Università degli studi di Torino

SCUOLA DI SCIENZE DELLA NATURA

DIPARTIMENTO DI INFORMATICA



PhD Dissertation

Towards Natural-Sounding Artificial Polyglot Voices: from Voice Conversion to Text-to-Speech via Generative Speech Modeling

Supervisor

Prof. Luigi Di Caro

Candidate

Giuseppe Ruggiero

PhD in collaboration with

Cerence AI

Academic Year 2024/2025

38th Cycle

Abstract

The demand for high-quality synthetic voices that can speak multiple languages natively and naturally is rapidly growing, especially in globalized applications such as virtual assistants, navigation systems, and human-computer interfaces. This PhD, conducted within the industrial research framework between the Department of Computer Science at the University of Turin and Cerence AI, focuses on the development of artificial polyglot voices that maintain native-level quality across multiple languages.

The research began with the implementation of a voice conversion (VC) framework that leverages two widely adopted audio representations: self-supervised learning (SSL) features for capturing linguistic and semantic content, and mel-spectrograms for modeling the acoustic characteristics of speech. While this approach achieved strong results in intra-lingual settings, two major limitations emerged in cross-lingual scenarios: (1) the converted speech was often colored by the target speaker's native accent; and (2) the SSL features retained speaker-specific information, making it more difficult for the system to convert the source speaker identity with the target's, particularly in challenging cases such as male-to-female and female-to-male conversions. To overcome these challenges, two key contributions were proposed. First, a cross-lingual fine-tuning strategy was introduced to mitigate the native-language bias of the target speaker, significantly improving cross-lingual VC performance. Second, a method to generate speaker-independent SSL features was developed, allowing the system to better separate linguistic content from speaker identity and further improve conversion quality. Together, these

innovations enabled the VC system to achieve state-of-the-art results in both intra-lingual and cross-lingual VC benchmarks.

Building on the success of the VC system, the architecture was integrated into a text-to-speech (TTS) system by learning to generate the same intermediate SSL (or speaker-independent SSL) representations directly from text. This adaptation enabled the creation of high-quality polyglot TTS voices that inherited the strengths of the original VC framework. Preliminary tests indicate promising results in polyglotism and naturalness, opening interesting directions for lifelike and natural-sounding TTS models.

Contents

1 Introduction **1**

2 Background and State of the Art **7**

2.1 Self-Supervised Speech Representation Learning 7

2.1.1 Historical Context of Representation Learning 9

2.1.2 Modern Speech Representation Learning Approaches 10

2.1.3 Speaker and Content Disentanglement 16

2.2 Voice Conversion 21

2.2.1 Historical Context of Voice Conversion 23

2.2.2 Modern Voice Conversion Approaches 25

2.3 Text-to-Speech 27

2.3.1 Historical Context of Text-to-Speech 29

2.3.2 Modern Text-to-Speech Approaches 31

2.4 Evaluation Metrics 35

2.4.1 Objective Metrics 35

2.4.2 Subjective Metrics 37

3 High-Quality Cross-Lingual Voice Conversion System **39**

3.1 Introduction 39

3.2 Model Architecture 41

3.2.1 Problem Definition 41

3.2.2 Cross-Lingual Fine-Tuning 44

3.2.3 Content Encoder . . . 46
3.2.4 Acoustic Model . . . 47
3.2.5 Vocoder . . . 48
3.3 Experiments . . . 48
3.3.1 Experimental Setup . . . 48
3.3.2 Evaluation Metrics . . . 49
3.3.3 Results . . . 51
3.4 Conclusion and Limitations . . . 54

4 Eta-WavLM: Efficient Speaker Identity Removal in Self-Supervised Speech Representations 56

4.1 Introduction . . . 56
4.2 Method . . . 58
4.2.1 Problem Definition . . . 58
4.2.2 Computation of Latent Basis and Bias . . . 59
4.2.3 Creation of Eta-WavLM Representations . . . 61
4.3 Experiments . . . 62
4.3.1 Speaker-Related Classification Task . . . 63
4.3.2 Voice Conversion Task . . . 65
4.3.3 Ablation: Speaker Encoder and PCA . . . 69
4.4 Conclusion and Limitations . . . 71

5 High-Quality Polyglot Text-to-Speech System 73

5.1 Introduction . . . 73
5.2 Model Architecture . . . 74
5.2.1 Problem Definition . . . 76
5.2.2 TTS Content Encoder . . . 78
5.3 Experiments . . . 79
5.3.1 Experimental Setup . . . 79
5.3.2 Evaluation Metrics . . . 80

5.3.3 Results 81
5.4 Conclusion and Limitations 82

6 Conclusion **84**

A Mel-spectrogram **86**

B Publications and Patents **89**

List of Figures **91**

List of Tables **93**

References **119**

Chapter 1

Introduction

In recent years, the creation of artificial voices that closely resemble human ones has seen remarkable advancements, driven by the rapid evolution of deep learning methods in the generative speech domain. Modern neural architectures and training strategies have significantly improved the quality, intelligibility, and naturalness of synthesized speech, making high-quality artificial voices increasingly indistinguishable from human speech. These advances have broadened the deployment of synthetic voices across a wide range of real-world applications, including virtual assistants, navigation systems, and human-computer interaction, where the presence of lifelike and expressive speech can enhance the user experience.

Building upon these technological advancements, this thesis focuses on the development of high-quality generative speech systems to address concrete needs within the industry. The research was conducted within the framework of an industrial collaboration between the Department of Computer Science at the University of Turin and Cerence AI, a leader in developing in-car virtual assistants for the automotive domain, focusing on their text-to-speech (TTS) division in Italy and Belgium. This division is responsible for generating high-quality artificial voices for various customers worldwide, ensuring that their in-car systems can communicate effectively and naturally with drivers across different languages and scenarios. The research project started to address specific needs identified within the company, including:

- **Reducing costs** associated with the expansion of internal voice databases, which traditionally rely on extensive studio recordings with professional voice talents;
- Aligning with advancements in the literature and maintaining competitiveness by developing **a high-quality polyglot TTS system**, capable of producing natural-sounding speech across multiple languages with native-level quality;
- Exploring and implementing **modern and highly expressive TTS models** to enhance the emotional and prosodic variability of generated voices, creating more engaging and lifelike user experiences.

Developing such advanced systems poses significant challenges, primarily due to the extensive data requirements associated with training high-quality neural TTS models. This process typically requires large quantities of high-fidelity, studio-recorded speech data, which involves substantial financial investments in recording sessions with professional voice talents. The complexity of this requirement increases further when the goal is to produce polyglot voices, as it is particularly challenging to find individuals who can speak multiple languages fluently and with native-level pronunciation and accent consistency across languages. While publicly available datasets in the speech research community provide a valuable resource for model training and experimentation, they often fall short of meeting the requirements for high-quality TTS systems. Even large-scale multilingual datasets frequently exhibit a notable disparity in quality, particularly between English data and other languages, and often lack the level of consistency necessary for training production-level TTS systems. Furthermore, the challenge of sourcing authentic polyglot data is even more pronounced within public datasets. To address these challenges and align with the identified needs, this research adopted a progressive approach, transitioning from the creation of a system able to generate artificial polyglot data to extend in-house voice databases at reduced costs, to the implementation of a modern and polyglot TTS system.

In the first phase of the research, voice conversion (VC) was identified as the task best suited to extend internal voice databases efficiently and cost-effectively. VC systems are

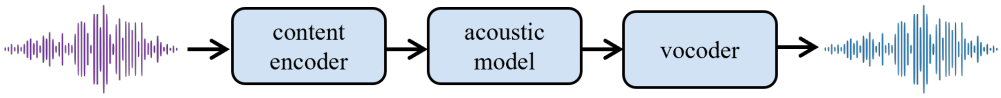


Figure 1.1: High-level overview of the voice conversion system.

designed to convert the speaker identity of a source speaker to that of a target speaker while preserving the linguistic content, emotions, accent, and prosody of the source speech. Leveraging this capability, the idea was to artificially augment existing in-house databases by converting recordings of internal speakers into those of other speakers, enabling data expansion not only within the original language but also across additional languages without requiring expensive new recording sessions.

