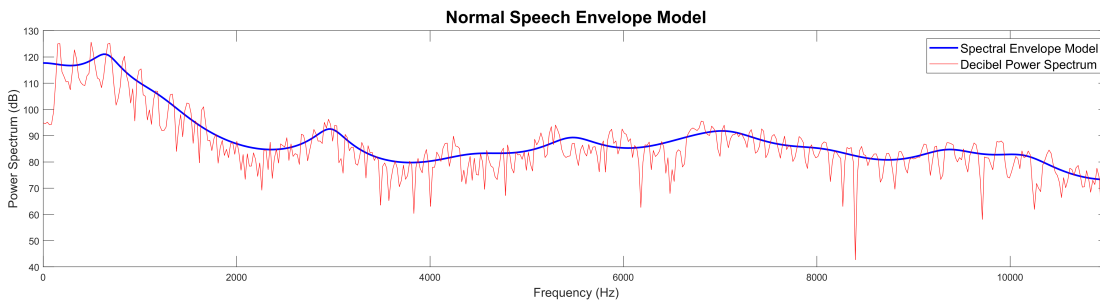
The PSD of the framed signal was retrieved from the obtained ODFT representation of the signal. As the input signal was a real signal, the PSD is characterized by $\frac{N}{2}$ coefficients (i.e. 512 coefficients). PSD is derived by

$$PSD = |X(w)|^2, 0 \leq w < \frac{N}{2} \quad (5.3)$$

From the obtained PSD , considering a linear frequency scale, the auto correlation coefficients were derived using the Wiener-Khintchine Theorem. Lastly, the LPC all-pole model coefficients were obtained, with the use of the Levinson-Durbin algorithm, taking into account a proposed LPC order of 22. Figure 5.4 displays the obtained LPC model, according to the PSD .



(a) Whispered speech spectral envelope model



(b) Normal speech spectral envelope model

Figure 5.4: Spectral envelope models for (a) whispered speech and (b) normal speech using Levinson-Durbin Recursion

5.3 Cepstral Features

In the proposed method, cepstral features were represented by Mel Frequency Cepstral Coefficients (MFCCs), and acted as the input features to the GAN-based models. Neural networks are composed of a finite number of variables - weights - that transform the input data - input features - into an output. Weights in the network layers are previously initialized. In training, the respective weights are updated according to the proposed loss function and gradient descent model. In the present dissertation, the proposed system is composed of input features that characterize the spectral envelope. As such, the dataset was firstly converted into features. This process is known as feature extraction.

Later, feature batching allowed the concatenation of data into multiple multidimensional arrays, with whispered MFCCs and normal MFCCs in each column, to represent the neural network input features. A fixed size is given to the multidimensional arrays, known as batch size. Each batch of data is fed to the neural network for training, and posterior testing scenarios. The GAN-based system is responsible to convert whispered speech cepstral features into normal speech cepstral features. Figure 5.5 summarizes the executed steps to convert the dataset information into features as inputs to the neural network.

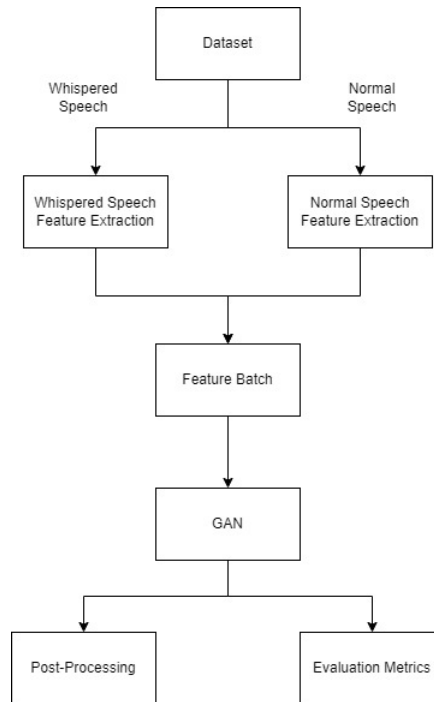


Figure 5.5: Block diagram of feature retrieval and methodology

5.3.1 Feature Extraction

Feature extraction is the conversion of individual frames of speech into information that represents the spectral envelope for the specified domains - whispered speech framed data and normal speech

framed data.

Cepstral features were derived from the spectral envelope models explained in the previous section. Considering that normal speech and whispered speech hold different formant shapes and energy distributions in the frequency domain, the process to retrieve cepstral features was different according to speech type, in the following manner:

- Whispered speech cepstral features are based in the whispered spectral envelope model.
- Normal speech cepstral features are based in the normal speech spectral envelope model and peak analysis derived from harmonic structures of voiced speech.

As spectral envelope models have already been explained, the conversion to MFCCs needs to be tackled. To convert spectral information into MFCCs, firstly, a filterbank based on a Mel-scale was necessary. Applying the filters to the power spectrum allowed the extraction of frequency bands and respective magnitudes. Lastly, Discrete Cosine Transform (DCT) of type II and liftering were utilized to convert to MFCCs. Figure 5.6 illustrates the process to retrieve the cepstral features.

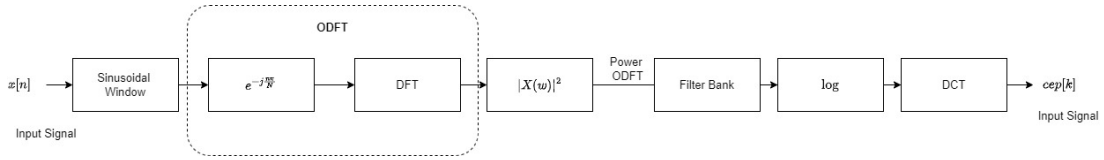


Figure 5.6: Block diagram of spectral envelope modelling

The Mel-Spectrum was computed by passing the spectral envelope models through a set of band-pass filters known as mel-filter bank. The mel scale was utilized using 22 frequency bins and a sub-mel scale utilizing 40 frequency bins. Posterior results highlight the differences in scaling. The Mel-scale aims to mimic the non-linear human ear perception of sound, by being more discriminative at lower frequencies and less discriminative at higher frequencies. The applied triangular filters are represented in figure 5.7.

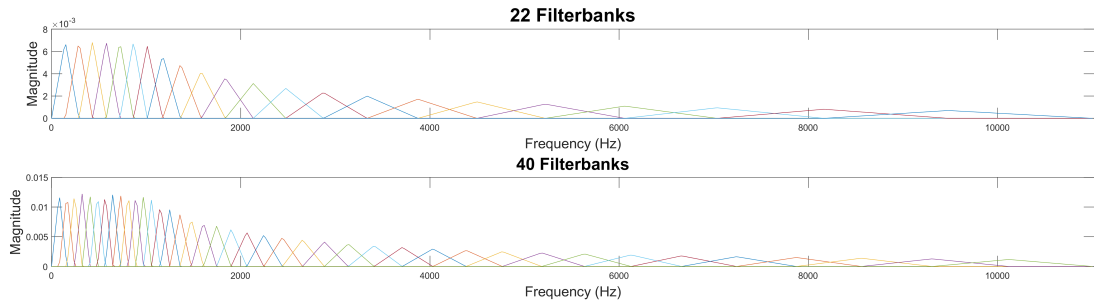


Figure 5.7: Cepstral Filterbanks: 22 versus 40 triangular filters

To define each filter, firstly, a central frequency, f_i , determines the highest magnitude of the triangular filter. A function $H_i(k)$ is utilized to describe the filters, considering that i represents the

i^{th} output band, and k represents the k^{th} energy spectrum bin. $H_i(k)$ may be defined as

$$H_i(k) = \begin{cases} 0 & \text{if } k \leq f_{i-1} \\ \frac{k-f_{i-1}}{f_i-f_{i-1}} & \text{if } f_{i-1} < k < f_i \\ 1 & \text{if } k = f_i \\ \frac{f_{i+1}-k}{f_{i+1}-f_i} & \text{if } f_i < k < f_{i+1} \\ 0 & \text{if } k \geq f_{i+1} \end{cases} \quad (5.4)$$

The result from applying the filter bank to the spectral envelope is that frequency information was separated into frequency bands, emulating the critical band in the human auditory system.

However, a problem arises considering the smoothness of the vocal tract. Energy levels in adjacent bands tend to be correlated. As such, DCT was applied to the logarithmic mel-spectrum. This resulted in the used representation for the cepstral domain. The use of DCT leads to significant information of the signal to be represented by the first MFCC coefficients.

After filtering, Discrete Cosine Transform (DCT) of type II was applied to the filtered signal, and this operation can be defined by

$$c_j = \sum_{i=1}^{N_{bands}} 10 \log_{10}(s(i)) \cos\left(\frac{j\pi}{N_{bands}}(i-0.5)\right) \quad (5.5)$$

where $s(i)$ represents the Mel-Spectrum, and i represents the i^{th} frequency band, in the total of N_{bands} .

Lastly, sinusoidal liftering was applied, according to a factor $D = -0.05$, where $MFCC_i$, represent the MFCCs scaled by $w_i = 1 + \frac{D}{2} \sin\left(\frac{\pi i}{D}\right)$, according to $\widehat{MFCC}_i = w_i MFCC_i$. This leads to the following number coefficients for cepstral information in the proposed methodology: (a) 22 MFCCs derived from 22 triangular filters, DCT and posterior subtle lifting, and (b) 40 MFCCs derived from 40 triangular filters (sub-mel scale), DCT and posterior subtle lifting.

5.3.2 Whispered Speech Spectral Envelope

The whispered spectral envelope derived from cepstral features is based in the spectral envelope models, displayed as an example in Fig. 5.8.

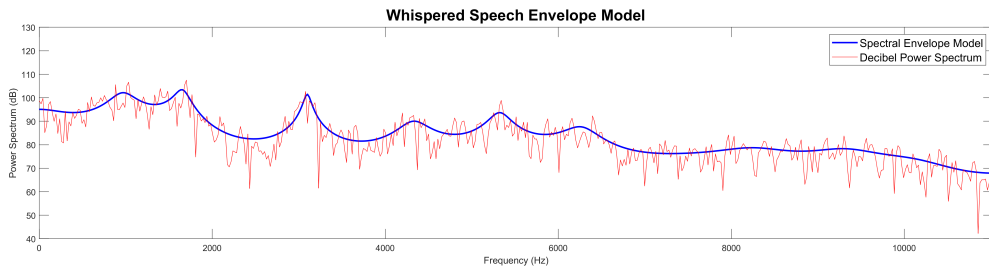


Figure 5.8: Whispered speech spectral envelope model

The retrieved MFCCs are derived from the spectral envelope model. Thus, the number of coefficients should be sufficient to map the spectral envelope with high quality. As Fig. 5.9 explicits, 22 coefficients in the cepstral domain did not retain information in mid-high frequencies. The second formant was almost nullified, leading to poor results. However, 40 coefficients showed that mid-high frequency information was still retained, and second and third formant were visible.

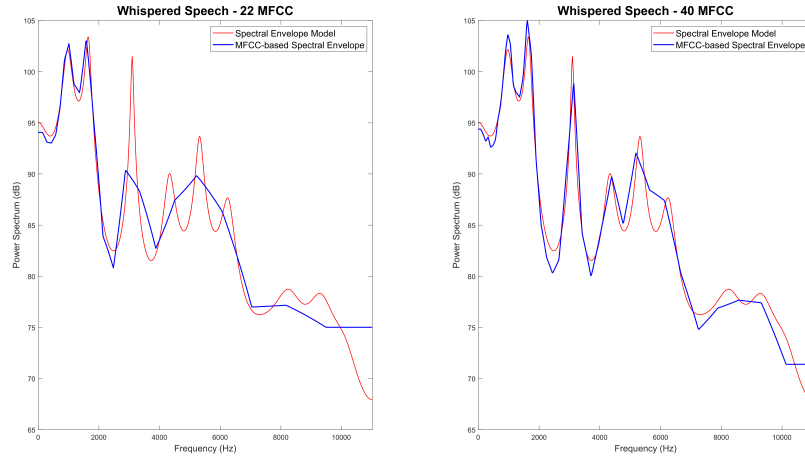


Figure 5.9: Spectral envelope from MFCCs: 22 versus 40 Coefficients

5.3.3 Normal Speech Spectral Envelope

For normal speech, the process to obtain the spectral envelope model explained in Section 5.2 was used to obtain the power spectral density, PSD . The main difference considered, in regard to whispered speech, was that, considering normal speech, the fundamental frequency and respective harmonic components of the signal were taken into consideration in order to retrieve the spectral envelope model. Fig. 5.10 illustrates the obtained spectral envelope model, utilizing 22 LPC coefficients, and no harmonic structure taken into consideration.

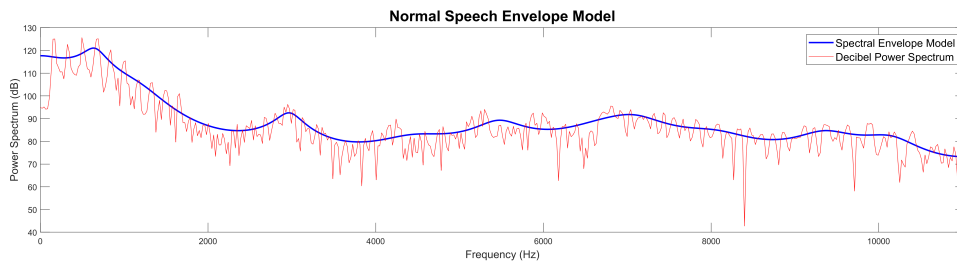


Figure 5.10: Normal speech spectral envelope model: 22 LPC coefficients

Utilizing the same methodology as in whispered speech, using 22 LPC coefficients, the results are highlighted in Fig. 5.11.

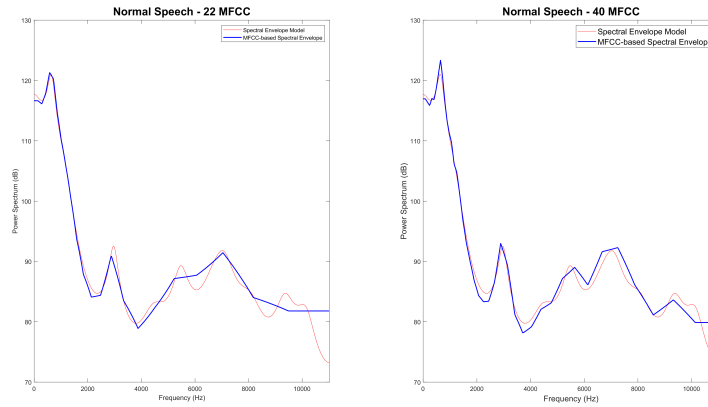


Figure 5.11: MFCC-derived spectral envelope: 22 versus 40 MFCCs

The main issue with applying the same methodology from whispered speech to voiced speech is that abrupt variations derived from the periodic component in the PSD of the voiced signal are not taken into account. As abrupt variations occur, the tendency for the LPC model is to introduce smoothness in the spectral envelope.

For a higher quality WHSP2SPCH conversion, an algorithm was utilized to predict fundamental frequency and the periodic component of the signal. This algorithm is SearchTonal[56], created by the current dissertation advisor. Utilizing the proposed method, the relevant peaks in spectral magnitude were obtained for voiced speech. The new spectral envelope was derived using linear interpolation based in the predicted spectral peaks of the harmonic structure (in dB), using the same tools as previously (Wiener-Khintchine theorem and Levinson-Durbin algorithm) to obtain the LPC model, as illustrated in Fig. 5.12. The Levinson-Durbin algorithm used 128 coefficients instead of 22 coefficients.

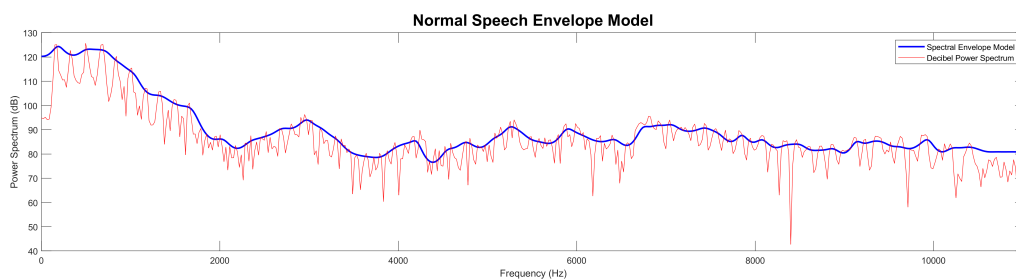


Figure 5.12: Normal speech spectral envelope model: 128 LPC coefficients and peak analysis

The difference between the spectral envelope based in 128 coefficients and 22 coefficients is relevant for the capacity of MFCCs to retain spectral information. As illustrated in Fig. 5.13, considering a higher order LPC, the second and third formant shapes were preserved and less smoothness occurred.