Inspired by [1], this research proposed an any-to-one VC (A2O VC) system capable of high-fidelity speaker identity conversion towards a single target speaker, effectively extending internal databases while simultaneously generating polyglot data. The VC system consists of three key components (illustrated in Figure 1.1):

- **A content encoder**, which extracts semantic speech representations called self-supervised learning (SSL) features from the input audio of any speaker;
- **An acoustic model**, which translates these semantic SSL vectors into acoustic representations called mel-spectrograms[1] of the target speaker;
- **A vocoder**, which converts the resulting mel spectrograms into samples in the time domain (waveforms).

This initial prototype achieved excellent results in intra-lingual VC, particularly in conversions within the same gender (e.g., female-to-female or male-to-male) and within the same language. However, performance degraded in cross-lingual and cross-gender scenarios, motivating further improvements.

To enhance the system's cross-lingual capabilities, a simple yet effective approach called "Cross-Lingual Fine-Tuning" was developed. The core idea of this strategy is to leverage the phonetic and prosodic knowledge of a model trained on a specific language

[1]Mel-spectrogram: see Appendix A.

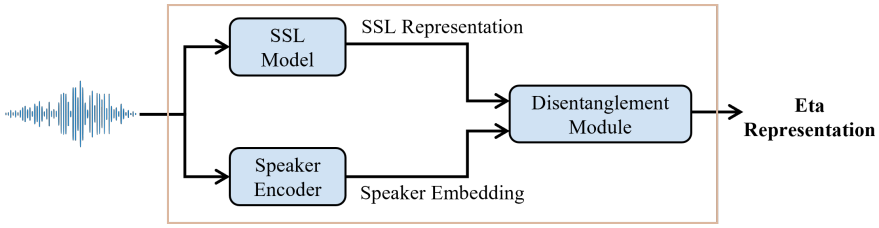


Figure 1.2: Eta content encoder.

(e.g., French) to enhance the pronunciation and linguistic content of a specific target speaker (e.g., an English speaker) when speaking that language. To implement this, the acoustic model of the VC system is first pre-trained on data in the target language to improve (e.g., French), and then fine-tuned with the actual target speaker data (e.g., English). This approach significantly improved cross-lingual performance, enabling the target speaker's synthetic voice to exhibit native-like pronunciation in new languages, achieving state-of-the-art (SOTA) results in cross-lingual VC tasks.

After improving the system's cross-lingual capabilities, the focus shifted to enhancing cross-gender conversions. To address this, the research introduced a novel speaker identity removal strategy within SSL representations through the design of an offline speaker-content disentanglement module, resulting in the creation of speaker-independent SSL representations. Conceptually, the approach is straightforward: two pre-trained models are used to extract an SSL representation and a speaker embedding[1], respectively, from the same input audio. The disentanglement module then derives a *speaker-independent eta representation* from the SSL representation, conditioned on the speaker embedding. Unlike prior approaches that have attempted to remove speaker information from SSL representations using quantization approaches [2; 3] or by employing online training strategies with specific loss functions [4; 5], the proposed method requires no training, fine-tuning, quantization, or significant computational resources, making it lightweight and efficient while preserving the quality of the representations.

At this stage, the content encoder of the VC system (Figure 1.1) was replaced with the *eta content encoder* shown in Figure 1.2, which now outputs speaker-independent eta

[1]Speaker embedding: vector describing speaker characteristics.

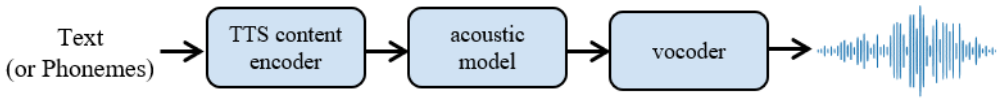


Figure 1.3: High-level overview of the text-to-speech system.

representations instead of raw SSL features. This adjustment fundamentally simplifies the conversion task for the acoustic model, as it no longer needs to learn to remove the source speaker's identity while simultaneously injecting the target speaker's identity. Instead, the acoustic model can focus entirely on applying the target speaker's voice characteristics during the conversion process, with less risk of compromising the linguistic content, accent, or prosody of the source speech. Consequently, this simplification not only improves the accuracy of speaker conversion but also enhances overall performance, enabling the system to achieve SOTA results in cross-gender scenarios as well.

Building upon the high-quality results obtained with the VC system, the next step of this research was to extend the developed pipeline towards TTS synthesis. Taking inspiration from recent TTS works that predict semantic features from text within the synthesis pipeline [6; 7], a simple yet effective modification of the VC system was proposed. In the VC framework, the content encoder takes audio as input and outputs semantic SSL (or eta) representations. To enable TTS synthesis within the same framework, this module was redesigned to take text (or phonemes) as input while continuing to output the same type of semantic representations. This adjustment allowed the acoustic model and vocoder to remain unchanged, reusing the robust components of the original VC system while adapting it for TTS.

Thus, the key objective became the creation of an additional module capable of converting text into semantic representations compatible with the pipeline. In this new framework, the original VC content encoder served as the generator of the ground-truth semantic representations, which were used as training targets for the new module, referred to as the *TTS content encoder*. With this modification, illustrated in Figure 1.3, it became possible to develop a modern TTS system that retained the most valuable characteristics of the VC system, namely the ability to generate high-quality, polyglot

voices. Furthermore, this system was inherently versatile, capable of performing both VC (when the input to the content encoder is audio) and TTS (when the input is text), depending on the input modality. While this system remains in the early stages of research, preliminary tests have demonstrated promising results, indicating the potential to advance the capabilities of the in-house TTS system.

The remainder of this thesis is organized as follows. Chapter 2 provides background and related work in VC and TTS, positioning the contributions of this research within the broader context of generative speech modeling. Chapter 3 examines the developed VC system and the proposed cross-lingual fine-tuning strategy, presenting the experimental setup and validation demonstrating the system's effectiveness. Chapter 4 describes the speaker identity removal strategy within SSL representations and its integration into the VC system, detailing experiments and validations that assess its impact on cross-gender scenarios. Chapter 5 reports on the extension of the VC system to TTS, outlining the proposed modifications, the resulting system architecture, and initial results obtained with the proposed approach. Finally, Chapter 6 summarizes the contributions of this thesis, offering a comprehensive overview of the results achieved.

Chapter 2

Background and State of the Art

Summary

This chapter provides the background and state-of-the-art for the main topics explored in this research. The work primarily focuses on voice conversion and text-to-speech synthesis, two tasks that have seen significant progress with the adoption of deep learning and, more recently, self-supervised techniques. Since a core component of this research is the use of self-supervised speech representations, the chapter begins with an in-depth overview of self-supervised learning. This section introduces the key ideas, historical developments, and major models that have shaped the field. Following this, the chapter explores the evolution of voice conversion and text-to-speech, covering both early traditional approaches and the most recent advancements. Finally, the chapter concludes with an overview of the objective and subjective metrics typically employed to assess the performance of voice conversion and text-to-speech systems.