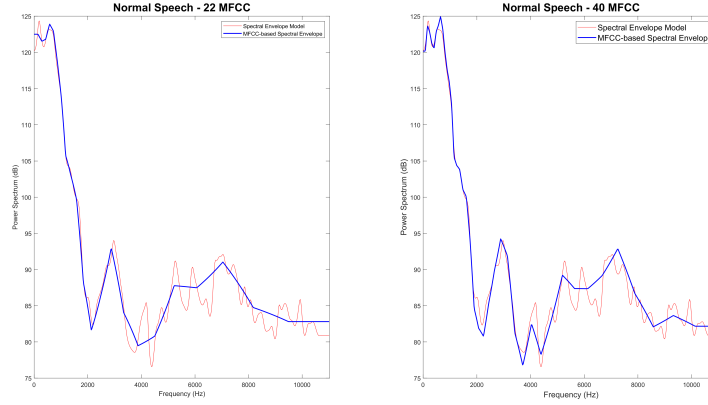


Figure 5.13: Higher quality MFCC-derived spectral envelope: 22 versus 40 MFCCs

5.3.4 Feature Batching

The discussed extraction of MFCC features from whispered speech and normal speech leads to the batching of respective features. Feature batching is the process to concatenate and organize features in multidimensional arrays that hold both domain features, i.e. whispered speech domain features and normal speech domain features, where, in the proposed methodology, MFCCs represent the domain features. Each array has a fixed size, called batch size. Batch size is a term used in machine learning that refers to the number of training examples utilized in each iteration.

Increasing the performance of a Generative Adversarial Network (GAN) requires experimentation in choosing the batch size as a hyperparameter. There is no consensus on optimal batch sizes in GANs, which makes it a “trial-and-error” process [57].

5.4 GAN Topologies

Generative Adversarial Networks are introduced in spectral envelope mapping. Considering whispered speech MFCCs and normal speech MFCCs have been extracted and batched, the GAN-based network had two main tasks - training and testing. The training and testing processes are detailed in figure 5.14. The training phase of the GAN-based network is the adjustment of weights that in the testing phase will be used to generate normal speech spectral envelopes.

Considering that different GAN models may be implemented, this project focused on whisper to normal speech conversion while using the following architectures:

- Vanilla GAN.
- CycleGAN [58].
- DiscoGAN [59].

The GAN-based systems - CycleGAN, DiscoGAN and vanilla GAN - were implemented utilizing pytorch [60] and respective documentation of the library is explicitly referenced for clarity.

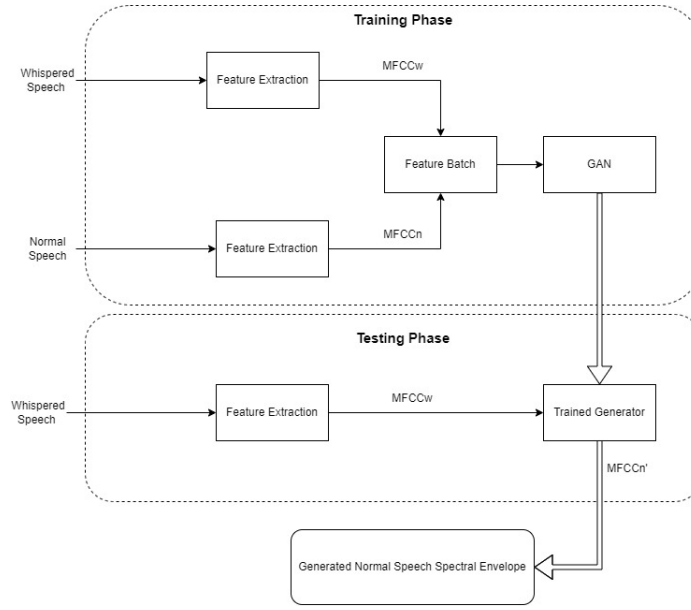


Figure 5.14: Block diagram of training and testing processes

CycleGAN and DiscoGAN are GAN-based topologies that attempt to discover cross-domain relationships. Their training relies on two generators, and two discriminators. Additionally, convergence relies on specific loss functions: cycle-consistency loss for cycleGAN and reconstruction loss for discoGAN. Vanilla GAN acts as a baseline to measure improvements in respect to the objective criterion. Table 5.7 presents the topology for the Vanilla GAN, Table 5.8 presents the topology for DiscoGAN, and Table 5.9 presents the topology for CycleGAN. The introduced variable X denotes the number of input/output coefficients for the generator and input for the discriminator, representing the number of MFCCs fed to the system. In the proposed method, 22 MFCCs or 40 MFCCs act as inputs.

Vanilla GAN	Topology
Generator G_{WS}	$X/512/512/512/X$
Discriminator D_S	$X/512/512/512/1$

Table 5.7: Vanilla GAN Topology.

DiscoGAN	Topology
Generator G_{WS}	$X/512/512/512/X$
Generator G_{SW}	$X/512/512/512/X$
Discriminator D_S	$X/512/512/512/1$
Discriminator D_W	$X/512/512/512/1$

Table 5.8: DiscoGAN Topology.

Cycle-GAN	Topology
Generator G_{WS}	$X/512/512/512/X$
Generator G_{SW}	$X/512/512/512/X$
Discriminator D_S	$X/512/512/512/1$
Discriminator D_W	$X/512/512/512/1$

Table 5.9: Cycle-GAN Topology.

5.4.1 Nomenclatures

Nomenclatures are used as specified, in order to describe the system and analyze results. This aids in simplifying explanations and notations. The nomenclature regarding corresponding domains are as follows:

- S - normal speech MFCCs.
- W - whispered speech MFCCs.
- S' - normal speech MFCCs generated by the network.
- W' - wshipered speech MFCCs generated by the network.

The nomenclature utilized for the following sections and chapters to describe the generators and discriminators are as follows:

- G_{WS} - generator that attempts to convert input, whispered speech MFCCs, into normal speech MFCCs.
- G_{WS} - generator that attempts to convert input, normal speech MFCCs, into whispered speech MFCCs.
- D_S - discriminator that attempts to distinguish if the input is from the S domain (real) or not (fake). In other words, if the input is real normal speech or not.
- D_W - discriminator that attempts to distinguish if the input is from the W domain (real) or not (fake). In other words, if the input is real whispered speech or not.

5.4.2 Neural Network Layers

There are various types of layers in state-of-the-art deep learning models. It can be considered as the architecture of the model. Different layers have their importance based on respective features. In the present dissertation, the input layer, the first layer of the network, has a size of 40 or 22 neurons as input, where the input represents 40 MFCCs or 22 MFCCs. The input linear layer outputs 512 coefficients. Linear layers use matrix multiplication to transform their input features into output features using a weight matrix and a bias. As such, linear layers apply a linear transformation to the incoming data,

$$y = xA^T + b \quad (5.6)$$

The parameter initialization for A and b is given by [61], following a uniform distribution.

The activation functions utilized for the generators were the rectified linear activation function (ReLU), a piecewise linear function that outputs the input directly if it is positive, otherwise, it will output zero. The discriminator had ReLU activation function in each layer, except the output. The output was activated utilizing the sigmoid function. For clarity, the ReLU function is described as

$$ReLU(x) = (x)^+ = \max(0, x) \quad (5.7)$$

and the sigmoid function is described as

$$\text{Sigmoid}(x) = \sigma(x) = \frac{1}{1 + \exp(-x)} \quad (5.8)$$

For illustrative purposes, figure 5.15 highlights both activation functions and their differences.

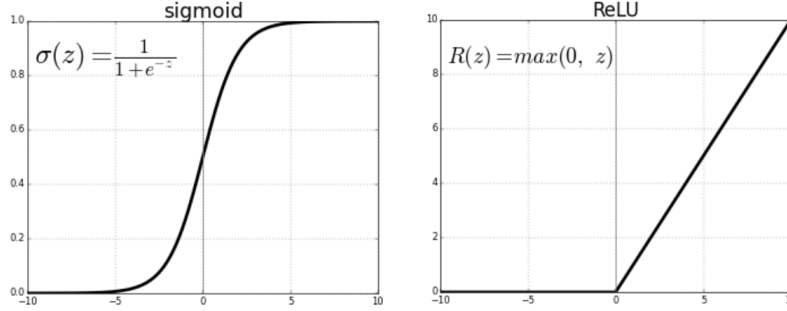


Figure 5.15: Network Activation Functions: Sigmoid and ReLU

To the extent of the author’s knowledge, no studies have been made utilizing cycleGAN and discoGAN in respect to qualitative results for different activation functions for spectral envelope mapping. Studies have been made regarding speech enhancement [62], but not in voice conversion. Different architectures use different activation functions and, as such, the activation functions utilized are based in the literature [49, 10, 9]. For the proposed DiscoGAN, the literature indicates that best results are produced introducing ReLU activation functions instead of sigmoid or tanh functions [11] for fundamental frequency prediction. However, no study is known for spectral envelope mapping.

5.4.3 Graphical Overview

For simplicity and clarity, the neural networks used may be over-viewed utilizing the aid of a graphical representation. As the generators G_{WS} and G_{SW} shared the same layers, activation functions and output shape, Fig. 5.16 illustrates the generators in a graphical context. The same logic applies to the discriminators D_S and D_W as the same layers existed in both discriminators. Figure 5.17 illustrates the discriminators in a graphical context.

As already pointed out, the main difference in the discriminator and generator is that the discriminator utilized the sigmoid to predict if the input is real or fake. Instead, the generator outputs the cepstral coefficients for the target domain.

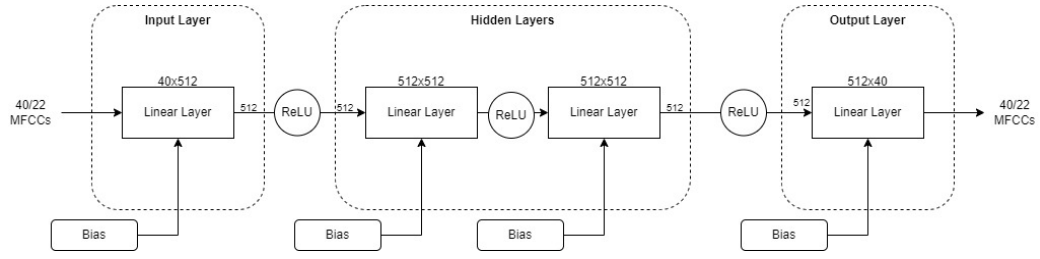


Figure 5.16: Block diagram of the Generator

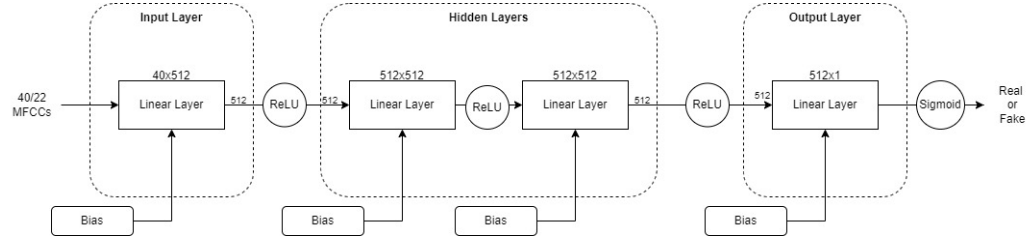


Figure 5.17: Block diagram of the Discriminator

5.4.4 Optimizer

Deep learning neural networks are trained using stochastic gradient descent optimization algorithm. Utilizing pytorch [60], two optimizers are commonly utilized: Adaptive Moment Estimation (Adam optimizer) and Stochastic Gradient Descent (SGD optimizer), as the documentation reveals [63].

Adaptive optimization methods perform local optimization with a metric constructed from the history of iterations [64]. The main advantage is rapid training time. The optimizer utilized in the project's system is Adam - Adaptive Moment Estimation. As for disadvantages, the generalization of systems using adaptive gradient methods remains poorly understood [64].

Studies have shown that the standard gradient methods may not be able to find complex solutions for a given amount of epochs [64], compared to Adam methods. However, SGD tends to be more stable. Thus, solutions with simpler dimensionality will be found, and are prone to generalize well. Generalization is neural network's ability to handle unseen data. The generalization capability of a given network is mostly determined by system complexity and training of the network [65]. As such, different hypothesis are present in the choice of an optimizer for a neural network model.

The main purpose of using Adam optimizer in the project was due to rapid training time, as the computational power for the project may have not been enough to complete all studies intended, if SGD was utilized. For clarification, the GPU utilized for training and testing was the Nvidia RTX 3060Ti, with 4864 cuda processors. The CPU was an i7 10700K @ 4GHz base-clock, with 8 cores and 16 threads. As for the operating system, Windows 10 was used, and the pytorch version was Pytorch 3.7 (via Anaconda).

5.4.5 Loss Functions

The simple adversarial loss in the vanilla GAN is explained according to the dynamic between discriminator D and generator G , and their dependencies in the loss function, given by a binary cross entropy loss, as

$$\mathcal{L}_{GAN}(D, G) = E_{W \sim p_W(w)}[\log(D(w))] + E_{p_z}[\log(1 - D(G(w)))] \quad (5.9)$$

CycleGAN was introduced for unpaired image-to-image translation [58]. Three loss functions are considered: the standard adversarial loss, a cycle-consistent loss and an identity loss. The adversarial loss is applied to both G_{WS} with discriminator D_S , and G_{SW} with discriminator D_W . The objective function is declared as

$$\mathcal{L}_{GAN}(G_{WS}, D_S) = E_{S \sim p_S(s)} [(D_S(s) - 1)^2] + E_{W \sim p_W(w)} [(D_S(G_{WS}(w)))^2] \quad (5.10)$$

where G_{WS} tries to generate MFCCs $G_{WS}(W)$ similar to MFCCs from domain S , and D_S attempts to distinguish generated normal speech $G_{WS}(W)$ and the real reference sample s . In the sense of the mix-max game, G_{WS} aims to minimize the objective function, and D_S tries to maximize it as

$$\min_{G_{WS}} \max_{D_S} \mathcal{L}_{GAN}(G_{WS}, D_S, W, S) \quad (5.11)$$

A parallel adversarial loss function is utilized for the opposite mapping function used by G_{SW} and the discriminator D_W .

Cycle-Consistency Loss is implemented to map the distribution between original and reconstructed data. In addition, this loss tries to preserve contextual information across different speech [9]. It is declared as

$$\mathcal{L}_{cc}(G_{WS}, G_{SW}) = E_{W \sim p_W(w)} [\|G_{SW}(G_{WS}(w)) - w\|_1] + E_{S \sim p_S(s)} [\|G_{WS}(G_{SW}(s)) - s\|_1] \quad (5.12)$$

Lastly, identity loss is utilized to encourage preservation of input linguistic content [9]. This loss is not present in the original paper for cycleGAN [58]. It can be written as

$$\mathcal{L}_i(G_{WS}, G_{SW}) = E_{W \sim p_W(w)} [\|G_{SW}(w) - w\|_1] + E_{S \sim p_S(s)} [\|G_{WS}(s) - s\|_1] \quad (5.13)$$

The total loss function is defined as:

$$\mathcal{L}_{total} = \mathcal{L}_{GAN}(G_{WS}, D_S) + \mathcal{L}_{GAN}(G_{SW}, D_W) + \lambda_{cc} \mathcal{L}_{cc}(G_{WS}, G_{SW}) + \lambda_i \mathcal{L}_i(G_{WS}, G_{SW}) \quad (5.14)$$

where λ_i and λ_{cc} are 5 and 10, respectively, and represent the weight given to identity loss function and cycle-consistency loss function. λ_i and λ_{cc} were derived from referenced work, where λ_{cc} is proposed as the original cycleGAN paper [58] and λ_i is derived from experimental work, based in referenced work [9][11]. From the weights given to the losses, $\mathcal{L}_{cc}(G_{WS}, G_{SW})$ is the most prominent.

Figure 5.18 illustrates the training process and identified losses in a graphical manner.

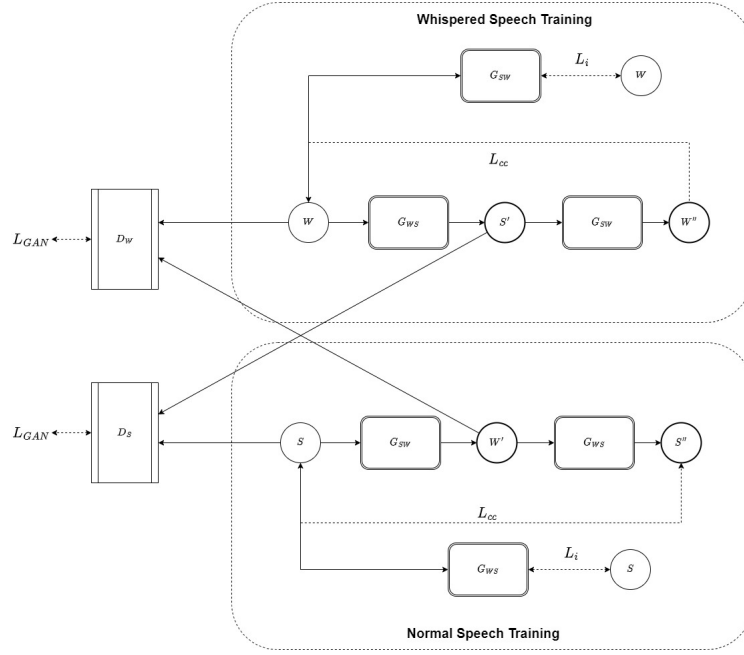


Figure 5.18: Block diagram of the training process and respective losses for CycleGAN

Loss functions for DiscoGAN are different from the proposed referenced literature [59] and similar to reference [11]. The utilized losses are two adversarial loss $\mathcal{L}_G$ (generator loss) and $\mathcal{L}_D$ (discriminator loss) and optimization is objectively defined by minimizing the two regularized adversarial losses [11]. The reconstruction loss is introduced, similar to cycleGAN, to emphasize the need to understand both domains by utilizing $G_{SW}(G_{WS}(W))$ and $G_{WS}(G_{SW}(S))$ to validate the generator's behavior. By minimizing the reconstruction loss, with the WHSP2SPCH conversion for the MFCCs, enforces the generated whispered speech parameters, W'' to be close to the original whispered speech parameters, W and the converted normal speech parameters, S'' to be close to the normal speech parameters, S [11]. The adversarial losses are defined as $\mathcal{L}_G(G_{SW}, D_W)$ and $\mathcal{L}_D(G_{SW}, D_W)$:

$$\begin{aligned}\mathcal{L}_G(G_{SW}, D_W) &= -\mathbb{E}_{s \sim p_S(s)} [\log(D_W(G_{SW}(s)))] + \frac{1}{2} \mathbb{E}_{w \sim p_W(w), s \sim p_S(s)} [\log(w) - \log(\log(G_{SW}(s)))]^2 \\ \mathcal{L}_D(G_{SW}, D_W) &= -\mathbb{E}_{w \sim p_W(w)} [\log(D_W(w))] - \mathbb{E}_{s \sim p_S(s)} [\log(1 - D_W(G_{SW}(s)))]^2\end{aligned}\quad (5.15)$$

Here, G_{WS} , G_{SW} , D_W and D_S are jointly trained. Hence, the generator loss for G_{WS} can be defined as:

$$\begin{aligned}\mathcal{L}_{G_{WS}} &= \mathcal{L}_W + \mathcal{L}_{G_S} \\ \mathcal{L}_{G_{SW}} &= \mathcal{L}_S + \mathcal{L}_{G_W}\end{aligned}\quad (5.16)$$

where $G = G_S$, representing whispered speech generator, replacing $\mathcal{L}_G(G_{SW}, D_W)$ by

$\mathcal{L}_G(G_{WS}, D_S)$ and $\mathcal{L}_W$ represents the reconstruction loss for whispered speech, where w should be as similar as possible to $G_{SW}(G_{WS}(w))$. The same logic applies for normal speech, as $G = G_W$, we utilize $\mathcal{L}_G(G_{SW}, D_W)$ and $\mathcal{L}_S$ represents the reconstruction loss for normal speech. The reconstruction losses can be modelled by:

$$\mathcal{L}_W(G_{SW}, G_{WS}) = d((G_{SW}(G_{WS}(w))), w) \quad (5.17)$$

and

$$\mathcal{L}_S(G_{SW}, G_{WS}) = d((G_{WS}(G_{SW}(s))), s) \quad (5.18)$$

Thus,

$$\mathcal{L}_{Rec} = \mathcal{L}_S(G_{SW}, G_{WS}) + \mathcal{L}_W(G_{SW}, G_{WS}) \quad (5.19)$$

and

$$\mathcal{L}_{GAN} = \mathcal{L}_{D_W} + \mathcal{L}_{D_S} \quad (5.20)$$

where $\mathcal{L}_{D_W}$ and $\mathcal{L}_{D_S}$ are represented according to Eq. 5.21, where $\mathcal{L}_D$ is specified.

The total loss, $\mathcal{L}_{total}$ is defined as

$$\begin{aligned} \mathcal{L}_{G_{total}} &= \mathcal{L}_{G_{WS}} + \mathcal{L}_{G_{WS}} \\ \mathcal{L}_{D_{total}} &= \mathcal{L}_{D_S} + \mathcal{L}_{D_W} \\ \mathcal{L}_{total} &= \mathcal{L}_{G_{total}} + \mathcal{L}_{D_{total}} \end{aligned} \quad (5.21)$$

Figure 5.19 displays the training process and identified losses - $\mathcal{L}_{GAN}$ and $\mathcal{L}_{Rec}$ for simplicity - in a graphical manner.

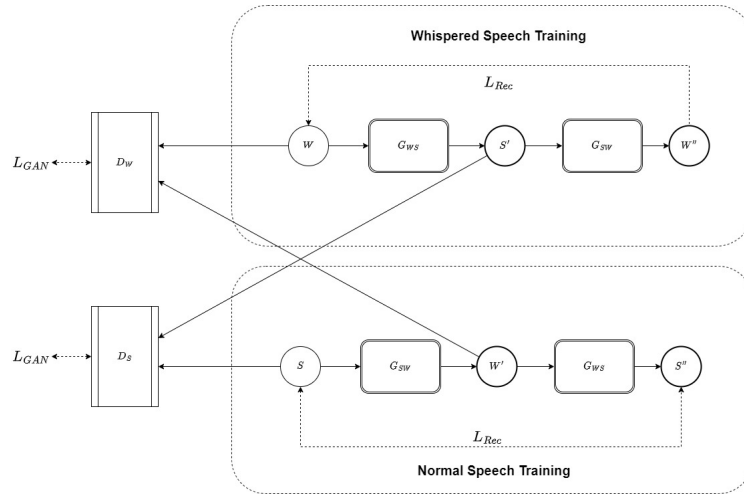


Figure 5.19: Block diagram of the training process and respective losses for DiscoGAN

The proposed architectures - DiscoGAN and CycleGAN - were chosen due to their similarities. Their approach is similar in measuring loss functions, as reconstruction loss shares the same purpose as cycle-consistency loss.

Posterior to the losses that represent both discoGAN and cycleGAN, a supervised loss is added to the system. This supervised loss is declared as a Mean Squared Error Loss Function. To put simply, this supervised loss determined by Eq. 5.22, where $\mathcal{L}_{supervised}$ represents the total supervised loss introduced in the system, and respective $L1$ and $L2$ components, representing the loss from generated output in comparison to the parallel-corpora reference.

$$\begin{aligned}\mathcal{L}_{1_{supervised}} &= \frac{1}{2} \mathbb{E}_{w \sim p_W(w)} [(G_{WS}(w) - s)]^2 \\ \mathcal{L}_{2_{supervised}} &= \frac{1}{2} \mathbb{E}_{s \sim p_S(s)} [(G_{SW}(s) - w)]^2 \\ \mathcal{L}_{supervised} &= \lambda_{sup} \mathcal{L}_{1_{supervised}} + \mathcal{L}_{2_{supervised}}\end{aligned}\tag{5.22}$$

Introducing this equation with $\lambda_{sup} = 1$ does not affect the system greatly, as the reconstruction and cycle-consistency still out scale this loss. Despite this, improvements in results are still expected. Consideration of variations in λ_{sup} where not present in results, to analyze the impact the loss makes in the system. Future work is needed to understand supervision in cross domain GAN architectures and their impact in performance.

5.4.6 Hyperparameters

Hyperparameters were changed from the original literature of proposed methods [9][11]. As training data holds a significantly lower number of frames, batch size and learning rate were updated accordingly, based on empirical observation. Table 5.10 highlights the hyperparameters used. Unlike common practice in WHSP2SPCH conversion, batch size used is represented 100 frames. Initial learning rate was lowered to 0.0001, compared to proposed architectures utilizing 0.0002 [58].

Table 5.10: Hyperparameters obtained through experimentation for considered architectures

Hyperparameters	Values
Initial Learning Rate	0.0001
Training Epochs	500
Batch Size	100

Learning rate is the hyperparameter that regulates the change that can occur to the model, in response to the estimated error, each time the model weights are updated. Deriving the optimal learning rate is challenging due to the scale utilized, as little variations may greatly affect the system.

For the proposed system, the learning rate is adaptive after training epochs arrive to the middle point of 250 epochs, until the last epoch is reached, i.e. epoch 500, as expressed in Fig. 5.20.

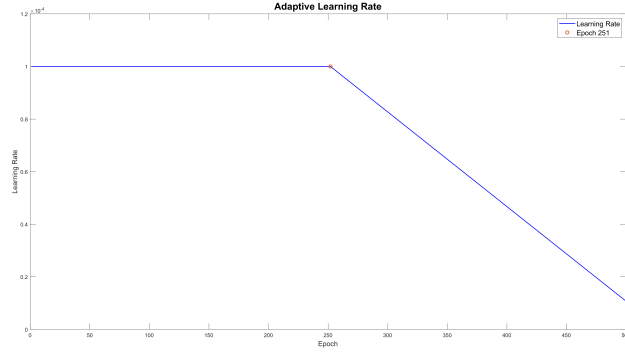


Figure 5.20: Adaptive learning rate overview given 500 epochs, introducing a linear decrease after epoch 250

5.5 Evaluation Metrics

Evaluation metrics were implemented for validation and evaluation of the GAN-based system. One objective criterion was utilized, allowing comparison of obtained results in different experiments. Lastly, one subjective criterion was used to measure perceived quality of the given system.

5.5.1 Objective Criterion

Mel cepstral distortion (MCD) is the measure of difference between two sequences of cepstral features. In the context of this dissertation, the sequences are represented by MFCCs. This criterion is used to assess the quality of parametric speech synthesis systems, including statistical parametric speech synthesis systems. The following equation expresses the traditional MCD measure [66].

$$MCD[dB] = \frac{10}{\ln(10)} \sqrt{2 \sum_{i=1}^N (m_i^t - m_i^c)^2} \quad (5.23)$$

The variable m_i^t and m_i^c represent the i_{th} MFCC of t and c . t represent the generated speech $G_{WS}(w)$, while c represents the reference (ground truth) normal speech, s . Considering that the smaller the distance obtained for an MCD result means higher proximity, objectively a better solution for the system implies a lower MCD result.

5.5.2 Subjective Criterion

To evaluate perceived voice quality, a Mean Opinion Score (MOS) was used. Subjective tests were prepared that included three utterances by a female speaker, both in normal and whispered speech, for each of the following Portuguese words:

Table 5.11: MOS tasks: specific words uttered in each task

Value	Explanation
1	face
2	pica
3	velho

The subjects were then asked to evaluate the generated samples with the spectral envelope manipulated in each of the following modes:

- Spectral envelope based on the original whispered speech current frame, with gain and slope added as to better replicate the typical normal speech spectral envelope.
- Spectral envelope consisting in templates previously obtained from normal speech samples of the same vowels, from the same speaker.
- Spectral envelope obtained from the MFCCs inferred from the original whispered speech current frame by the pre-trained CycleGAN (NP).
- Spectral envelope obtained from the MFCCs inferred from the original whispered speech current frame by the pre-trained DiscoGAN (PSS).

The evaluation was obtained considering two questions. First, the subjects were asked to evaluate perceived quality, considering undesired artifacts. The scale considered was the standard 1-5, where each evaluated value is explicit according to Table 5.12.

Table 5.12: MOS Tasks: scale and indications

Value	Explanation
1	inaudible
2	Abundant identifiable artifacts
3	Some identifiable artifacts
4	Few identifiable artifacts
5	No artifacts

Regarding the second question, the subjects were asked to evaluate proximity to the reference signals. The scale considered was the standard 1-5, where each evaluated value is explicit according to Table 5.13.

Table 5.13: MOS Tasks: scale and indications

Value	Explanation
1	Identical to the whispered speech audio reference
2	Similar to the whispered speech audio reference
3	Intermediate
4	Similar to the normal speech audio reference
5	Identical to the normal speech audio reference

In short, the present subjective test attempted to evaluate perceived proximity and quality between two parametric systems and two statistical-based systems. The proposed approach allowed the comparison between parametric and statistical-based systems, and revealed differences between proposed GAN-based architectures. Moreover, parallel-corpora was used for discoGAN and non-parallel corpora for cycleGAN. This enabled results to verify the relative importance of parallel corpora in WHSP2SPCH conversion.

Speech synthesis is achieved using frame-by-frame spectral envelope remapping. For an individual whispered frame, features are extracted - spectral envelope model and MFCCs - as previously explained. To decide when to implant synthetic voice, a pre-trained deep neural network is used. If the whispered frame is to be converted to an unvoiced normal speech frame, no posterior transformation is applied to the frame. If the whispered frame is to be converted to a voiced normal speech frame, the extracted spectral envelope models are utilized in the rule-based approaches and MFCCs are utilized in the GAN-based systems - cycleGAN and discoGAN. Through the proposed mechanisms, the input whispered spectral envelope is mapped to a normal speech spectral envelope. Fundamental frequency prediction and NRD information are template-based. Considering this approach, there is one template for each individual parameter: fundamental frequency contour, average fundamental frequency, and NRD information. Thus, the differences between the samples are the result of the different spectral envelopes applied by each of the above-described methods.

Chapter 6

Results and Discussion

The present chapter discusses the obtained results of proposed architectures, cycleGAN and discoGAN, for spectral envelope remapping in the context of WHSP2SPCH conversion. For clarification, considered architectures use MFCCs as input features, obtained using the methods presented in Chapter 5.

The following experiments are explained in the subsequent sections of the chapter:

- objective MCD analysis for cycleGAN and discoGAN, considering vanilla GAN as baseline.
- objective MCD analysis for cycleGAN and discoGAN, for zero-shot learning vowels.
- Subjective Mean Opinion Score (MOS) evaluation and discussion of obtained results.

As previously explained, four vowels were considered in the training dataset. The remaining five vowels from the dataset were used to obtain MCD results for zero-shot learning, as explained in Chapter 5. Table 6.1 indicates the datasets and vowels present in each.

Table 6.1: Vowels and respective purposes: training dataset and zero-shot learning dataset, with respective examples.

Training Dataset	Example	Zero-Shot Dataset	Example
/i/	<u>i</u> lha	/ê/	p <u>e</u> so
/á/	<u>a</u> gua	/é/	<u>e</u> la
/â/	<u>a</u> marelo	/ó/	<u>o</u> culos
/u/	<u>u</u> va	/ô/	<u>o</u> vo
		/e/	sed <u>e</u>

Lastly, the datasets were further separated into gender categories - female, male, and all speakers - retrieving gender-specific information related to spectral envelope remapping.

6.1 MCD Analysis

Mel Cepstral Distortion analysis is the first evaluation metric presented. This objective criterion was utilized to measure the difference between synthetic voiced speech cepstral features ($G_{WS}(w)$) and natural voiced speech cepstral features (s). As validation data has parallel corpora, s acted as reference (ground-truth). The proposed objective criterion was also utilized due to its relevance in current GAN-based systems[9, 11, 67, 10], for WHSP2SPCH conversion.

The MCD distance is method-dependent as different cepstral derivations of MFCCs lead to different distance measurements. Therefore, the objective criterion was not compared with the literature, since the obtained MCD results were obtained differently. Given this non-comparability, we used vanilla GAN MCD results to assess relative improvements in the system [9, 11, 67, 10].

As a starting point, $MCD(w, s)$ was obtained by computing the differences between natural whispered speech cepstral features, w , and natural voiced speech cepstral features, s . Later, $MCD(G_{WS}(w), s)$ was obtained by computing the differences between synthetic voiced speech cepstral features ($G_{WS}(w)$) and natural voiced speech cepstral features (s). As such, the MCD results considering synthetic voiced speech, $MCD(G_{WS}(w), s)$, should show a decrease in the measured distance compared to $MCD(w, s)$.

The distance between cepstral features for synthetic voiced speech, $MCD(G_{WS}(w), s)$, should be equal or less than the distance between cepstral features for whispered speech, $MCD(w, s)$. (H1)

As explained in hypothesis (H1), MCD results for natural whispered speech cepstral features ($MCD(w, s)$) were obtained in order to understand posterior improvements in experimental results. As such, expected results for the synthetic voiced cepstral features, $MCD(G_{WS}(w), s)$ are expressed by

$$MCD(W, S) \geq MCD(G_{WS}(W), S) \quad (6.1)$$

The posterior section indicates the MCD results obtained for the different experiments. Finally, two proposed architectures are presented considering the results obtained. These architectures are also used in the subjective evaluation, using the Mean Opinion Score (MOS) as the quality criterion.