2.1 Self-Supervised Speech Representation Learning

Over the past decade, deep learning has dramatically advanced the field of speech processing, particularly through supervised methods that rely on large volumes of labeled data [8; 9; 10]. While these approaches have achieved remarkable success in real-world

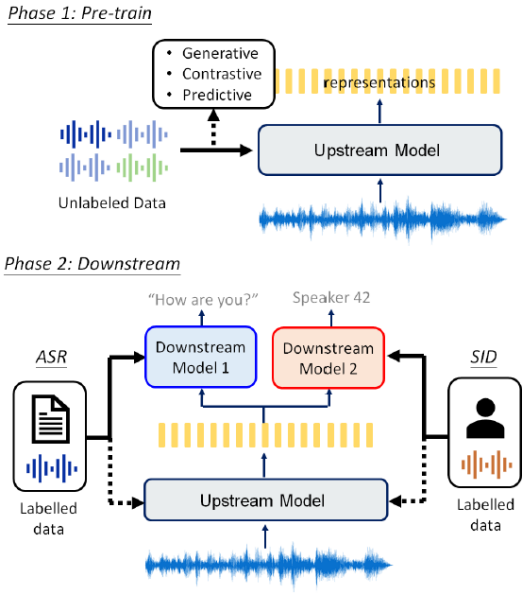


Figure 2.1: Self-supervised representation learning framework. Figure reproduced from [15] © 2022 IEEE.

speech applications, their effectiveness remains limited to well-resourced languages and domains with annotated corpora. To overcome the need for labeled data, researchers have increasingly explored the use of unpaired audio-only data, enabling progress in low-resource languages and industrial applications [11; 12; 13]. In this context, self-supervised learning (SSL) has emerged as a powerful paradigm due to its ability to learn rich and transferable speech representations without the need for manual annotation.

Self-supervised learning (SSL) is a subfield of unsupervised learning [14], comprising techniques that use information extracted from the input data itself as the label to learn representations useful for downstream applications [15]. These representations are typically learned by general-purpose models during a pre-training phase, which can then be fine-tuned or directly applied to a variety of downstream tasks with minimal labeled data. Figure 2.1 shows the SSL framework in the context of two downstream applications: automatic speech recognition (ASR) and speaker identification (SID).

2.1.1 Historical Context of Representation Learning

Before the advent of SSL, speech representation learning was primarily driven by supervised or unsupervised techniques, each with distinct goals and limitations. In supervised settings, representations were learned implicitly during task-specific model training for applications such as ASR, SID, or emotion classification. However, these representations were often closely coupled with the task objectives, which limited their generalizability across different applications. In contrast, early unsupervised approaches aimed to uncover general patterns from unannotated data, with the goal of building versatile models applicable to a variety of tasks. Initial research in this area relied on simple clustering techniques such as k-means [16; 17], where speech segments with similar characteristics were grouped together and used for basic recognition of isolated words [15]. Although such methods represented an important early step, they mainly captured surface-level similarities and were not able to model complex speech dynamics.

With the introduction of probabilistic generative models, most notably Gaussian mixture models (GMMs) and hidden Markov models (HMMs), speech representation learning advanced significantly. GMMs [18] were used to represent short-term acoustic variability by expressing each frame as a weighted combination of Gaussian components, while HMMs [19] incorporated temporal dynamics, enabling the modeling of continuous speech. Both GMMs and HMMs were trained by maximizing the likelihood of the observed data given the model parameters, either in a supervised or unsupervised fashion [15]. For decades, GMM- and HMM-based architectures dominated speech technology, until discriminative training methods were found to produce better results compared to models trained using maximum likelihood [20].

Since then, representation learning has shifted toward neural models, which introduced distributed representations capable of encoding richer acoustic and temporal information. Examples of such approaches include restricted boltzmann machines (RBMs) [21], deep neural networks (DNNs) [22], and recurrent neural networks (RNNs) [23; 24]. These models gained considerable traction due to their ability to learn feature representations automatically during training, thereby reducing the dependence on hand-

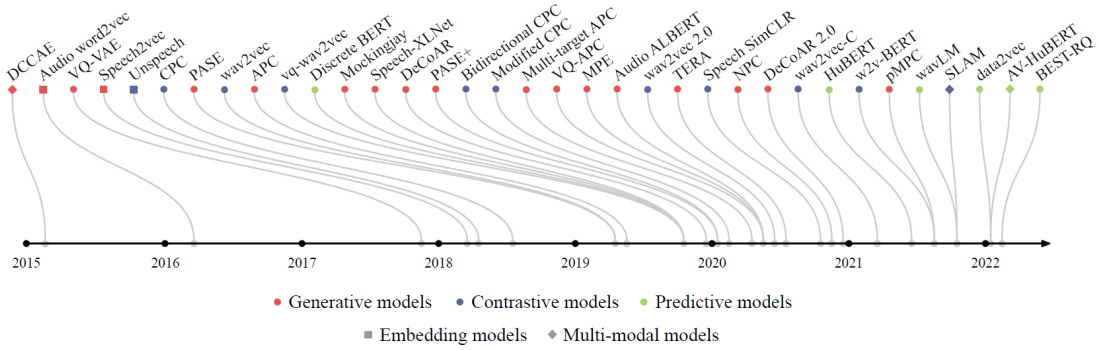


Figure 2.2: Generative, contrastive and predictive SSL models. Figure reproduced from [15] © 2022 IEEE.

engineered features that characterized traditional methods. This property can be further enhanced by stacking multiple layers, allowing the network to construct progressively higher level concept representations [15]. Such an ability has proven particularly important, as it enabled the pre-training of deep networks on auxiliary tasks designed to extract informative and transferable intermediate features, thereby establishing the foundation for modern SSL methods.

2.1.2 Modern Speech Representation Learning Approaches

Modern SSL pre-training tasks designed for speech can be broadly categorized into three main types: *generative*, *contrastive*, and *predictive* approaches [15]. Figure 2.2 presents a timeline highlighting the most influential SSL models within each category.

Generative Approaches

In generative approaches, the pre-training task involves reconstructing the original input from a limited or altered view [15]. Autoencoders (AEs) represented one of the earliest formulations of this concept [25]. By compressing the input into a lower-dimensional latent space and reconstructing it through a decoder, AEs encouraged the extraction of salient patterns rather than low-level details. Subsequent variants, such as denoising AEs (DAEs) [26] and variational AEs (VAEs) [27], introduced stochastic and regularized latent spaces that improved the robustness and quality of the learned representations

[28; 29; 30; 31]. Building on these ideas, the vector-quantized VAE (VQ-VAE) [32] incorporated discrete latent units by mapping the encoder output to a finite codebook, yielding representations that correlate closely with phoneme-like units and proved effective for tasks such as VC [33].

While traditional AE-based models learn to reconstruct an input from its compressed latent representation, more recent generative approaches pursue the same goal through masked reconstruction. In this paradigm, the network is challenged to restore deliberately hidden regions of the input, forcing it to capture deeper contextual and semantic dependencies within the speech signal. Although specific masking policies vary across models, the core architecture typically consists of an RNN [34] or, more recently, a Transformer encoder [35; 36] as the backbone, followed by a multilayer perceptron (MLP) head. The reconstruction loss is usually computed only over the masked timesteps by comparing the model predictions with the original input. Additionally, to make the pre-training task more meaningful, masking is applied over contiguous spans that roughly correspond to higher-level linguistic units, making the process more analogous to token masking in text. For example, pMPC [37] selects masked regions based on phonetic segmentation of utterances, allowing the model to reconstruct linguistically coherent units. However, this approach relies on labeled data to obtain phonetic boundaries.

Other studies have then explored label-free alternatives to direct input masking. Mockingjay [38], for instance, demonstrated that Transformer-based encoders can learn robust contextual representations by reconstructing randomly masked regions of spectrograms, while the TERA model [39] extended this idea by combining time, frequency, and channel masking, leading to improved generalization across downstream tasks. Other methods shifted their focus from masking the input itself to masking portions of the model's hidden states. A representative example is Non-Autoregressive Predictive Coding (NPC) [40], which introduced time masking through masked convolutional blocks rather than directly on the input features [39].