6.1.1 MCD for input speech

Considering the difference between the input cepstral domains, this measurement is suitable to present the differences between the zero-shot learning dataset $MCD_z(w, s)$ and the training dataset, $MCD_t(w, s)$, in the cepstral domain. Table 6.2 indicates the obtained results for the training dataset, considering the distance between natural whispered MFCCs, w , and natural voiced speech MFCCs, s . Lastly, Table 6.3 indicates the results for the zero-shot learning data, considering the distance between whispered vowels and normal speech vowels.

Table 6.2: MCD between input whispered speech and input normal speech for the training dataset consisting of four vowels.

Architecture	Dataset *	MCD
$MCD_t(w, s)$	Female	6.236
	Male	5.868
	All Speakers	6.061

* Dataset contains utilized training vowels

Table 6.3: MCD between input whispered speech and input normal speech for the zero-shot learning vowels.

Architecture	Dataset*	MCD
$MCD_z(w, s)$	Female	5.365
	Male	4.933
	All Speakers	5.153

* Dataset contains the zero-shot learning vowels

The observed results in Table 6.3 and Table 6.2 indicate that zero-shot learning MFCCs have lower MCD scores, as expressed by $MCD_z(w, s)$. Thus, zero-shot learning MFCCs show higher similarity in domains (whispered speech and normal speech domains), compared to the training dataset. Overview of the posterior experimental results should regard current MCD observations as relevant when comparing the obtained results between training and zero-shot vowels.

6.1.2 MCD analysis for generated speech

As discussed in Chapter 5, the baseline for measuring improvements in the MCD analysis was the vanilla GAN, considering two modes of training - parallel and non-parallel. To specify training hyperparameters, 100 frames were used for the batch size, the number of training epochs was defined as 500, and the initial learning rate was 0.0001, with linear adaptive decrease starting at epoch 250. Results are shown in Table 6.4.

The obtained results, illustrated in Table 6.4, allow us to understand that cycleGAN and discoGAN outperform vanilla GAN for both training modes - parallel and non-parallel - disregarding that cycleGAN is outperformed in parallel training, for the male dataset by -2.1%.

CycleGAN demonstrates satisfactory MCD results for non-parallel training, but MCD results in parallel training are not as satisfactory as expected. Despite this, the unsupervised non-parallel training is the most flexible mode of training presented. As such, obtained results emphasize the hypothesis that cycleGAN is a good candidate for non-parallel voice conversion.

DiscoGAN shows the best objective results, with improvements up to +38.5% with parallel training, compared to baseline. Despite the need for parallel corpora in the dataset, the objective

Table 6.4: Mel Cepstral Distortion (MCD) for the baseline model, Vanilla GAN, and proposed architectures. Training used 500 epochs. The results are separated into training type and gender-specific data. The percentage in brackets indicates the relative reduction in the MCD w.r.t the baseline model.

Architecture	Training	Dataset	Epochs	MCD
Baseline	Non-Parallel	Female	500	4.845*
		Male	500	4.769*
	Parallel	Female	500	5.488*
		Male	500	4.121*
cycleGAN	Non-Parallel	Female	500	4.002 (+17.40%)
		Male	500	4.075 (+14.55%)
	Parallel	Female	500	3.969 (+27.68%)
		Male	500	4.206 (-2.06%)
discoGAN	Non-Parallel	Female	500	4.530 (+6.50%)
		Male	500	4.292 (+10.00%)
	Parallel	Female	500	3.375 (+38.50%)
		Male	500	3.368 (+18.27%)

* Constant Learning Rate

* $\text{lr} = 0.0001$

MCD results are the most successful up to this point. However, in non-parallel training, discoGAN is outperformed by cycleGAN.

The results presented used a linear descent learning rate, considering 500 epochs for training, where the number of epochs was established through experimentation. Despite this, the first experiments in the current dissertation were introduced using a constant learning rate and a variable size of training epochs. These experiments led to the results obtained indicated in Appendix C. Considering the constant learning rate, the generator performance first improved with a general linear sense, later decreased, and lastly, a new increase in performance was obtained, leading to a well-trained model. This concept of performance variability is similar to the concept of deep double descent [68]. Although this behavior appears to be fairly universal to deep learning, no consensus is given on why it happens [68]. This behavior leads to fluctuating results based on the total number of epochs. As indicated in Table C.3, the use of a linear descent approach to the learning rate reduced variability and this led to more independent results based on the number of epochs. For more detail regarding results obtained given a constant learning rate, and a simple comparison to the used linear descent learning rate, appendix C displays all necessary information, to understand the improvements in the latter approach.

Variations of batch size were also attempted, as different batch sizes have been shown to produce different results. Batch size controls the accuracy of the estimate of the error gradient when training neural networks. A higher number of frames during initial training leads the discriminator to retrieve more information per epoch, allowing the possibility of overpowering the

generator, thus leading to worse results in comparison. A smaller batch size introduces variability, offering a regularizing effect and lower generalization error. The results obtained considering increasing batch size are also explicitly present in the Appendix D. Increasing the batch size led to detrimental results. Due to time constraints, no further experiments were performed on variations in batch size.

The total training epochs depend on the size of the dataset. Through empirical experiments, 500 epochs were chosen for the dataset size, as this increased the stability of the results. Through experiments, the optimal learning rate was the linear descent learning rate, and the batch size utilized was 100 frames. Taking into account these hyperparameters, cycleGAN and discoGAN presented incremental results when compared to the baseline and improved results compared to other hyperparameters (learning rate and batch size). On the one hand, the obtained results indicate that cycleGAN achieves rather similar results with parallel training when compared to non-parallel. On the other hand, discoGAN improves MCD results considerably for the parallel mode. The difference in the obtained results highlight a new hypothesis:

discoGAN is suitable for parallel training, unlike cycleGAN, that presents higher results with non-parallel training. Thus, adding a supervised loss to (H2) the architectures may considerably affect discoGAN, unlike cycleGAN.

As explained in the hypothesis (H2), an experiment was made, adding a self-supervised loss, $\mathcal{L}_{sup}$, to the parallel training. System dynamics rely on the proposed loss functions - cycle-consistency loss for cycleGAN and reconstruction loss for discoGAN. As such, proposed self-supervised loss attempts to minimally disrupt the convergence in unsupervised loss functions as they impose stability on the architectures. Future studies are needed to clarify the effects of self-supervision, given a scalar value starting at $\lambda_{sup} = 1$, and the respective increase to introduce scalability, as in $\lambda_{sup}\mathcal{L}_{sup}$, as better results may be obtained.

The results for the self-supervised parallel mode, considering cycleGAN and DiscoGAN, are presented in table 6.5, and in parentheses the percentage of variation obtained in the self-supervised parallel mode, when compared to the unsupervised parallel mode, is presented.

Table 6.5: Mel Cepstral Distortion (MCD) for the proposed architectures considering self-supervised parallel training. Training utilized 500 epochs. The results are separated by training type and gender-specific data. The percentage in brackets indicates the relative reduction in the MCD w.r.t the baseline, which is the unsupervised parallel mode of training.

Architecture	Training	Dataset	Epochs	MCD
cycleGAN	Parallel Supervised	Female	500	3.965 (+0.10%)
		Male	500	4.161 (-1.08%)
discoGAN	Parallel Supervised	Female	500	3.336 (+1.16%)
		Male	500	3.281 (+2.58%)

DiscoGAN shows improvements of 1.16% and 2.58%, respectively, for female and male data. CycleGAN results indicate similarity to the unsupervised parallel training mode, since the results

for the female data increase 0.10%, and results for male data decrease 1.08%. Moreover, as reasoned in hypothesis (H2), the self-supervised loss influenced discoGAN up to 41.86% more than cycleGAN.

6.2 Generator Modal Effectiveness

Considering a modal approach for the training phase, the experiments utilized parallel non-parallel speech corpora, to compare obtained results in different training modes, in the attempt to understand the need for parallel corpora in the proposed GAN-based system.

The results highlighted in Tables 6.4 and 6.5 illustrate that cycleGAN and discoGAN, despite having similar training and loss functions, did not behave similarly for the considered training modes, as:

- CycleGAN shows superior objective evaluation utilizing an unsupervised non-parallel training.
- DiscoGAN shows superior objective evaluation utilizing a supervised parallel training.

As such, a discussion is made to assess if parallel corpora is necessary for the considered system, as non-parallel corpora makes training more flexible, as there is higher availability for whispered and normal speech data for non-parallel corpora. wTIMIT [69] holds 17GB of normal speech and whispered speech data, with different speakers, and phoneme-balanced and gender-balanced data. Considering the possibility of obtaining an improved dataset, the need for parallel voice conversion may not be necessary in a GAN-based WHSP2SPCH conversion.

6.2.1 Non-Parallel versus Parallel Training

For non-parallel training, cycleGAN demonstrates higher objective results, improving 17.40% when compared to the baseline model, vanilla GAN. This improvement for non-parallel training is not trivial, as it does not require a parallel corpus for training.

Table 6.6 presents the MCD results obtained dependent on the epochs. Using a constant learning rate throughout epochs displays similar results compared to obtained results using linear decrease of learning rate, as observed in the Appendix C. Despite similarities, instability of results using linear learning rate is over-viewed in table C.3. As the epochs increase to 1000, the results even get comparatively worse in comparison to 500 epochs.

CycleGAN showcases superior objective evaluation utilizing an unsupervised non-parallel training, where MCD scores are of 3.70 and 3.73, for, respectively, female and male data, at 100 epochs of training. These results improve up to 23.5% in comparison to vanilla GAN with non-parallel training.

The main problem regarding the non-parallel approach is in the bias for the remapped generated spectral envelope, as illustrated in Fig. 6.1, as an example.

Table 6.6: Mel Cepstral Distortion (MCD) for cycleGAN over specified number of epochs, considering the models are trained using 100, 250, and 500 epochs. The highlighted MCD scores reveal the best results obtained for the considered architecture.

Architecture	Training	Dataset	MCD			
			Epochs:	100	250	500
cycleGAN	Non-Parallel	Female		3.706*	4.094	4.002
		Male		3.731*	3.876	4.075
	Parallel	Female		3.950	4.012	3.969
		Male		3.819	3.889	4.206
	Parallel Supervised	Female		3.851	4.005	3.965
		Male		3.759	3.996	4.161

* Best results for the considered architecture.

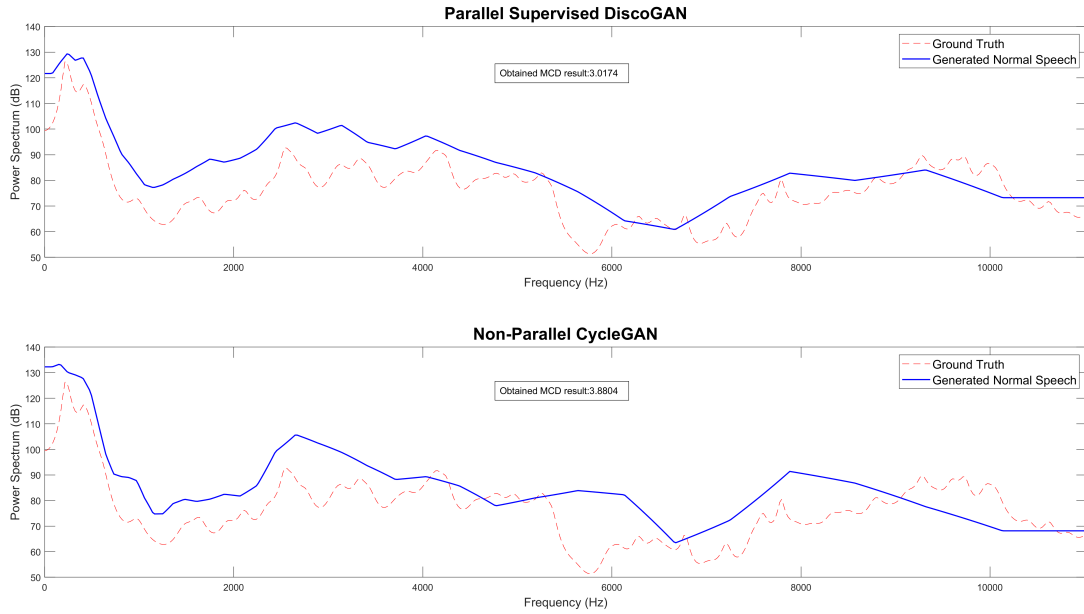


Figure 6.1: DiscoGAN versus cycle GAN: parallel supervised in comparison to non-parallel unsupervised learning, respectively

The behavior found in 6.1 is typical to the extent of observed frames from the obtained results. The main issue regarding non-parallel training is that magnitude bias mapping has a significantly higher error. Meaning that the curvature is well mapped, but the function tends to have a relative linear bias error, in terms of magnitude, as seen in Fig. 6.1.

For parallel training, cycleGAN is outperformed by discoGAN. The introduction of a supervised loss to discoGAN highlights the improvements, as obtained results increase +64.5% for female data, and +20.4% for male data, compared to the baseline, vanilla GAN in parallel training mode. Table 6.7 illustrates the behavior of discoGAN throughout training epochs.

Table 6.7: Mel Cepstral Distortion (MCD) for discoGAN over a specified number of epochs, considering that the models are trained using 100, 250, and 500 epochs. The highlighted MCD scores reveal the best results obtained for the considered architecture.

Architecture	Training	Dataset	MCD			
			Epochs:	100	250	500
discoGAN	Non-Parallel	Female		4.001	3.974	4.530
		Male		4.012	3.914	4.292
	Parallel	Female		3.533	3.505	3.375
		Male		3.398	3.446	3.368
	Parallel Supervised	Female		3.443	3.449	3.336*
		Male		3.569	3.365	3.281 *

*Best results for the considered architecture.

6.2.2 Zero-Shot Learning

For all considered modes of training, discoGAN showed the best results overall, with self-supervised parallel training, and cycleGAN showed the best results in unsupervised non-parallel training. These are the topologies of interest for zero-shot learning. Table 6.8 indicates obtained MCD results for the respective architectures.

Table 6.8: Mel Cepstral Distortion (MCD) for training data (seen vowels) considering cycleGAN in non-parallel training and discoGAN in supervised parallel training. The models were trained using 100, 250, and 500 epochs.

Architecture	Training	Dataset	MCD			
			Epochs:	100	250	500
cycleGAN	Non-Parallel	Female		3.443	3.449	3.336
		Male		3.569	3.365	3.281
discoGAN	Parallel Supervised	Female		3.443	3.449	3.336
		Male		3.569	3.365	3.281

For zero-shot learning, the considered systems - cycleGAN and discoGAN - are tested using the unseen vowels. The higher the generalization the system is capable of handling, the higher the similarity between synthetic voiced speech and the natural voiced speech. Table 6.2 displays the results obtained for zero-shot learning, where tested vowels were not seen in the training phase.

Self-supervised parallel discoGAN presented the best results, even leading to the improvement of 1.7% in comparison to results considering seen vowels, for male speakers. Figure 6.2 illustrates the generation of a normal speech spectral envelope, for an unseen vowel.

Despite the fact that the best results were obtained with DiscoGAN, it is worth noting that cycleGAN is trained in a non-parallel mode, and results are still comparable to the supervised

Table 6.9: Mel Cepstral Distortion (MCD) considering zero-shot learning (unseen vowels), for cycleGAN and discoGAN. Training used only seen vowels, and results are separated into training mode and gender-specific data. The information in brackets presents the relative variation of the zero-shot learning (unseen vowels) MCD compared to the training dataset (seen vowels). The models were trained using 500 epochs.

Architecture	Training	Dataset	Epochs	MCD
cycleGAN	Non-Parallel	Female	500	3.905 (-15.6%)
		Male	500	3.703 (-11.4%)
discoGAN	Parallel Supervised	Female	500	3.386 (-1.4%)
		Male	500	3.226 (+1.7%)

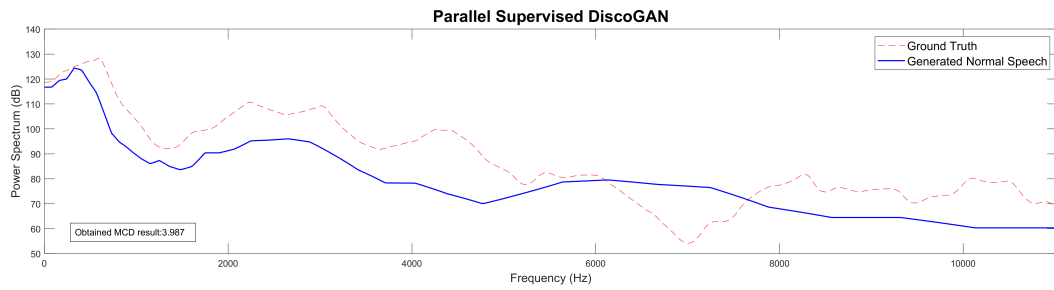


Figure 6.2: DiscoGAN - Zero-Shot Learning spectral envelope compared to the ground truth.

parallel mode in the other architecture. Figure 6.3 illustrates the generation of a normal speech spectral envelope, for an unseen vowel.