Contrastive Approaches

Speech contains many entangled features, making raw signal reconstruction an imperfect approach for discovering contextualized latent factors of variation [15]. Contrastive learning methods address this limitation by training models to distinguish between related and unrelated samples in a latent space, encouraging the extraction of representations that are both informative and invariant to low-level variability [41]. The key idea is to bring closer the representations of segments that share information (for example, two portions of the same utterance) while pushing apart unrelated segments, effectively shaping a structured embedding space that reflects meaningful dependencies in the data.

A major breakthrough in this direction was contrastive predictive coding (CPC) [42], which proposed learning speech representations by predicting future latent features from past context rather than reconstructing the input itself. The model encodes high-dimensional audio into compact latent vectors using a convolutional encoder and summarizes these into a context vector through a recurrent network that captures long-range dependencies. By contrasting true future latent vectors with negative samples drawn from other parts of the sequence, CPC encourages the context representation to be predictive and informative about the future, yielding rich and context-aware representations that transfer effectively to downstream applications such as speech recognition and speaker identification, even with limited labeled data.

Building on CPC, Wav2Vec [43] simplified the architecture while maintaining the same core learning principle. It employs one-dimensional convolutional encoders for both latent feature extraction and contextual representation learning, allowing faster and fully parallelized computation. After being trained on large amounts of unlabeled audio, Wav2Vec produces representations that can be used as effective input features for downstream speech recognition systems, significantly enhancing accuracy.

The subsequent Wav2Vec 2.0 model [44], illustrated in Figure 2.3, extended this line of research by combining the contrastive prediction framework with the masked reconstruction paradigm described earlier. In this approach, a convolutional encoder first transforms the raw waveform into a sequence of latent features, which are then partially

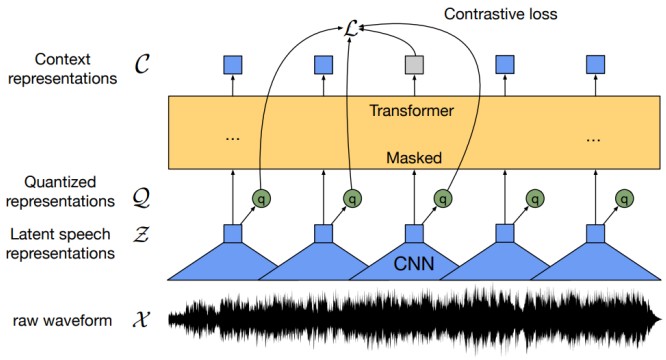


Figure 2.3: The Wav2Vec 2.0 SSL model. Figure reproduced from [44].

masked and processed by a Transformer network. In parallel, a quantization module [45] discretizes the encoder outputs into a set of learned codebook entries, enabling the model to discover compact, phoneme-like units. The training objective encourages the Transformer to predict the correct discrete representations for the masked segments while distinguishing them from randomly sampled alternatives. This design combines the predictive strength of contrastive learning with the contextual abstraction of masking, enabling the model to encode both fine-grained acoustic details and higher-level linguistic structure. Notably, Wav2Vec 2.0 was the first SSL model to achieve a single-digit word error rate (WER) on LibriSpeech [46] using only low-resource subsets from Libri-Light [47] for fine-tuning, setting a new benchmark for data-efficient speech recognition [15]. For this reason, Wav2Vec 2.0 is widely considered as one of the most influential SSL models for speech to date. Subsequently, multilingual extensions including XLSR-53 [48], XLS-R [49], and massively multilingual speech (MMS) [50] further scaled the Wav2Vec 2.0 framework to data covering dozens to thousands of languages, demonstrating the robustness and versatility of contrastive pre-training also in cross-lingual and low-resource settings.

Predictive Approaches

Similar to contrastive approaches, predictive methods also learn their representations using target labels during the pre-training phase. These targets are typically derived

automatically beforehand, often through an external model or a clustering procedure. However, unlike contrastive approaches, predictive methods rely on standard predictive objective functions, such as categorical cross-entropy (CCE), rather than contrastive losses [15]. The development of predictive methods was largely motivated by some inherent limitations of contrastive frameworks, most notably their dependence on the selection of negative samples. Choosing appropriate negatives is both critical and non-trivial: if they are too similar, the task becomes ambiguous; if they are too different, the task becomes too simple. Moreover, contrastive methods require a large number of negatives to perform effectively, which increases both memory usage and computational cost. Predictive approaches naturally avoid these issues, resulting in more stable and resource-efficient training. In addition, they align more closely with the masked prediction paradigm that has proven highly successful in NLP, enabling speech models to capture linguistic dependencies in a manner similar to text-based language models.

Early work in this direction was inspired by the success of BERT-style pre-training in NLP [51]. The main challenge in extending such approaches to speech was that, unlike text, the speech signal is continuous and lacks an explicit vocabulary of discrete symbols such as words or tokens. To bridge this gap, Discrete BERT [52] introduced a two-stage process. First, a pre-trained vq-wav2vec model [53], a contrastive model similar to Wav2Vec but augmented with quantization to learn and output discrete representations, was employed to convert raw audio into sequences of discrete units, effectively building a learned speech vocabulary. Then, these units were used as both input and prediction targets for a standard BERT model trained with a CCE loss. This approach demonstrated that language-modeling techniques could be successfully transferred to the speech domain once an appropriate discrete representation was established.

Building on this idea, hidden unit BERT (HuBERT) [54] proposed a more flexible and data-driven formulation that eliminated the need for a pre-trained model for target generation. Instead of deriving discrete units from a learned encoder, HuBERT applies k-means clustering directly to acoustic features such as mel-frequency cepstral coefficients

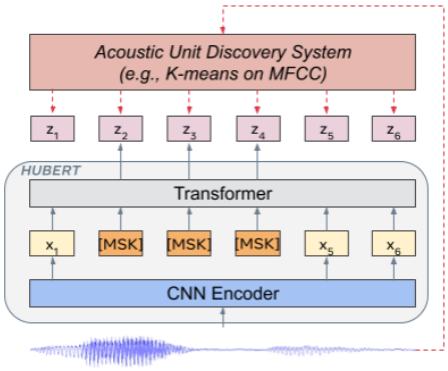


Figure 2.4: The HuBERT SSL model. Figure reproduced from [54] © 2021 IEEE.

(MFCCs)[1] [55]. The resulting cluster assignments serve as pseudo-labels for the model's masked prediction task. As shown in Figure 2.4, the architecture closely follows that of Wav2Vec 2.0, combining a convolutional feature encoder with a Transformer-based context network, but it replaces the contrastive objective with a frame-level classification loss applied only on masked time steps. Through this mechanism, HuBERT learns to predict the cluster label of each masked frame, effectively capturing both acoustic and linguistic structures in speech. This design led to SOTA results on several speech recognition benchmarks, establishing HuBERT as one of the foundational SSL models for speech alongside Wav2Vec 2.0.

An extension of HuBERT is WavLM [56], a model designed to perform well across a wide range of speech tasks, not just speech recognition. By combining masked speech prediction with denoising pre-training, WavLM learns rich, noise-robust, and speaker-aware universal representations that generalize effectively across domains. As shown in Figure 2.5, WavLM is largely identical to HuBERT but introduces two key enhancements. First, it augments the Transformer's self-attention mechanism with a gated relative position bias (GReP) [57], enabling the model to capture temporal relationships between speech frames more flexibly. This is particularly beneficial for phonetic and prosodic modeling. Second, in addition to the same masking strategy used in HuBERT, WavLM employs an utterance mixing strategy, where random segments from other speakers are added to the

[1]MFCC: a type of cepstral representation of an audio signal. MFCCs are designed to approximate human auditory perception by applying the Mel scale to frequency representation.