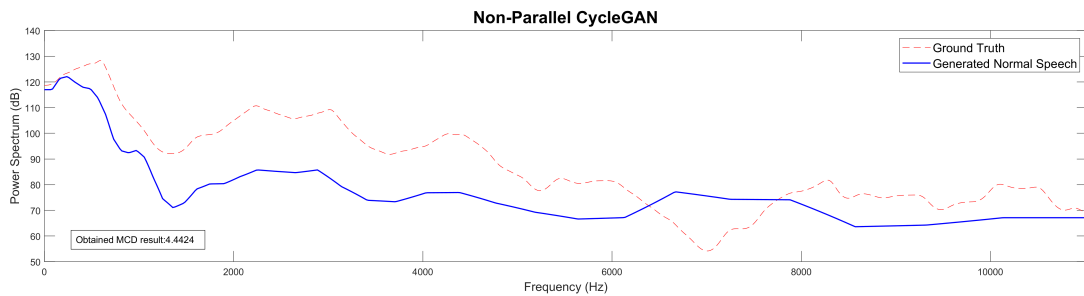


Figure 6.3: CycleGAN - Zero-Shot Learning spectral envelope compared to the ground truth.

Different vowels hold different formant shapes, as observed in Fig. 6.4. Trained vowels (circled red in Fig. 6.4) hold maximum values for the first formant (/i/ and /u/) as well as the minimum values in the first formant (/a/). The same applies for the second formant.

This led the generator and discriminator to learn marginal values for the formant shapes, for trained vowels. Thus, if the considered architectures, cycleGAN and discoGAN, are able to generalize from empirical experience, zero-shot vowels are also mapped properly, as these vowels present values for first and second formant inside the range of learned formant shapes.

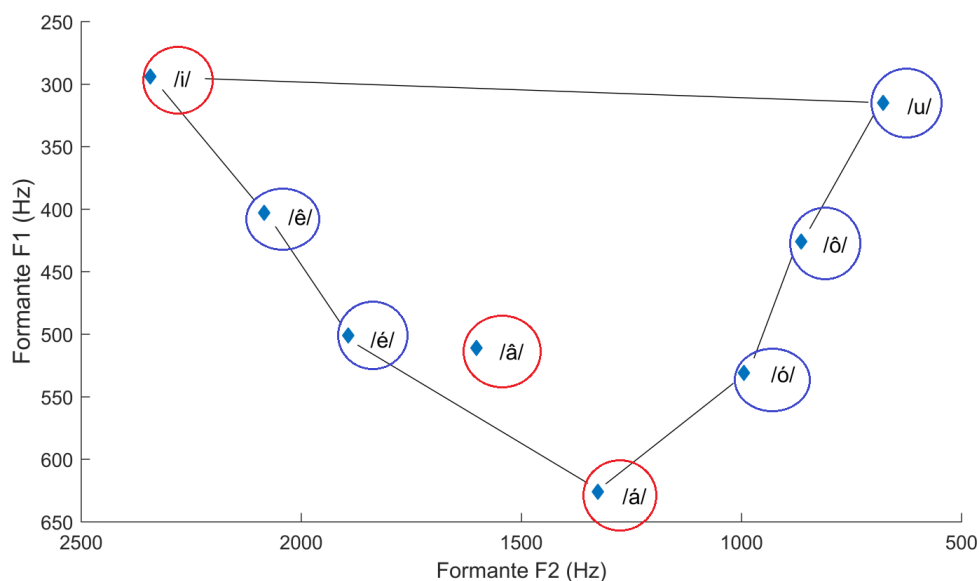


Figure 6.4: European Portuguese oral vowels positions in the vowel triangle (vowels used in the training phase are highlighted with a red circle).

Overall, results obtained in zero-shot learning (unseen vowels) are comparable to other presented results (seen vowels). As such, the generalization for both discoGAN and cycleGAN is sufficient to infer unseen oral vowels. Generalization is utilized as the ability of the implemented model to adapt to data that was not seen in training. High generalization also led to the belief that the generators do not overfit. DiscoGAN, however, presented the best overall results, even leading to the improvement of 1.7% in comparison to the results obtained by seen vowels.

6.3 Subjective Test Results

With respect to the subjective criterion, a Mean Opinion Score (MOS) is taken to measure the perceived quality of the proposed architectures, as explained in Chapter 5. The considered GAN-based architectures have shown high quality MCD results, especially the following models:

1. Unsupervised non-parallel cycleGAN trained for 500 epochs. The quality of results obtained with this non-parallel approach is not far from other considered parallel architectures. This leads to the belief that non-parallel corpora may be comparable to high quality parallel WHSP2SPCH conversion.
2. self-supervised parallel DiscoGAN trained for 500 epochs. This architecture demonstrated the best obtained experimental results considering the objective criterion. However, self-supervised discoGAN requires self-supervision and parallel corpora, demanding additional requirements for the system, such as phonetic alignments, in contrast to non-parallel unsupervised cycleGAN.

The two highlighted GAN-based models were used, in order to understand how unsupervised non-parallel corpora subjectively compares to the high quality self-supervised parallel corpora in WHSP2SPCH conversion.

DyNaVoiceR currently presents rule-based solutions for parametric spectral mapping. To compare the GAN-based models with the DyNaVoiceR rule-based approach, these models were employed in the auditory test as:

1. DyNaVoiceR parametric spectral mapping, with no spectral models or templates, considering only a direct mapping function between whispered speech and normal speech, i.e. using simply amplified whispered spectral envelopes.
2. DyNaVoiceR parametric spectral mapping, considering spectral templates based on the original dataset pertaining to voiced vowels, that were previously saved and used accordingly.

The introduction of highlighted DyNaVoiceR models in the subjective test also allows for the comparison between rule-based models and machine learning-based models.

For the fundamental frequency and harmonic components of the signal, since the current project did not focus on the specified components, all the models present in the subjective task utilize the same fundamental frequency contour and the same periodic component. Thus, the models used only diverge in their respective spectral envelopes, allowing analysis on how the spectral envelope modeling affects the quality of synthetic speech and its perception.

The implemented models were categorized according to Table 6.10. The instructions given to the subjects did not specify the description of the tested models, nor their respective characteristics. As such, each model is only represented by a single letter - A, B, C, and D.

Table 6.10: MOS models: model specifications, and respective nomenclatures used.

Model	Description
A	DyNaVoiceR parametric spectral mapping, without spectral models or templates, considering only a direct mapping function between whispered speech and normal speech
B	DyNaVoiceR parametric spectral mapping, considering spectral templates based on the original dataset, that were previously saved and used accordingly
C	cycleGAN, considered an unsupervised non-parallel training mode
D	discoGAN, considering a supervised parallel training mode

The evaluation comprises two questions. The first question requires the subject to classify the quality in regard to the presence of audible artifacts in the audio samples, following the scale explained in Table 5.12. The second question requires evaluating proximity of the audio sample to an audio reference of normal speech, following the scale explained in Table 6.11.

Both questions query the same three auditory tasks. An auditory task is an audible activity where test subjects are given explicit instructions to listen to two audio references. Each reference is a female speaker vocalizing the word comprised in the task, in both whispered and normal

speech. After this process, the auditory task gives subjects instructions to classify the considered models (models A, B, C, and D), in accordance to the related question at hand (presence of artifacts or proximity to the normal speech reference). Moreover, subjects were instructed to, if possible, use headphones to increase the audio quality.

To clarify, Table 6.11 illustrates the two questions presented, as well as respective auditory tasks. Also, the words uttered in each task are also present.

Table 6.11: MOS questions, auditory task, and the uttered word for each task.

Question	Task	Word
Quality	1	'face'
	2	'pica'
	3	'velho' *
Proximity	4	'face'
	5	'pica'
	6	'velho' *

* Vowel /e/ is not seen in the training phase for the GAN-based models.

Auditory tasks 1 and 2 evaluated the quality of synthetic speech for seen training vowels in GAN-based models (models C and D) and the quality of synthetic speech for rule-based models (models A and B). For the auditory task 3, one vowel unseen in the training (/e/) was present, thus evaluating zero-shot learning for models C and D. Models A and B are not training-dependent, as such, the concept of seen and unseen vowels do not apply to these models.

Auditory tasks 4 and 5 evaluated proximity to the normal speech audio reference, for seen training vowels in GAN-based models (models C and D). Similarly to the auditory task 3, task 6 presents one vowel unseen in training (/e/). As explained, the concept of seen and unseen vowels do not apply to models A and B.

Overall, tasks 1, 2, 4, and 5 evaluated synthetic voiced speech of seen training vowels for GAN-based models, and tasks 3 and 6 evaluated synthetic voiced speech of one seen vowel and one unseen vowel for GAN-based models.

A total of 20 subjects ($N = 20$), 55% male ($n=11$), between 19 and 51 years of age ($M = 25.1$; $STD = 8.5$) took the perceptual test. No subject reported hearing loss or any related problems that may hinder results. Moreover, a total of 40% ($n=8$) reported to have had musical education.

6.3.1 Artifacts and Speech Quality

Regarding the first question, the subject were asked to classify the quality of synthetic speech in regard to the presence of audible artifacts in the audio samples, following the scale explained in Table 5.12. Tasks 1 and 2 evaluated the synthetic voiced speech of training data, for GAN-based models (models C and D), and the respective results are present in Fig. 6.5.

The obtained results for tasks 1 and 2 indicate that both cycleGAN and discoGAN (models C and D) are comparable to the DyNaVoiceR template-based approach (model B). Moreover, the

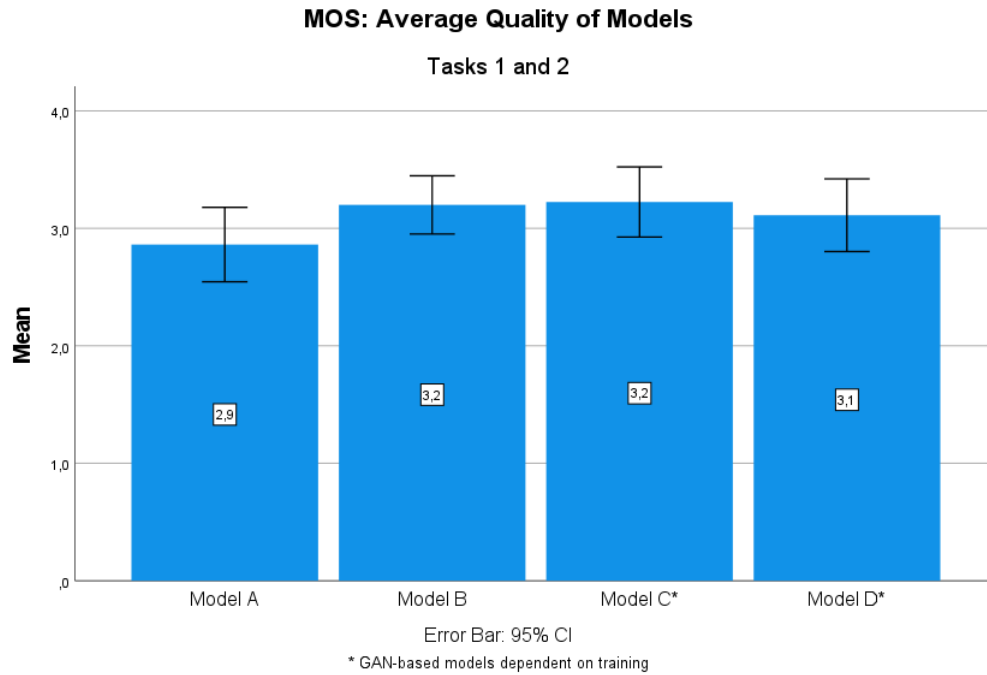


Figure 6.5: MOS average results for tasks 1 and 2: synthetic voiced speech of vowels seen in training, with respect to the presence of undesired artifacts, with 95% confidence intervals.

DyNaVoiceR linear model (model A) shows an average score lower than other considered models. Despite this, differences between average scores are not considered significant, as confidence intervals present high overlap.

For the GAN-based architectures (models C and D), task 3 presents the uttered word /velho/, that is constituted by one seen vowel, /o/, and one unseen vowel, /e/. As such, this task evaluates perceived quality for zero-shot learning. Figure 6.6 illustrates obtained results for task 3.

As indicated in Fig. 6.6, cycleGAN and discoGAN decreased to a score of 2.2. Moreover, the rule-based models also decreased to 2.0, and 2.2, for models A and B, respectively. Overall, all models decreased in perceived quality in comparison to tasks 1 and 2. However, rule-based models did not rely on the distinction of training data (seen vowels) and zero-shot learning data (unseen vowels). Results decreased in average, but standard deviation (STD) remained similar for all modes. This leads to the main hypothesis, that:

All models decreased in perceived quality in comparison to tasks 1 and 2.

Therefore, the chosen word for tasks 3 (and task 6) - /velho/ - may be, in (H3) essence, less intelligible and prone to error in synthesis.

Overall, despite the decrease in average scores for the task 3, similarities are found in both discoGAN and cycleGAN with respect to the presence of artifacts. DiscoGAN showed a mean result of 3.31, while cycleGAN showed a mean result of 3.23, considering seen training vowels as explicit in table 6.12. Moreover, model A was outperformed by other models in all tasks. Overall,

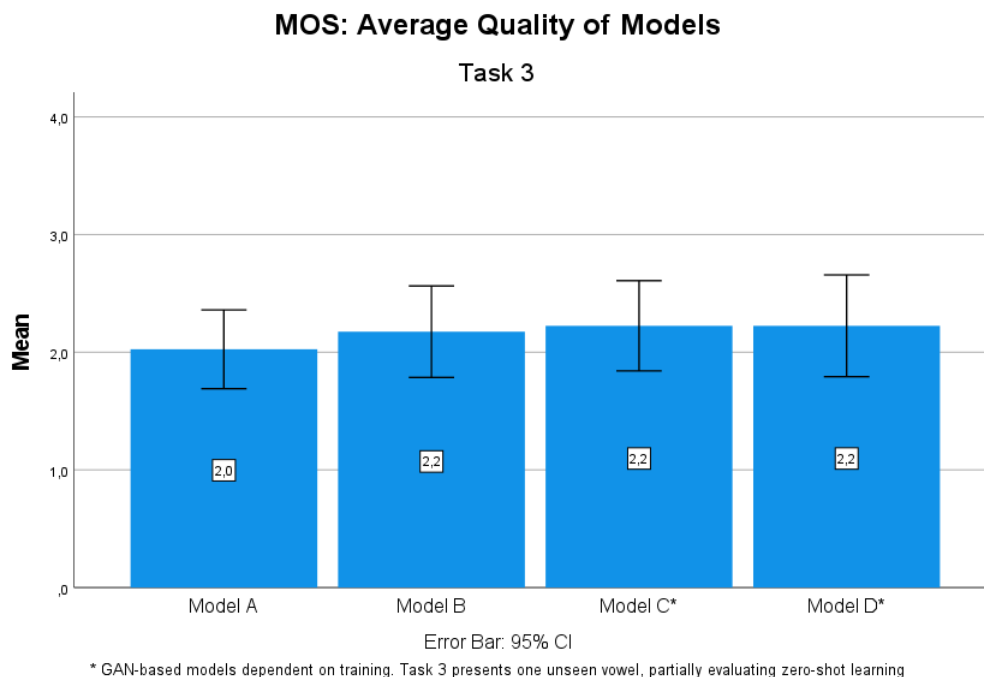


Figure 6.6: MOS average results for task 3: synthetic voiced speech of one seen vowel and an unseen vowel for GAN-based models (models C and D), with respect to the presence of undesired artifacts, with 95% confidence intervals.

similar results from both cycleGAN and discoGAN were obtained, and both models showed comparable results to the template-based model (model B). Despite this, differences between average scores are not considered significant, as confidence intervals present high overlap.