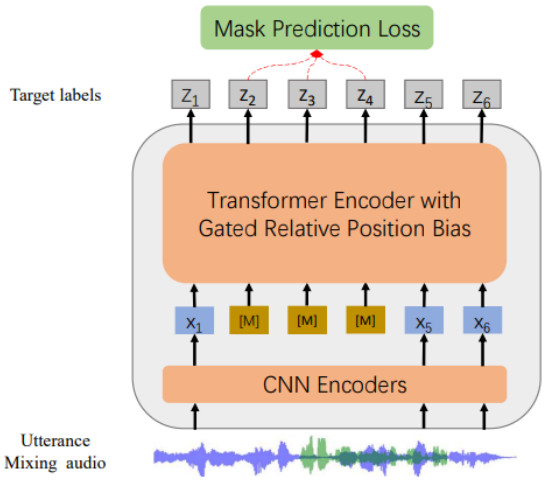


Figure 2.5: The WavLM SSL model. Figure reproduced from [56] © 2022 IEEE.

input during pre-training, while the model must still predict the targets corresponding to the original utterance. This encourages the network to focus on the primary speaker and remain invariant to background or overlapping speech. Thanks to these innovations, WavLM achieved significant improvements in tasks such as speech separation, speaker verification, and speaker diarization, while also outperforming HuBERT and Wav2Vec 2.0 in many other speech-related benchmarks.

2.1.3 Speaker and Content Disentanglement

Despite the success of SSL models like Wav2Vec 2.0, HuBERT, and WavLM in extracting robust and versatile features directly from speech waveforms, their representations encode a wide range of speech attributes, including linguistic content, speaker identity, emotions, and background conditions, making them versatile but often task-agnostic. For example, while tasks such as SID benefit from low-level acoustic features, applications like speech-to-speech translation (S2ST), VC, or TTS require more linguistically and semantically rich representations of the input utterance [58]. As a result, disentangling speaker-related and non-speaker-related information within SSL representations has become a critical challenge for improving task-specific performance [59], though it remains highly challenging [60]. Over time, numerous approaches have been proposed to

tackle this issue, which can broadly be categorized into three main types: *quantization-based*, *perturbation-/normalization-based*, and *learned* disentanglement methods [61].

Quantization-based Approaches

In quantization-based approaches, speaker and content disentanglement is achieved by compressing high-dimensional continuous SSL representations into more compact discrete tokens, a technique commonly referred to as *speech tokenization*. In particular, the goal is to map all the frames of the SSL representations into a single discrete value that should describe as best as possible that frame. The key idea is that quantization imposes an information bottleneck separating content from speaker details [1], thereby creating pseudo-text representations of the speech utterances. These tokens have been shown to capture high-level speech information closely related to phonemes [15].

One of the earliest strategies for quantizing SSL representations due to its simplicity and unsupervised nature is the use of k-means clustering. Similar to the approach adopted by HuBERT and WavLM for quantizing MFCCs, this method involves clustering the SSL representation frames during the training phase of k-means. Then, during inference, each SSL frame is assigned to its nearest cluster center, resulting in a sequence of discrete SSL representations. These discrete tokens can subsequently be used as inputs and/or targets to enhance performance in content-related tasks [3; 62; 63]. However, despite its simplicity, k-means clustering has two major drawbacks. First, it often leads to a loss of linguistic and semantic information when compared to the original continuous speech representations [64; 65]. Second, not all SSL representations are well-suited to clustering. For example, the features extracted from models like Whisper [66] have been shown to present challenges for k-means-based quantization [67].

To overcome these limitations, a more recent method called RepCodec [67] has been introduced. RepCodec is a VQ-VAE-based tokenization approach specifically designed to extract semantic tokens from SSL representations. The model consists of an encoder, a VQ module, and a decoder, and is trained to reconstruct the input representations as accurately as possible. During inference, only the encoder and the VQ modules are used

to convert continuous SSL representations into discrete tokens. Extensive experiments with models like HuBERT and Whisper have shown that RepCodec significantly outperforms traditional k-means in both speech understanding and generation tasks.

Perturbation- and Normalization-based Approaches

Unlike quantization-based approaches, perturbation and normalization techniques do not rely on speech tokenization. As a result, the SSL representations remain in their continuous domain also after applying the strategy. This helps preserve the richness of the original representation and avoids the information loss that can result from aggressive quantization. Additionally, these methods are typically simple, both conceptually and in terms of implementation.

A pioneering work in this category is neural analysis and synthesis (NANSY) [68], which introduced straightforward information perturbation techniques applied directly to the input waveform to disentangle speaker-related characteristics from other speech features. The perturbed version of the waveform is then used as input in place of the original, without requiring modifications to the rest of the processing pipeline. Due to its simplicity and compatibility with existing models, the perturbation strategies introduced by NANSY continue to be adopted in modern speech synthesis systems such as [69] as an effective way to reduce speaker-specific information in SSL representations.

Closely related to perturbation in terms of simplicity and compatibility with existing models is the normalization-based strategy. Unlike perturbation, which is typically applied to the input waveform, normalization techniques are applied to the output of a model in order to adjust or refine the resulting representations for specific downstream tasks. For instance, the output of an SSL model can be normalized to suppress irrelevant factors such as speaker identity, depending on the application. A notable example is the work of [63], which demonstrated that a simple per-utterance standardization of CPC features can effectively remove speaker information. Specifically, for a given utterance (or set of utterances) from a speaker, speaker characteristics are attenuated by subtracting the mean and scaling to unit variance. This is based on the observation that

the per-utterance mean of CPC features encodes significant speaker-specific information. Later, [70] extended this idea to more recent SSL models such as WavLM, showing that utterance-level normalization is similarly effective in suppressing speaker-related information in these representations. Interestingly, both studies also found that normalization can improve the performance of lossy quantization techniques such as k-means, helping preserve more of the linguistic content in the representations.

Although perturbation and normalization techniques are less aggressive than quantization in removing speaker-related components from representations, they are still known to compromise linguistic and semantic content, particularly in multilingual settings [71]. In addition, other speaker-related features (e.g., the accent) are also distorted [60].

Learned Disentanglement Approaches

Similar to quantization-based methods, this category of approaches relies on training models to achieve the desired disentanglement. These techniques can be further divided into two subcategories: (1) approaches that aim to extract content-only features from SSL representations through dedicated neural architectures and training procedures; and (2) approaches that integrate specific model components, training strategies, or loss functions to achieve disentanglement during the training of VC or TTS systems.

While quantization-based methods like k-means and VQ-VAE could also be considered part of the first subcategory of methods, they were discussed separately, as they do not explicitly enforce disentanglement. Moreover, quantization fundamentally alters the domain of SSL representations, potentially leading to information loss. In contrast, the models discussed here aim to preserve the continuous nature of SSL features while explicitly separating content from speaker-related information in a more targeted manner.

One example is presented in SoftVC [1], which introduces *soft speech units* derived from discrete tokens obtained via k-means clustering over the output of a HuBERT model. A neural network is trained to predict a distribution over these discrete tokens, thereby retaining more content information and mitigating the information loss caused by hard quantization. The soft units act as an intermediate representation between con-

tinuous SSL features and discrete tokens: while their dimensionality is reduced compared to the original features, they remain in a continuous domain and do not collapse into a single discrete value per frame. In the paper, the authors trained a VC system using both soft speech units and discrete tokens. The soft units achieved superior performance, especially in terms of intelligibility and naturalness of the converted speech.

Another notable approach is ContentVec [60], an SSL method designed to achieve speaker disentanglement without severe loss of content. ContentVec can be seen as a variant of HuBERT that generates speaker-independent SSL representations. To accomplish this, ContentVec introduces three key disentanglement mechanisms into the HuBERT training framework: 1) Removing speaker information from the training labels; 2) Adding a contrastive loss [72] to explicitly enforce speaker invariance in the learned representations; and 3) Injecting speaker identity embeddings [73] into the masked prediction task, so that the need for the speech representation to encode speaker information is relieved. These enhancements enabled ContentVec to outperform baseline models on a variety of content-related speech processing tasks.

Shifting to the second subcategory of methods, a representative example is S2VC [74], a VC model built on a dual-encoder architecture: a source encoder that extracts content from the input utterance and a target encoder that captures speaker-specific characteristics. Both encoders take SSL representations as input. To promote disentanglement between speaker and content information during VC training, S2VC introduces two key mechanisms. First, it applies self-attention pooling on the target encoder outputs, guiding the source encoder to produce similar high-level representations. Second, it incorporates an attention information bottleneck, combining instance normalization [75] (known to suppress speaker-dependent information [76]) with a dimensionality-reduction bottleneck layer in the cross-attention module. Together, these mechanisms encourage the model to focus the attention mechanism on content-related features, effectively disentangling speaker information in an implicit manner during training. As a result, S2VC achieves strong VC performance, outperforming several baselines in both intelligibility and target speaker similarity.

Another approach that performs online disentanglement during VC training is FreeVC [4]. To remove speaker-related features from the 1024-dimensional WavLM representations, FreeVC introduces a bottleneck extractor that compresses them into d-dimensional vectors, where $d \ll 1024$. This dimensionality reduction acts as an information bottleneck, encouraging the resulting low-dimensional features to discard non-content information such as speaker identity and background noise.

Over time, other studies have explored alternative learning strategies to enhance disentanglement, including adversarial learning [5] and mutual information minimization [77; 78; 79], both of which have shown promise in further improving the separation between speaker and content representations.

2.2 Voice Conversion

Voice Conversion (VC) is the task of modifying a speaker's voice to sound like that of another speaker while preserving the original linguistic content. More specifically, VC methods aim to transform the speaker identity of an utterance from a *source speaker* to a *target speaker*, without altering what is being said. VC falls within the broader field of speech synthesis, which includes tasks such as TTS and the modification of speech attributes like voice identity, emotion, and accent.

As with any machine learning approach, VC systems require data to learn this transformation. There are two main types of training data: *parallel* and *non-parallel* [80]. In parallel data settings, both the source and target speakers are required to utter the same sentences, and their recordings must be temporally aligned. In contrast, non-parallel data allow speakers to say different things, and no alignment is required between their utterances. When perfectly aligned parallel data are available, building a VC system is relatively straightforward. However, collecting such data is often expensive and time-consuming. Even when this data is available, automatic time-alignment is typically necessary, which may occasionally fail, leading to quality degradation in the converted speech. For these reasons, recent research has increasingly focused on VC methods that

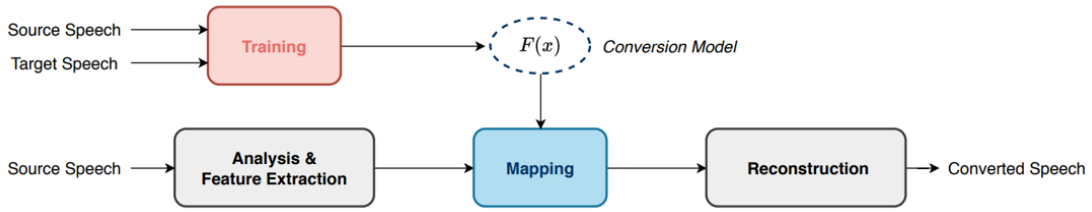


Figure 2.6: The typical flow of a voice conversion system. Figure reproduced from [80] (CC-BY-4.0).

use non-parallel data, which are more practical and reflective of real-world conditions. However, working with non-parallel data presents greater challenges.

The earliest VC systems had several limitations: they typically required parallel data and were only capable of converting a specific, known source voice into a specific, known target voice. These systems are commonly referred to as one-to-one (O2O) parallel VC systems. Due to their restricted applicability in real-world scenarios, research in the field quickly shifted toward more flexible and generalizable approaches. A more advanced VC system not only works with non-parallel data, but can also handle multiple speakers. Unlike O2O, many-to-many (M2M) and any-to-any (A2A) systems can establish conversion mappings between a wide range of speakers at both the source and target levels. The key distinction between M2M and A2A lies in their generalization capability: while M2M systems require all speakers to be seen during training, A2A systems are designed to generalize to unseen speakers, making them the most flexible but also the most challenging VC systems to implement effectively. Consequently, researchers have introduced intermediate configurations to balance flexibility and practicality. Among these, any-to-one (A2O) VC represents a particularly effective compromise. In A2O systems, the model learns to convert speech from any source speaker, including unseen ones, into the voice of a fixed target speaker. This approach offers greater versatility than M2M systems while avoiding the full complexity of A2A architectures by only partially generalizing to unseen speakers. Additionally, by focusing on a single target, A2O models can dedicate more modeling capacity to accurately capturing that speaker's characteristics, which often results in higher-quality and more stable conversions.

Figure 2.6 illustrates a typical VC pipeline, commonly referred to as the *analysis-mapping-reconstruction pipeline* [80]. The goal of this pipeline is to produce converted speech that closely resembles the target speaker's voice, starting from the speech of a source speaker. The pipeline consists of three main modules:

- **Analysis and Feature Extraction**: decomposes a speech signal from the source speaker into an intermediate representation that captures the acoustic properties of speech. These intermediate representations are referred to as *speech features*.
- **Mapping**: transforms the speech features of the source speaker into those of the target speaker. During training, a *conversion model* learns a mapping function using the training data. At inference time, the learned function is applied to convert source features into target features.
- **Reconstruction**: generates an audible speech signal from the features produced by the mapping module. Typically, a vocoder[1] is used in this stage.

A large portion of VC research focuses on the design of the conversion model, as it is the core component of a VC system. Over time, a variety of approaches have been explored, ranging from traditional statistical models to modern deep learning-based techniques.

2.2.1 Historical Context of Voice Conversion

The earliest VC systems relied on parallel training datasets composed of frame-aligned pairs of source and target utterances. In this context, the alignment [81] between corresponding frames played a crucial role, as the goal of the conversion model was to learn how to map the speech features extracted from the source speaker to those of the target speaker on a frame-by-frame basis. If the alignment was inaccurate, the resulting converted speech would often sound distorted or unnatural.

Within this framework, statistical models dominated the field, with GMMs representing one of the most influential approaches [82; 83]. In standard GMM-based VC,

[1]Vocoder: model responsible for converting intermediate representations into raw audio waveforms.

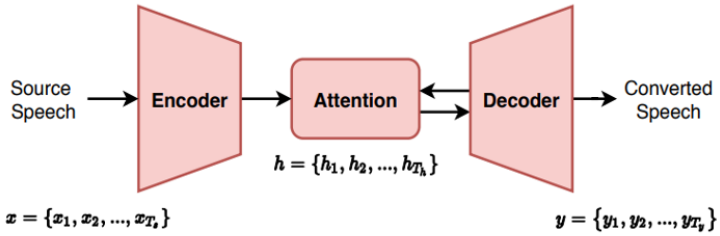


Figure 2.7: Encoder-decoder with attention for voice conversion. Figure reproduced from [80] (CC-BY-4.0).

source and target features were jointly modeled, and conversion was performed by estimating the expected target features conditioned on the observed source features. However, because the mapping operated independently at each frame, it frequently produced over-smoothed spectra and failed to capture the dynamics of natural speech, leading to muffled or less natural outputs [84; 85]. Several extensions [86; 87] attempted to improve temporal coherence and reduce over-smoothing, but these models remained limited by their statistical averaging behavior rather than capturing fine-grained spectral and temporal details [80].