Table 6.12: Mean Opinion Score (MOS) results obtained considering presence of artifacts, for cycleGAN and discoGAN, regarding seen and unseen vowels, i.e. training and zero-shot learning vowels

Model	Data	Min	Max	Mean	STD
cycleGAN	Seen	2.0	4.0	3.23	0.64
	Unseen	1.0	4.0	2.23	0.82
discoGAN	Seen	2.0	4.3	3.31	0.66
	Unseen	1.0	4.5	2.23	0.92

6.3.2 Proximity and Speech Naturality

Regarding the second and last question, subjects were instructed to evaluate the proximity of the synthetic speech to the normal speech reference, following the scale explained in Table 6.11. Tasks 4 and 5 evaluated the synthetic voiced speech of training data, for GAN-based models (models C and D), and the respective results are present in Fig. 6.7.

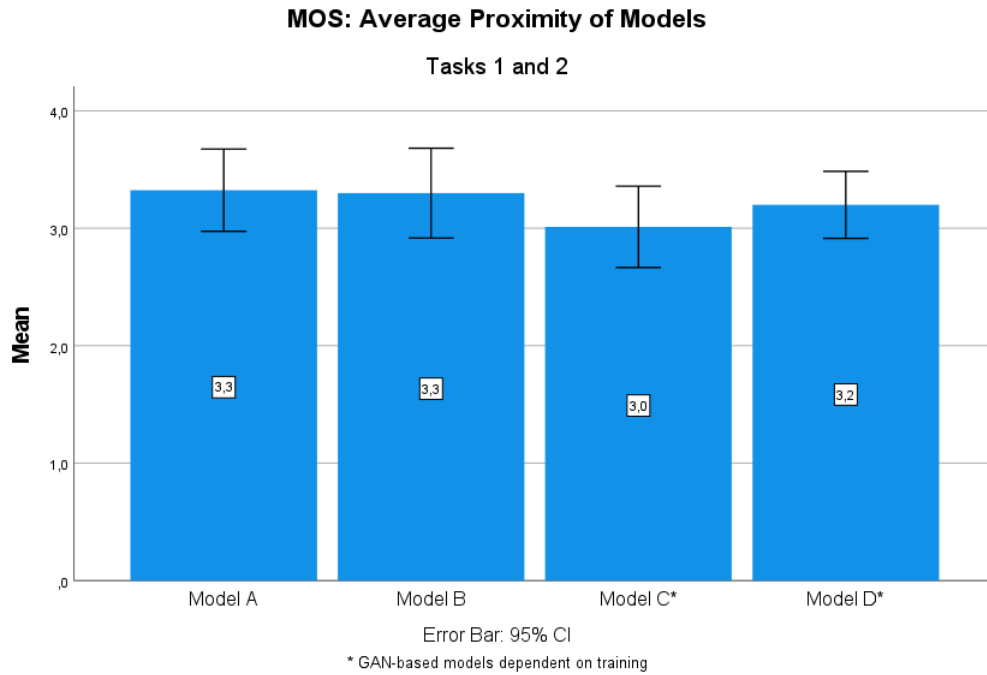


Figure 6.7: MOS average results for tasks 4 and 5: synthetic voiced speech of vowels seen in training, with respect to the proximity to the normal speech reference (ground truth), with 95% confidence intervals.

Surprisingly, the rule-based conversion applied by the linear transformation (model A) outperformed the other modes in terms of perceived naturalness with respect to proximity to the normal speech audio reference. This mode, which produces a nasal effect on the converted speech samples, tended to be evaluated unfavorably by the subjects with respect to the presence of artifacts. However, perceived results are lower than expected for the machine learning-based models (models C and D). Despite this, differences between average scores are not considered significant, as confidence intervals present high overlap.

Task 6 presents the uttered word /velho/, that is constituted by one seen vowel, /o/, and one unseen vowel, /e/, for GAN-based models. Figure 6.8 indicates obtained results for task 6.

The results obtained indicate that machine learning-based systems score lower than expected, with $MOS < 3$. However, rule-based models do not rely on training, but also show lower results compared to the other tasks. Moreover, similar to task 3, task 6 indicates an average decrease of perceived proximity for all modes, while maintaining similar STD. Therefore, the proposed hypothesis (H3) that the chosen word for the task - /velho/ - is, in essence, less intelligible and prone to error in synthesis, is assumed to be correct.

Lastly, machine learning-based systems (models C and D) show that parallel corpora differentiated results in terms of perceived proximity. CycleGAN shows a mean score of 3.01, while discoGAN outperforms cycleGAN with a mean of 3.2. The results obtained are explicit in Table 6.13.

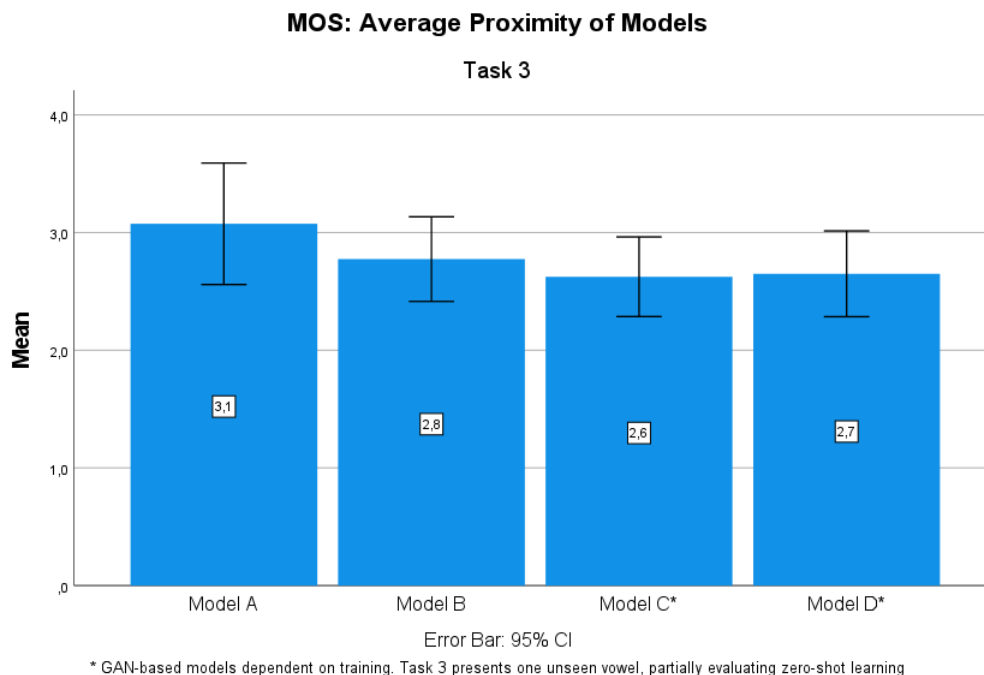


Figure 6.8: MOS average results for task 6: synthetic voiced speech of one seen vowel and an unseen vowel for GAN-based models (models C and D), with respect to the proximity to the normal speech reference (ground truth), with 95% confidence intervals.

Table 6.13: Mean Opinion Score (MOS) results obtained considering proximity to the normal speech reference, for cycleGAN and discoGAN, regarding seen and unseen vowels, i.e. training and zero-shot learning vowels

Model	Data	Min	Max	Mean	STD
cycleGAN	Seen	1.5	4.5	3.01	0.74
	Unseen	2.0	4.5	2.63	0.73
discoGAN	Seen	2.0	4.5	3.20	0.61
	Unseen	1.5	4.5	2.65	0.61

6.4 Summary

To review the concepts analyzed in the present chapter, the experiments derived in previous sections were as follows:

- MCD analysis for cycleGAN and discoGAN, with vanilla GAN acting as baseline.
- MCD analysis for cycleGAN and discoGAN for zero-shot learning vowels.
- Subjective Mean Opinion Score (MOS) evaluation and discussion of obtained results.

First, CycleGAN showed superior objective evaluation utilizing an unsupervised non-parallel

training, where MCD results improved up to 23.5% in comparison to vanilla GAN with non-parallel training. Second, for parallel training, cycleGAN was outperformed by discoGAN. The introduction of a supervised loss to discoGAN highlighted the improvements, as the results obtained increased up to +64.50% in comparison to the baseline vanilla GAN with parallel training. Moreover, zero-shot learning has shown surprisingly positive objective results in MCD scores, where cycleGAN and discoGAN showed similar results for training vowels and zero-shot vowels.

The considered GAN-based architectures have shown high quality MCD results, especially the following models:

1. DiscoGAN - supervised parallel training mode.
2. CycleGAN - unsupervised non-parallel training mode.

The decision to test subjectively the two highlighted models was to compare the perceptual difference between a parallel supervised setting and a non-parallel unsupervised setting. The obtained results indicate that machine learning-based models are similar to the template-based model, with respect to the presence of artifacts. Additionally, CycleGAN is indicated as preferable in comparison to discoGAN. As for the proximity to the normal speech reference, the rule-based models performed the best, followed by discoGAN, that outperformed discoGAN. Despite this, differences between obtained scores were not considered significant, as confidence intervals presented a high overlap.

Subjective zero-shot learning tasks, tasks 3 and 6, showed a marginal decrease in perceived quality for all models considered, where MCD scores decrease on average by 0.9 score points, maintaining a similar standard deviation (STD). Despite inconclusive, these tasks were assumed to be prone to subjective dissatisfaction.

Overall, for the DyNaVoiceR project, self-supervised DiscoGAN with parallel training is viable and highly effective. An existing framework retrieves previously specified phonemes. As such, phoneme alignment for training is straightforward and parallel corpora is retrieved directly. Moreover, objective results show excellent results for seen and unseen vowels, and the self-supervised loss shows room for improvement. Perceptual results also show that discoGAN obtains similar scores to the DyNaVoiceR template-based approach, in terms of proximity to the normal speech reference.

Chapter 7

Conclusions

The present dissertation explored GAN-based technologies in WHSP2SPCH conversion, as a countermeasure for patients with aphonia. The experience of living with a voice disorder may impact several dimensions of everyday life, such as social and occupational activities. Furthermore, it is rather frequent for people with these pathologies to battle with depression, anxiety, isolation, anger or other types of psychosocial difficulties [70, 71]. Bearing this in mind, this dissertation explored the spectral envelope conversion from whispered speech to its normal speech counterpart.

The literature for WHSP2SPCH conversion mainly deals with two issues. First, F_0 prediction is still a drawback, as the lack of objective results and metrics make the task difficult to approach. Second, GAN stability and optimization is a drawback due to the lack of theoretical formulation for cross-domain learning. However, further developments are expected in the area of cross-domain learning, specifically mapping functions for spectral and cepstral features, in order to obtain a statistically significant natural-sounding conversion.

This dissertation focused in WHSP2SPCH conversion, specifically on spectral envelope remapping from whispered speech to its normal speech counterpart. Throughout this work:

- The DyNaVoiceR dataset was explored, described, and analyzed, in order to create a dataset containing nine oral vowels from the European Portuguese language. The retrieved data was further separated into training, validation, and testing datasets.
- The process of obtaining spectral envelope models from whispered and normal speech domains was explained, and the respective differences between the domain spectral envelopes were emphasized and discussed.
- The process of converting spectral envelope models into cepstral features that act as input features to the generators in GAN-based architectures was explained.
- The implemented architectures, cycleGAN, discoGAN, and vanillaGAN were explained in detail, according to their topology, loss functions, optimizers, and hyperparameters.
- A study was proposed considering different approaches: training the proposed architectures with parallel corpora and non-parallel corpora, and with gender and non-gender specific

data. These approaches allowed a deeper understanding of the viability of non-parallel corpora in high-quality on-the-fly WHSP2SPCH conversion and highlighted differences in results according to gender. Lastly, zero-shot learning was used to understand the capacity of the generators in terms of generalization.

- The results obtained were analyzed and discussed in accordance with the validation metrics applied in the proposed experiments.

The main conclusions of the current dissertation are summarized in the next section.

7.1 Conclusion Synthesis

The solution presented in this dissertation relied on Generative Adversarial Networks for the spectral envelope remapping in the context of WHSP2SPCH, namely on the cycleGAN and discoGAN architectures. To preserve spectral information, the normal speech spectral envelope was estimated, including analytical tools that obtain the relevant peaks in spectral magnitude. The spectral envelope was derived using linear interpolation based in the predicted spectral peaks of the harmonic structure (in dB), using the same tools as previously (Wiener-Khintchine theorem and Levinson-Durbin algorithm) in Chapter 5. The accuracy of the obtained spectral envelopes was improved, as less smoothing effect occurred. Moreover, the use of the cepstral domain showed significant improvements when compared to the previous approaches, considering 22 LPC coefficients and 64 spectral bins.

The obtained results showed that cycleGAN is better suited for non-parallel training, as MCD results improved up to 23.5% when compared to the baseline vanilla GAN. This model presented the best obtained results for non-parallel training regarding all non-parallel architectures. Despite this, the self-supervised discoGAN with parallel corpora showed an increase in MCD results up to 64.5%. This model presented the highest objective score of all considered architectures. Subjective results indicate that cycleGAN trained with non-parallel corpora was perceived similarly to the self-supervised discoGAN trained with parallel-corpora in regard to converted speech sample quality. However, results showed that discoGAN is significantly preferred over cycleGAN in terms of proximity to the original normal speech.

Concerning the limitations of the current study, the dataset only considered nine oral vowels, and only four of those were used in the training phase. Moreover, for the subjective tests, the sample size ($N=20$) and the lack of diversity in sociodemographic variables may reduce the power of the study and increase the margin of error. Also, subjective results showed overlapping confidence intervals and similar average scores. Thus, no significant results were derived from the subjective test. Considering the objective metrics, Mel Cepstral distortion depends on the underlying MFCC implementation. As such, objective results were not comparable to the literature [9, 11, 10].

Lastly, the obtained results support the viability of the tested models for on-the-fly cross-domain learning in the context of whisper to normal speech conversion. CycleGAN has shown

satisfactory results. However, the self-supervised parallel discoGAN is more suitable for the DyNaVoiceR project. In this context, this architecture is highly effective. The DyNaVoiceR corpora is manually annotated, simplifying parallel training and enabling higher quality results. Also, perceptual results showed that discoGAN obtained similar scores to the DyNaVoiceR template-based approach, in terms of proximity to the normal speech reference, implying that discoGAN behaves similarly to the ground truth. Nevertheless, non-parallel cycleGAN highlighted optimistic possibilities and the potential for a natural-sounding conversion, when no parallel corpora is available. Taking together the research and the presented results, the applicability of machine learning for whisper-to-normal speech conversion highlights the need for future research and developments, ambitioning the possibility that aphonia patients may rely on technology to improve their means of communication. Thereby, we intended to provide a solution for the problem that carries such psychological burden and that impacts interpersonal relationships.

7.2 Future Work

In order to improve the developed work, based on the conclusions derived and the limitations of the proposed system, the following future work is proposed:

- Expanding phonemes in training, since the current dissertation focused only on oral vowels, while training only four of them. As the results show that the proposed architectures are viable for the framework, the extension of other vowels and the inclusion of voiced consonants should be studied and attempted.
- Increase the size of the DyNaVoiceR dataset and the balance between phoneme occurrences, as currently the dataset shows substantially higher number of occurrences for a specified group of oral vowels. Expanding the data to include more phonemes would increase the utility of the proposed GAN-based architectures. Moreover, with this approach, semi-supervised learning may also be attempted. As the DyNaVoiceR current corpora is annotated, the inclusion of non-annotated corpora should be viable.
- Improve and restructure feature extraction. This dissertation focused on improving the quality of the spectral envelope, especially considering normal speech. Despite this, improvements may still be made, as MFCCs still lack correct shapes in the third and fourth formants.
- Implementation of fundamental frequency (F_0) prediction. The current dissertation only tackled spectral envelope remapping in detail. As such, fundamental frequency allied to a high quality spectral envelope remapping should result in improvements in speech intelligibility and naturality. The framework built in this dissertation is expandable and can include F_0 prediction.
- Few-shot learning is also aligned with the DyNaVoiceR project and should be further studied.