To address the inherent limitations of statistical VC models, the field gradually transitioned to deep learning techniques, which offered greater modeling capacity, fewer restrictions on input feature types and dimensionality, and substantially improved performance. Initial efforts replaced statistical mapping functions with DNNs, demonstrating clear improvements in both spectral accuracy and speaker similarity [88; 89]. Later, recurrent architectures such as long short-term memory (LSTM) networks [23; 90; 91] were adopted to capture long-range temporal dependencies, resulting in more natural and temporally coherent converted speech. Nonetheless, both DNN- and LSTM-based VC still required carefully aligned parallel data, which posed an even greater challenge than in statistical approaches due to the data demands of neural models.

A major breakthrough came with the introduction of attention mechanisms [92] in encoder-decoder architectures (Figure 2.7). These models jointly learned both the alignment and the mapping function during training, eliminating the need for a separate

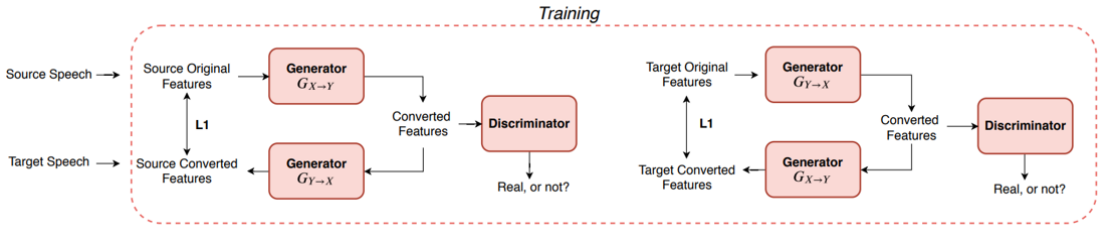


Figure 2.8: The CycleGAN-VC architecture. Figure reproduced from [80] (CC-BY-4.0).

alignment stage. At inference time, the network automatically determined output durations based on the learned alignment patterns. Several attention-based systems were proposed for parallel VC, including Seq2Seq conversion network (SCENT) [93], AttS2S-VC [94] which use RNNs, ConvS2S-VC [95] based on convolutional Seq2Seq settings, and voice transformer network (VTN) [96] which leverages the Transformer [97] architecture. Despite these methods still required parallel data and remained sensitive to data scarcity, attention mechanisms represent a critical step toward relaxing the strict parallel data requirement, marking a pivotal step toward modern non-parallel VC.

2.2.2 Modern Voice Conversion Approaches

Modern VC approaches have successfully overcome the dependency on parallel data through the adoption of generative modeling and representation-learning techniques. Among these, generative adversarial networks (GANs) [98] have played a particularly influential role. Inspired by image-to-image translation frameworks [99; 100], GAN-based VC models learn to transform source speech into the target domain by jointly training a generator and a discriminator in an adversarial setting. A key milestone in this direction was CycleGAN-VC [101] (illustrated in Figure 2.8), which introduced cycle-consistency and identity-mapping losses [102] to preserve the linguistic content of the source speech while modifying speaker characteristics. Remarkably, despite being trained on non-parallel data, CycleGAN-VC achieved performance comparable to a GMM-based system trained on twice as much parallel data [80]. Its success has led to a series of improved variants, including CycleGAN-VC2 [103], CycleGAN-VC3 [104], StarGAN-VC

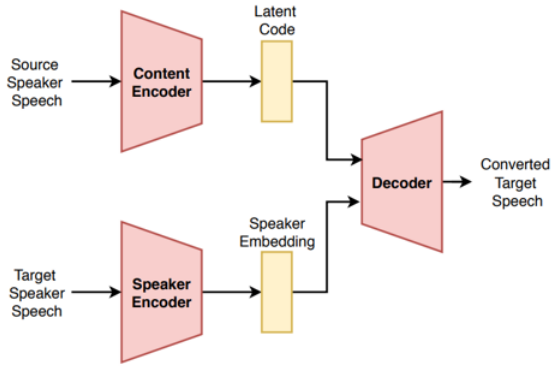


Figure 2.9: A typical auto-encoding network for voice conversion. Figure reproduced from [80] (CC-BY-4.0).

[105], StarGAN-VC2 [106], and MaskCycleGAN-VC [107], each introducing additional mechanisms to enhance conversion quality, speaker generalization, or training stability.

Another influential line of work is based on autoencoding frameworks that explicitly disentangle speaker identity from linguistic content. In these systems, the encoder compresses the input speech into a latent representation that captures speaker-independent content, while the decoder reconstructs speech conditioned on a target speaker embedding [108], as shown in Figure 2.9. By separating these factors, AE-based VC systems can perform flexible VC using either fixed or learned speaker representations such as i-vectors [109] or d-vectors [110]. VAEs [28] further extended this paradigm by introducing stochastic latent spaces, improving robustness and generalization across speakers and acoustic conditions. A well-known limitation of AE-based VC models, however, is their tendency to produce over-smoothed outputs, which often results in poor-quality, buzzy, or unnatural audio [80]. To address this, hybrid approaches such as VAE-GAN [111] integrate adversarial mechanisms into the VAE framework, substantially improving the naturalness and expressiveness of the synthesized speech while retaining the disentanglement benefits of the autoencoding architecture.

Ongoing research in VC with non-parallel data has recently intersected with the rapid progress of SSL models such as Wav2Vec 2.0, HuBERT, and WavLM (Section 2.1.2). This convergence has led to significant advancements in VC, enabling systems

that leverage SSL representations to surpass earlier approaches, even in low-resource settings. Conceptually, these methods share the same objective as AE-based systems: extracting speaker-independent content and recombining it with the target speaker's characteristics to generate converted speech. However, instead of relying only on latent bottlenecks to suppress speaker information, these newer approaches exploit SSL representations, often refined with additional disentanglement strategies such as those described in Section 2.1.3. Representative examples include the previously discussed SoftVC [1] and FreeVC [4], as well as more recent and general-purpose models like Ace-VC [59] and kNN-VC [112].

The introduction of SSL representations into the VC pipeline also brought notable improvements to multilingual scenarios, particularly in cross-lingual VC (XVC), where performance had historically lagged behind standard monolingual VC. The goal of XVC is to transform a source speaker's voice into that of a target speaker while preserving the original linguistic content, even when the source and target speak different languages [113]. A key factor behind this progress is the strong cross-lingual generalization capability of SSL models such as HuBERT and WavLM, which have been shown to capture linguistic structure in a way that transfers well across languages, even when trained only on monolingual datasets [58]. As a result, models like SoftVC, which leverages HuBERT representations, and kNN-VC, which builds on WavLM, have achieved SOTA performance not only in traditional VC settings but also in XVC, becoming leading solutions in A2O XVC and A2A XVC configurations, respectively.

2.3 Text-to-Speech

Text-to-speech (TTS), also known as speech synthesis, is a technology that transforms written text into spoken audio. TTS has a wide range of applications in human-computer interaction and has long been a central research area in artificial intelligence, natural language processing, and speech processing. Building an effective TTS system requires an understanding of both language and human speech production, drawing on multiple

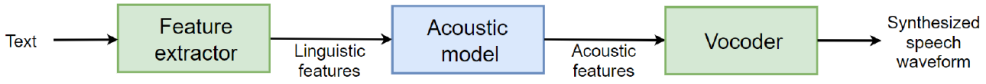


Figure 2.10: The three key components in a text-to-speech system.

disciplines including linguistics, acoustics, signal processing, and machine learning [114]. A modern TTS system typically consists of three components, as shown in Figure 2.10:

- A **text analysis** (or linguistic feature extractor) module, which processes a sequence of text and converts it into linguistic features.
- An **acoustic model**, which maps these linguistic features to acoustic features.
- A **vocoder**, which synthesizes the final speech waveform from the acoustic features.