Appendix A

DyNaVoiceR Dataset Tasks

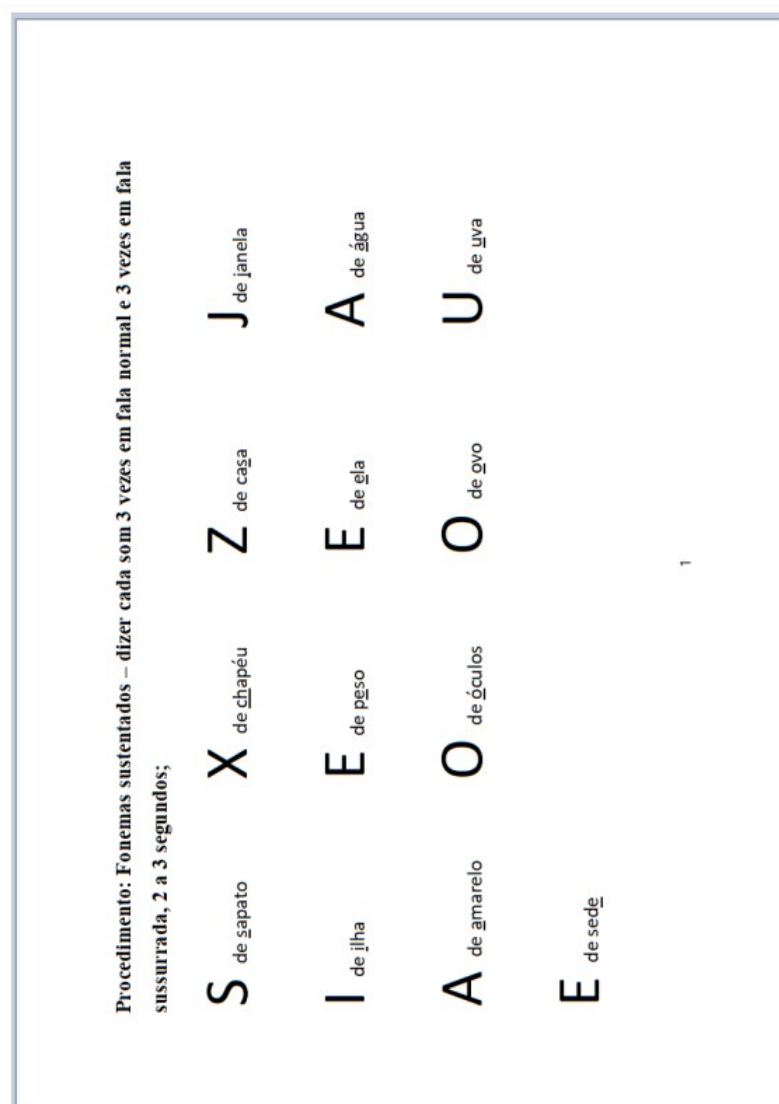


Figure A.1: Page 1 of the DyNaVoiceR tasks

Procedimento: Lettura de palavras – dizer cada palavra 3 vezes em fala normal e 3 vezes em fala sussurrada, pela ordem apresentada;

Col. 1	Col. 2	Col. 3	Col. 4	Col. 5	Col. 6
Nuca	Ripa	Sala	Zaro	Asa	Velho
Lupa	Laje	Juba	Assa	Rita	Hora
Chiba	Face	Chama	Luta	Haja	Buda
Minha	Pica	Vida	Jarra	Ache	Viga
Guga	Vaze	Acha	Fisga		

Procedimento: Lettura de frases –ler cada frase 3 vezes em fala normal e 3 vezes em fala sussurrada;

Figure A.2: Page 2 of the DyNaVoiceR tasks

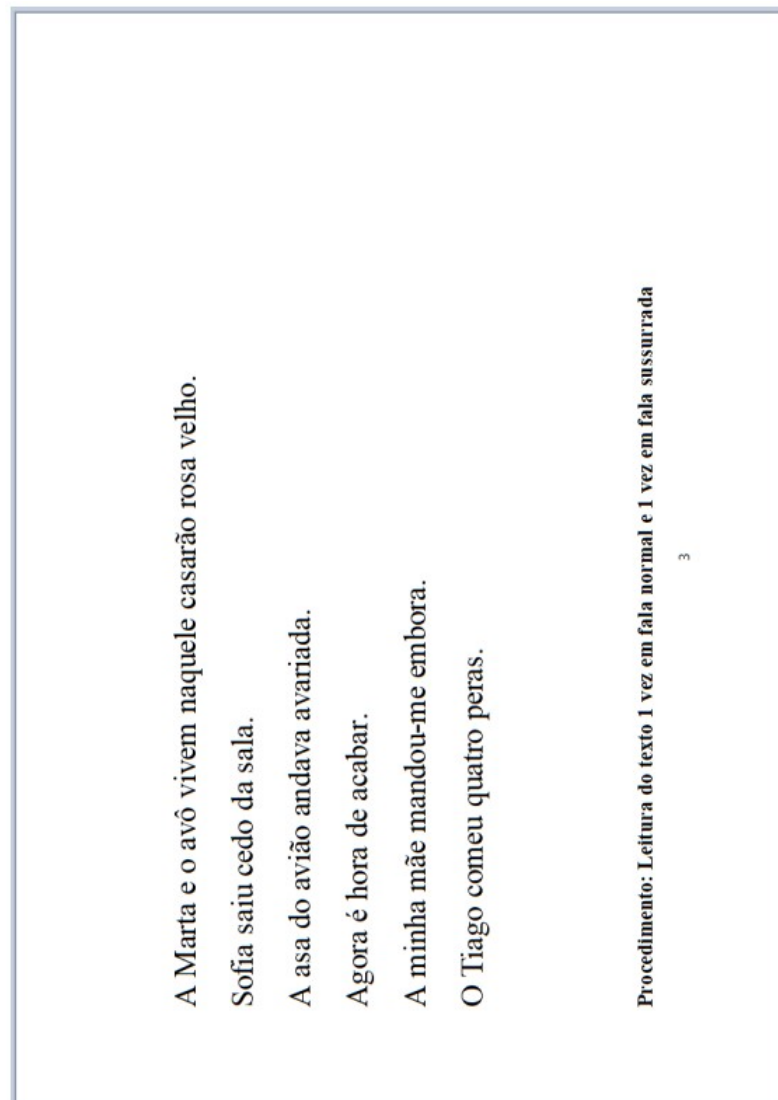


Figure A.3: Page 3 of the DyNaVoiceR tasks

Appendix B

Preliminary Study I: LPC-based Spectral Mapping

Two experiments were executed considering:

1. 64 bins for compressed representation of the spectral envelope.
2. 22 Levinson-Durbin coefficients representing autoregressive linear process coefficients.
3. WHSP2SPCH Conversion - Ground Truth.

B.1 64 Point Representation

The first trials of testing were done using 64 bins to represent framed input spectral envelopes. The graphical outputs are described in the figure [B.1](#).

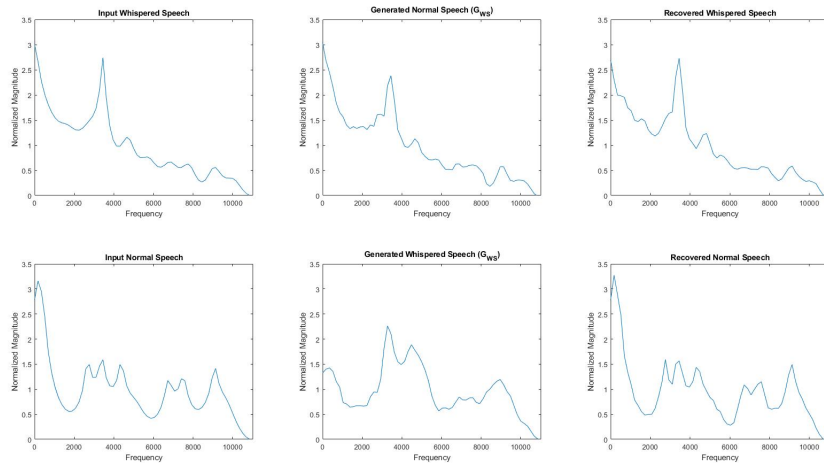


Figure B.1: CycleGAN - Parallel Training for 64 bins per frame

The obtained results are indicative of mode collapse tendency, as normal speech formants could not be learned from the whispered speech domain counterparts. Figure [B.2](#) shows a

generated normal speech envelope from a whispered speech exemplar and a generated whispered speech from a normal speech exemplar. As the figure illustrate, the original formant shapes and locations tend to be kept similar to their originals, particularly with regard to the whisper to normal speech conversion.

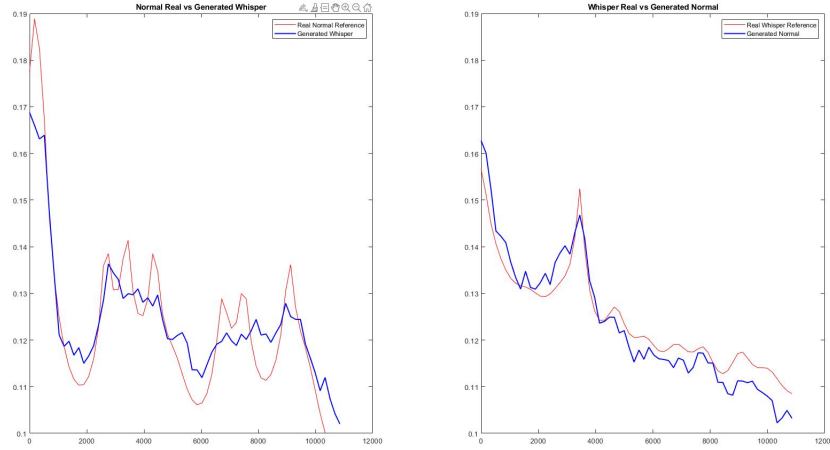


Figure B.2: CycleGAN - Comparison from input for 64 bins per frame

B.2 22 Levinson Coefficients

Considering this, the next approach was to base the conversion on 22 LPC coefficients, derived from Levinson-Durbin Recursion. Figure B.3 illustrates the now improved mapping of spectral features. Disregarding reconstruction instability, this approach has tremendous improvement.

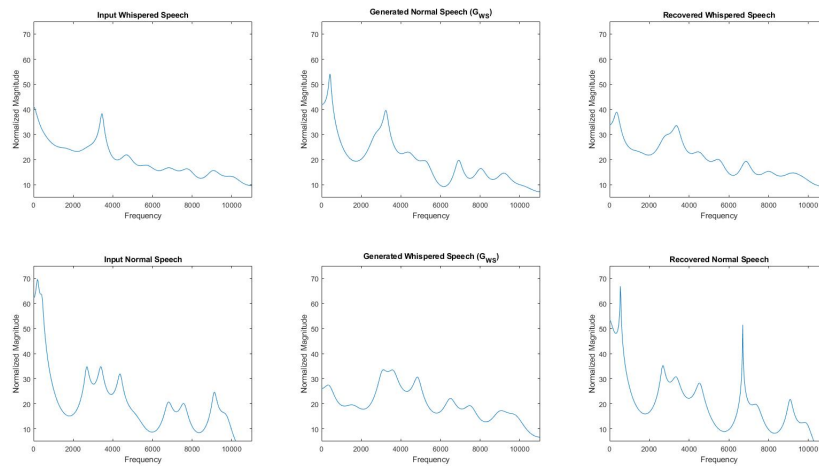


Figure B.3: CycleGAN - Comparison from input for 22 LPC coefficients per frame

Despite this, further testing using MFCCs showcases that the generator converges utilizing the cepstral domain, but no utilizing mfccs, with the current topology. Figure B.4 represents the loss of the generator and discriminator over periods of training. represents the same losses, utilizing MFCCs instead of LPC coefficients.

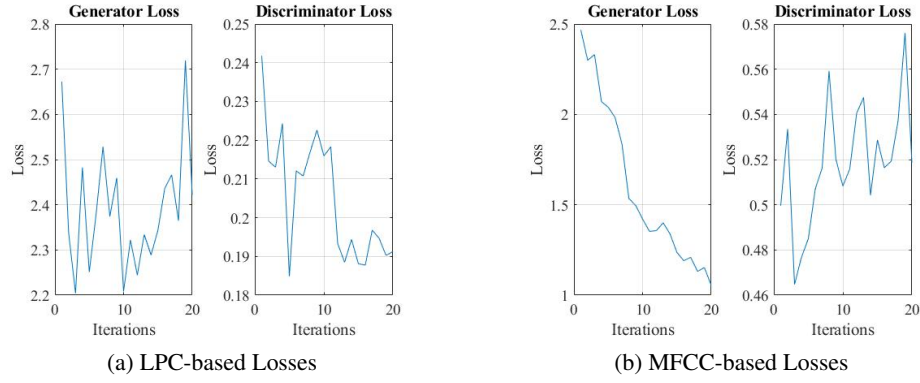


Figure B.4: CycleGAN - Loss given different inputs

Also, the MCD results clarify that MFCC tend to have more stability in testing scenarios, where the spectral envelopes derived from both methods highlight the superiority of MFCCs. Table B.1 displays results obtained in testing scenarios, for unseen frames.

GAN Arichitecture	LPC	MFCC
cycleGAN	8.190	3.704

Table B.1: MCD analysis of LPC versus MFCC methods, for cycle-GAN with Parallel Training

Appendix C

Results: Constant Learning Rate

Table C.1: Mel Cepstral Distortion (MCD) for the baseline model, Vanilla GAN, and utilized architectures. Training utilizes 500 epochs, and results are separated into training type and speaker gender. Results consider a constant learning rate.

Architecture	Training	Dataset	Epochs	MCD*
Baseline	Non-Parallel	Female	500	4.845
		Male	500	4.769
		All Speakers	500	7.309
	Parallel	Female	500	5.488
		Male	500	4.121
		All Speakers	500	5.040
cycleGAN	Non-Parallel	Female	500	3.792
		Male	500	3.816
		All Speakers	500	3.870
	Parallel	Female	500	3.839
		Male	500	3.636
		All Speakers	500	4.012
discoGAN	Non-Parallel	Female	500	4.562
		Male	500	4.028
		All Speakers	500	5.521
	Parallel	Female	500	3.381
		Male	500	3.303
		All Speakers	500	4.415

* Constant Learning Rate (lr = 0.0001)

Inconsistency in results is analyzed in table [C.3](#).

Table C.2: Inconsistencies in Mel Cepstral Distortion (MCD) for the proposed training epochs, considering 400, 500, and 1000 epochs. Inconsistencies are defined as decreases in MCD for an increasing number of epochs, or increases in MCD for a decreasing number of epochs.

Architecture	Training	Dataset	MCD			
			Epochs	400	500	1000
cycleGAN	Non-Parallel	Female		3.898	3.792	3.568
		Male		3.787*	3.816	3.514
	Parallel	Female		3.731*	3.839	3.614
		Male		3.655	3.636	3.498
	Parallel supervised	Female		3.695	3.656	3.689
		Male		3.854	3.766	3.594
discoGAN	Non-Parallel	Female		4.023*	4.562	4.882*
		Male		6.904	4.028	3.789
	Parallel	Female		3.510	3.381	3.624*
		Male		3.521	3.303	3.126
	Parallel supervised	Female		3.307	3.244	3.335*
		Male		3.518	3.301	3.455*

* Inconsistencies

* Considering gradient descent, MCD should decrease as epochs increase

Table C.3: Increase in stability utilizing linear descent learning rate, measured in Mel Cepstral Distortion (MCD) for the proposed training epochs, considering 100, 250, and 500 epochs.

Architecture	Training	Dataset	MCD			
			Epochs	100	250	500
CycleGAN	Non-Parallel	Female		3.706	4.094	4.002
		Male		3.731	3.876	4.075
	Parallel	Female		3.950	4.012	3.969
		Male		3.819	3.889	4.206
	Parallel Supervised	Female		3.695	3.656	3.689
		Male		3.709	3.996	4.161
DiscoGAN	Non-Parallel	Female		4.001	3.974	4.530
		Male		4.012	3.914	4.292
	Parallel	Female		3.533	3.505	3.375
		Male		3.398	3.446	3.368
	Parallel Supervised	Female		3.443	3.449	3.336
		Male		3.569	3.365	3.281

Appendix D

Results: Batch Size

Table D.1: Mel Cepstral Distortion (MCD) for the proposed 500 training epochs. Results for batch size $B = 100$, compared to $B = 200$, for all speakers.

Architecture	Training	Batch Size	Epochs	MCD*
cycleGAN	Non-Parallel	100	500	4.355
cycleGAN	Non-Parallel	200	500	5.561 *
discoGAN	Parallel Self-Supervised	100	500	3.487
discoGAN	Parallel Self-Supervised	200	500	4.036 *

* Decrease in performance

References

- [1] Finnian Kelly. *Automatic Recognition of Ageing Speakers*. PhD thesis, 04 2014.
- [2] Marco António da Mota Oliveira. Modelização de filtro de trato vocal para reconstrução de voz disfónica. Master's thesis, Faculdade de Engenharia da Universidade do Porto, 2020.
- [3] Ian Vince McLoughlin. *Speech and Audio Processing: A MATLAB®-based approach*. Cambridge University Press, 2016. URL: <https://www.cambridge.org/core/books/speech-and-audio-processing/736E38B7690AE09BF7513D0B652AD272>, doi:10.1017/CBO9781316084205.
- [4] Mfcc technique for speech recognition - analytics vidhya. URL: <https://www.analyticsvidhya.com/blog/2021/06/mfcc-technique-for-speech-recognition/>.
- [5] Daniel Michelsanti and Zheng-Hua Tan. Conditional generative adversarial networks for speech enhancement and noise-robust speaker verification. In *Interspeech 2017*. ISCA, aug 2017. URL: <https://doi.org/10.21437%2Finterspeech.2017-1620>, doi:10.21437/interspeech.2017-1620.
- [6] Sergios Theodoridis. *Machine learning : A Bayesian and Optimization Perspective*. 2nd edition, 2020.
- [7] Lars Hertel, Erhardt Barth, Thomas Kaster, and Thomas Martinetz. Deep convolutional neural networks as generic feature extractors. *Proceedings of the International Joint Conference on Neural Networks*, 2015-September, 10 2017. URL: <https://arxiv.org/abs/1710.02286v1>, doi:10.1109/IJCNN.2015.7280683.
- [8] Ian Goodfellow. Nips 2016 tutorial: Generative adversarial networks. 12 2016. URL: <https://arxiv.org/abs/1701.00160v4>.
- [9] Maitreya Patel, Mirali Purohit, Jui Shah, and Hemant A. Patil. Cinc-gan for effective f0 prediction for whisper-to-normal speech conversion. 8 2020. URL: <http://arxiv.org/abs/2008.07788>.
- [10] Mihir Parmar, Savan Doshi, Nirmesh J. Shah, Maitreya Patel, and Hemant A. Patil. Effectiveness of cross-domain architectures for whisper-to-normal speech conversion. pages 1–5, 2019. doi:10.23919/EUSIPCO.2019.8902961.
- [11] Nirmesh Shah, Mihir Parmar, Neil Shah, Nirmesh J Shah, and Hemant A Patil. Novel mmse discogan for cross-domain whisper-to-speech conversion, 2018. URL: <https://www.researchgate.net/publication/326668619>.

- [12] E. Zwicker. Subdivision of the audible frequency range into critical bands (frequenzgruppen). *The Journal of the Acoustical Society of America*, 2005. URL: <https://asa.scitation.org/doi/abs/10.1121/1.1908630>, doi:10.1121/1.1908630.
- [13] Dynavoicer project web page. URL: https://paginas.fe.up.pt/~voicestudies/dynavoicer/?page_id=24.
- [14] Hailun Lian, Yuting Hu, Weiwei Yu, Jian Zhou, and Wenming Zheng. Whisper to normal speech conversion using sequence-to-sequence mapping model with auditory attention. *IEEE Access*, PP:1–1, 09 2019. doi:10.1109/ACCESS.2019.2940700.
- [15] Tomoki Toda, Alan Black, and Keiichi Tokuda. Voice conversion based on maximum-likelihood estimation of spectral parameter trajectory. *Audio, Speech, and Language Processing, IEEE Transactions on*, 15:2222 – 2235, 12 2007. doi:10.1109/TASL.2007.907344.
- [16] Marius Cotescu, Thomas Drugman, Goeric Huybrechts, Jaime Lorenzo-Trueba, and Alexis Moinet. Voice conversion for whispered speech synthesis. *IEEE Signal Processing Letters*, 27:186–190, 2020. doi:10.1109/LSP.2019.2961213.
- [17] Berrak Sisman, Junichi Yamagishi, Simon King, and Haizhou Li. An overview of voice conversion and its challenges: From statistical modeling to deep learning. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 29:132–157, 2021. doi:10.1109/TASLP.2020.3038524.
- [18] Aamir Wali, Zareen Alamgir, Saira Karim, Ather Fawaz, Mubariz Barkat Ali, Muhammad Adan, and Malik Mujtaba. Generative adversarial networks for speech processing: A review. *Computer Speech & Language*, 72:101308, 3 2022. doi:10.1016/J.CSL.2021.101308.
- [19] Keith Rayner and Charles Clifton. Language processing in reading and speech perception is fast and incremental: Implications for event-related potential research. *Biological Psychology*, 80(1):4–9, 2009. Before the N400: Early Latency Language ERPs. URL: <https://www.sciencedirect.com/science/article/pii/S0301051108001245>, doi:<https://doi.org/10.1016/j.biopsycho.2008.05.002>.
- [20] Angela D. Friederici. The brain basis of language processing: From structure to function. *Physiological Reviews*, 91, 10 2011. doi:10.1152/PHYSREV.00006.2011/SUPPL_FILE/DESCRIPTIONS2.DOCX.
- [21] Israel Cohen and Sharon Gannot. *Springer Handbook of Speech Processing*. 01 2008. doi:10.1007/978-3-540-49127-9_44.
- [22] Philipos C. Loizou. Speech production and perception. *Speech Enhancement*, page 65, 10 2020. doi:10.1201/B14529-4/SPEECH-PRODUCTION-PERCEPTION-PHILIPPOS-LOIZOU.
- [23] John J. "Ohala. "Respiratory Activity in Speech", pages "23–53". "Springer Netherlands", "Dordrecht", "1990". doi:"10.1007/978-94-009-2037-8_2".

- [24] How does the larynx work? 11 2018. URL: <https://www.ncbi.nlm.nih.gov/books/NBK534630/>.
- [25] D. B. Fry. *The Physics of Speech*. Cambridge Textbooks in Linguistics. Cambridge University Press, 1979. doi:10.1017/CBO9781139165747.
- [26] Hiroya Fujisaki. Dynamic characteristics of voice fundamental frequency in speech and singing. *The Production of Speech*, pages 39–55, 1983. URL: https://link.springer.com/chapter/10.1007/978-1-4613-8202-7_3, doi:10.1007/978-1-4613-8202-7_3.
- [27] A. M. De Lima Araujo and F. Violaro. Formant frequency estimation using a mel-scale lpc algorithm. *ITS 1998 Proceedings - SBT/IEEE International Telecommunications Symposium*, 1:207–212, 1998. doi:10.1109/ITS.1998.713118.
- [28] Nichole Houle and Susannah V. Levi. Acoustic differences between voiced and whispered speech in gender diverse speakers). *The Journal of the Acoustical Society of America*, 148:4002, 12 2020. URL: <https://asa.scitation.org/doi/abs/10.1121/10.0002952>, doi:10.1121/10.0002952.
- [29] Hideaki Konno, Mineichi Kudo, Hideyuki Imai, and Masanori Sugimoto. Whisper to normal speech conversion using pitch estimated from spectrum. *Speech Communication*, 83:10–20, 10 2016. doi:10.1016/J.SPECOM.2016.07.001.
- [30] D. P. Rowe and S. J. O’Leary. Peripheral auditory system. *Encyclopedia of the Neurological Sciences*, pages 329–334, 1 2014. doi:10.1016/B978-0-12-385157-4.00121-4.
- [31] Jan Schnupp, Israel Nelken, and Andrew King. Auditory neuroscience : Making sense of sound. *Book*, page 347, 2011.
- [32] Paul Hill. *Audio and Speech Processing with MATLAB*. CRC Press, 2018. doi:10.2/JQUERY.MIN.JS.
- [33] Harvey Fletcher. Loudness, pitch and the timbre of musical tones and their relation to the intensity, the frequency and the overtone structure. *The Journal of the Acoustical Society of America*, 6:59, 6 2005. URL: <https://asa.scitation.org/doi/abs/10.1121/1.1915704>, doi:10.1121/1.1915704.
- [34] Brian R Glasberg and Brian C.J Moore. Derivation of auditory filter shapes from notched-noise data. *Hearing Research*, 47(1):103–138, 1990. URL: <https://www.sciencedirect.com/science/article/pii/037859559090170T>, doi:https://doi.org/10.1016/0378-5955(90)90170-T.
- [35] Gunnar Fant. *Acoustic Theory of Speech Production: With Calculations based on X-Ray Studies of Russian Articulations*. De Gruyter Mouton, 2012. URL: <https://doi.org/10.1515/9783110873429>, doi:doi:10.1515/9783110873429.
- [36] Jeremy Bradbury. Linear predictive coding. *Mc G. Hill*, 2000.
- [37] L. Hodgson, M.E. Jernigan, and B.L. Wills. Nonlinear multiplicative cepstral analysis for pitch extraction in speech. pages 257–260 vol.1, 1990. doi:10.1109/ICASSP.1990.115616.

- [38] John-Paul Hosom. Speech recognition. *Encyclopedia of Information Systems*, pages 155–169, 2003. URL: <https://linkinghub.elsevier.com/retrieve/pii/B0122272404001647>, doi:10.1016/B0-12-227240-4/00164-7.
- [39] Toshiaki Fukada, Keiichi Tokuda, Takao Kobayashi, and Satoshi Imai. An adaptive algorithm for mel-cepstral analysis of speech. *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, 1:137–140, 1992. URL: https://www.researchgate.net/publication/3532054_An_adaptive_algorithm_for_mel-cepstral_analysis_of_speech, doi:10.1109/ICASSP.1992.225953.
- [40] F. Rosenblatt. The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, 65:386–408, 11 1958. doi:10.1037/H0042519.
- [41] Kurt Hornik, Maxwell Stinchcombe, and Halbert White. Multilayer feedforward networks are universal approximators. *Neural Networks*, 2(5):359–366, 1989. URL: <https://www.sciencedirect.com/science/article/pii/0893608089900208>, doi:[https://doi.org/10.1016/0893-6080\(89\)90020-8](https://doi.org/10.1016/0893-6080(89)90020-8).
- [42] Qiao Liu, Jiaze Xu, Rui Jiang, and Wing Hung Wong. Density estimation using deep generative neural networks. *Proceedings of the National Academy of Sciences*, 118(15), 2021. URL: <https://www.pnas.org/content/118/15/e2101344118>, arXiv: <https://www.pnas.org/content/118/15/e2101344118.full.pdf>, doi:10.1073/pnas.2101344118.
- [43] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *Communications of the ACM*, 63:139–144, 6 2014. doi:10.1145/3422622.
- [44] Lecture 13 | generative models - youtube. URL: https://www.youtube.com/watch?v=5WoItGTWV54&ab_channel=StanfordUniversitySchoolofEngineering.
- [45] Vladimir Bok, Jakub Langr, and an O'Reilly Media Company. Safari. *GANs in Action: Deep Learning with Generative Adversarial Networks*. 10 2019.
- [46] Ling-Hui Chen, Zhen-Hua Ling, Li-Juan Liu, and Li-Rong Dai. Voice conversion using deep neural networks with layer-wise generative training. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 22(12):1859–1872, 2014. doi:10.1109/TASLP.2014.2353991.
- [47] Berrak Sisman, Junichi Yamagishi, Simon King, and Haizhou Li. An overview of voice conversion and its challenges: From statistical modeling to deep learning, 08 2020. doi:10.1109/TASLP.2020.3038524.
- [48] Masaya Tanaka, Takashi Nose, Aoi Kanagaki, Ryohei Shimizu, and Akira Ito. Scyclone: High-quality and parallel-data-free voice conversion using spectrogram and cycle-consistent adversarial networks. 5 2020. URL: <http://arxiv.org/abs/2005.03334>.
- [49] Rafael Ferro, Nicolas Obin, and Axel Roebel. Cyclegan voice conversion of spectral envelopes using adversarial weights. 10 2019. URL: <http://arxiv.org/abs/1910.12614>.

- [50] Hailun Lian, Yuting Hu, Weiwei Yu, Jian Zhou, and Wenming Zheng. Whisper to normal speech conversion using sequence-to-sequence mapping model with auditory attention. *IEEE Access*, PP:1–1, 09 2019. doi:10.1109/ACCESS.2019.2940700.
- [51] Hailun Lian, Yuting Hu, Jian Zhou, Huabin Wang, and Liang Tao. Whisper to normal speech based on deep neural networks with mcc and f0 features. In *2018 IEEE 23rd International Conference on Digital Signal Processing (DSP)*, pages 1–5, 2018. doi:10.1109/ICDSP.2018.8631888.
- [52] Harshit Malaviya, Jui Shah, Maitreya Patel, Jalansh Munshi, and Hemant Patil. Mspec-net : Multi-domain speech conversion network. pages 7764–7768, 05 2020. doi:10.1109/ICASSP40776.2020.9052966.
- [53] Aníbal Ferreira. On the possibility of speaker discrimination using a glottal pulse phase-related feature. In *2014 IEEE International Symposium on Signal Processing and Information Technology (ISSPIT)*, pages 000183–000188, 2014. doi:10.1109/ISSPIT.2014.7300585.
- [54] Textgrid file formats. URL: https://www.fon.hum.uva.nl/praat/manual/TextGrid_file_formats.html.
- [55] A.J.S. Ferreira. An odd-dft based approach to time-scale expansion of audio signals. *IEEE Transactions on Speech and Audio Processing*, 7(4):441–453, 1999. doi:10.1109/89.771312.
- [56] Jose Ventura, Ricardo Sousa, and Aníbal Ferreira. Accurate analysis and visual feedback of vibrato in singing. 05 2012. doi:10.1109/ISCCSP.2012.6217808.
- [57] Bhaskar Ghosh, Indira Dutta, Albert Carlson, Michael Totaro, and Magdy Bayoumi. An empirical analysis of generative adversarial network training times with varying batch sizes. 10 2020. doi:10.1109/UEMCON51285.2020.9298092.
- [58] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks, 2017. URL: <https://arxiv.org/abs/1703.10593>, doi:10.48550/ARXIV.1703.10593.
- [59] Taeksoo Kim, Moonsu Cha, Hyunsoo Kim, Jung Kwon Lee, and Jiwon Kim. Learning to discover cross-domain relations with generative adversarial networks, 2017. URL: <https://arxiv.org/abs/1703.05192>, doi:10.48550/ARXIV.1703.05192.
- [60] Pytorch documentation — pytorch 1.11.0 documentation. URL: <https://pytorch.org/docs/stable/index.html>.
- [61] Xavier Glorot and Y. Bengio. Understanding the difficulty of training deep feedforward neural networks. *Journal of Machine Learning Research - Proceedings Track*, 9:249–256, 01 2010.
- [62] Zhuohuang Zhang, Chengyun Deng, Yi Shen, Donald S. Williamson, Yongtao Sha, Yi Zhang, Hui Song, and Xiangang Li. On loss functions and recurrency training for gan-based speech enhancement systems, 2020. URL: <https://arxiv.org/abs/2007.14974>, doi:10.48550/ARXIV.2007.14974.
- [63] Pytorch optimizers — pytorch 1.11.0 documentation. URL: <https://pytorch.org/docs/stable/optim.html>.

- [64] Ashia C. Wilson, Rebecca Roelofs, Mitchell Stern, Nathan Srebro, and Benjamin Recht. The marginal value of adaptive gradient methods in machine learning, 2017. URL: <https://arxiv.org/abs/1705.08292>, doi:10.48550/ARXIV.1705.08292.
- [65] Siddhaling Urolagin, Prema K.V., and N. V. Subba Reddy. Generalization capability of artificial neural network incorporated with pruning method. In P. Santhi Thilagam, Alwyn Roshan Pais, K. Chandrasekaran, and N. Balakrishnan, editors, *Advanced Computing, Networking and Security*, pages 171–178, Berlin, Heidelberg, 2012. Springer Berlin Heidelberg.
- [66] Tomoki Toda, Alan W. Black, and Keiichi Tokuda. Voice conversion based on maximum-likelihood estimation of spectral parameter trajectory. *IEEE Transactions on Audio, Speech, and Language Processing*, 15(8):2222–2235, 2007. doi:10.1109/TASL.2007.907344.
- [67] Maitreya Patel, Mihir Parmar, Savan Doshi, Nirmesh Shah, and Hemant Patil. Novel inception-gan for whispered-to-normal speech conversion. pages 87–92. International Speech Communication Association, 9 2019. doi:10.21437/ssw.2019-16.
- [68] Preetum Nakkiran, Gal Kaplun, Yamini Bansal, Tristan Yang, Boaz Barak, and Ilya Sutskever. Deep double descent: Where bigger models and more data hurt, 2019. URL: <https://arxiv.org/abs/1912.02292>, doi:10.48550/ARXIV.1912.02292.
- [69] Boon Pang Lim. *COMPUTATIONAL DIFFERENCES BETWEEN WHISPERED AND NON-WHISPERED SPEECH*. PhD thesis, 01 2011.
- [70] Nelson Roy and Diane M. Bless. Personality traits and psychological factors in voice pathology. *Journal of Speech, Language, and Hearing Research*, 43:737–748, 6 2000. URL: <http://pubs.asha.org/doi/10.1044/jslhr.4303.737>, doi:10.1044/jslhr.4303.737.
- [71] Chris Code and Manfred Herrmann. The relevance of emotional and psychosocial factors in aphasia to rehabilitation. *Neuropsychological Rehabilitation*, 13(1-2):109–132, 2003. PMID: 21854330. doi:10.1080/09602010244000291.