In earlier TTS systems, the text analysis module played a critical role in extracting detailed information about pronunciation and prosody to facilitate speech synthesis. Common functionalities included text normalization [115], word segmentation [116], part-of-speech (POS) tagging [117], and grapheme-to-phoneme (G2P) conversion [118]. With the advent of neural TTS models, this module has been significantly simplified. Thanks to the high modeling capability of neural networks, characters or phonemes can now be directly used as input for synthesis, although basic pre-processing steps such as text normalization and G2P conversion are still often required.

Modern TTS research has primarily focused on the last two components of the pipeline: the acoustic model and the vocoder. The choice of acoustic features used by the acoustic model largely defines the structure of the TTS pipeline. A variety of features have been explored over time, including MFCCs [55], mel-generalized cepstral coefficients (MGC) [119], band aperiodicity (BAP) [120], fundamental frequency (F0), and the currently most widely adopted representation: mel-spectrograms[1] [114].

Based on the acoustic features generated by the acoustic model, the vocoder produces the audible waveform. While early vocoders generated waveforms directly from linguistic

[1]Mel-spectrogram: see Appendix A.

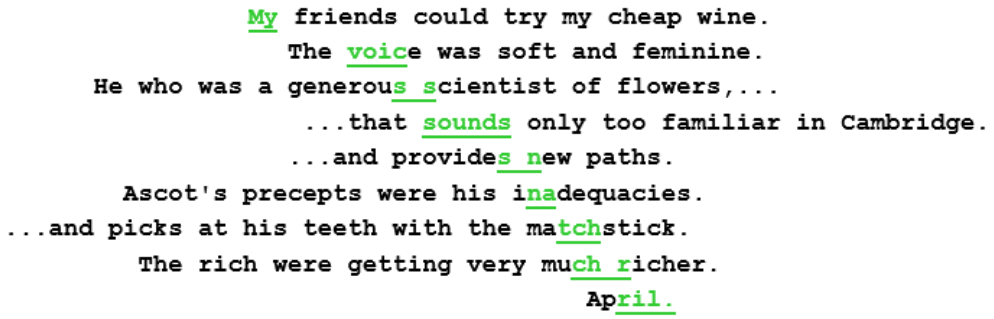


Figure 2.11: Exaple of concatenative approach to synthesize the sentence: *my voice sounds natural*.

features without the need for an acoustic model, modern systems adopt the acoustic model followed by vocoder architecture due to its effectiveness and modular design.

Although end-to-end TTS systems [121; 122] that generate waveforms directly from text have emerged in recent years, the three-stage pipeline in Figure 2.10 remains the most widely adopted and continues to deliver the best overall performance. For this reason, this research work focuses primarily on systems that follow the three-stage pipeline, leaving end-to-end approaches outside the scope of this discussion.

2.3.1 Historical Context of Text-to-Speech

Early TTS systems date back to the 1960s, but one of the most influential and widely adopted techniques, the concatenative synthesis, began gaining traction in the 1980s. This approach is based on the concatenation of pre-recorded speech units, acquired from a single speaker under controlled conditions [123]. These units can range from full words and syllables to smaller segments like half-syllables or phonemes. The recordings are annotated with contextual and phonetic information and stored in a database. Each recorded signal is typically segmented into fundamental units, known as demiphones, and labeled with phonetic, prosodic, and contextual attributes. Figure 2.11 shows an example of the concatenative synthesis approach. This method can produce speech with high intelligibility and a timbre that closely resembles the original speaker. However, it requires a large recording database to cover the most common combinations of

speech units for a target language, and the generated speech often lacks naturalness and emotional expressiveness. This is due to the inherent discontinuities introduced by unit concatenation, which can affect the smoothness of stress and prosody in the final output.

To overcome the limitations of concatenative TTS, statistical parametric speech synthesis (SPSS) emerged as a promising alternative [124]. In SPSS, the relationship between features extracted from input text and the corresponding output acoustic features is modeled using a statistical generative model, referred to as the acoustic model. A complete SPSS system includes a pipeline for extracting linguistic features from the text, an acoustic model that predicts acoustic parameters, and a vocoder that reconstructs the audio waveform from those predicted features [114]. Thus, SPSS methods were the first to adopt the three-stage pipeline illustrated in Figure 2.10. For a long time, the SOTA in SPSS was the HMM-based framework [124]. In this approach, linguistic features extracted from the input text are used to compile decision trees whose leaves cluster the corresponding acoustic features and an HMM is trained for each cluster [125]. These HMMs predict distributions over spectral coefficients, along with their first and second derivatives, as well as a binary flag indicating voiced/unvoiced segments. Using the maximum likelihood parameter generation (MLPG) algorithm [126], spectral coefficients are sampled from this distribution and then passed to the Mel log spectrum approximation (MLSA) vocoder [127] to generate the final waveform. SPSS methods required significantly less recorded data than concatenative approaches, making them more practical in low-resource settings. However, their synthesized speech tended to sound robotic and could still be easily distinguished from human speech [114].

Around the early 2010s, with the rapid progress of neural networks, researchers began incorporating deep learning techniques into SPSS. Zen et al. [128] proposed to replace HMMs with DNNs, demonstrating improved speech quality with a model of comparable complexity to HMM-based systems. Subsequent works further validated these findings [129; 130]. Fan et al. [131] later introduced RNNs into acoustic modeling. Thanks to their ability to model temporal dependencies, RNNs enabled the system to generate more consistent acoustic sequences without relying on dynamic features or the MLPG

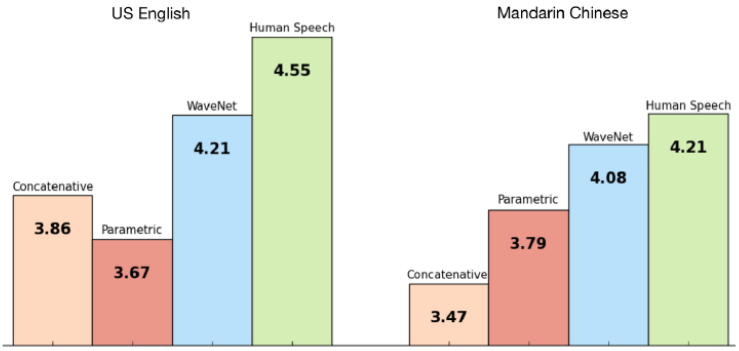


Figure 2.12: Mean opinion scores (MOS) for Concatenative, Parametric, and WaveNet TTS systems in US English and Mandarin Chinese, compared with human speech [133]
.

algorithm. However, these early neural approaches still adhered to the traditional SPSS framework, where acoustic features were predicted from rich linguistic features. A more significant shift came later when Wang et al. [132] proposed generating acoustic features directly from phoneme sequences, effectively simplifying the text analysis module. This work marked one of the first steps toward modern neural TTS systems.

2.3.2 Modern Text-to-Speech Approaches

The introduction of WaveNet [133] marked a major breakthrough in TTS and is widely regarded as the first modern neural TTS model. WaveNet uses dilated convolutions to autoregressively generate waveform samples and, unlike SPSS approaches, it incorporates almost no prior knowledge about audio signals, relying entirely on end-to-end learning. Initially, WaveNet was designed to generate waveforms directly from linguistic features, effectively combining the roles of both the acoustic model and the vocoder into a single unified model. Subsequent studies adapted WaveNet to act only as a vocoder by conditioning it on acoustic features such as linear spectrograms [134] or mel-spectrograms [135; 136; 137]. WaveNet significantly outperformed previous parametric and concatenative TTS methods in terms of perceived naturalness, becoming the first system to achieve a mean opinion score (MOS) approaching that of human speech (Figure 2.12).

Later on, Tacotron [138] was introduced, a Seq2Seq model that generates linear spectrograms directly from characters, further reducing the need for linguistic domain expertise. In this framework, the vocoder used is the simple Griffin-Lim algorithm [139]. Tacotron follows an encoder-decoder architecture where, at each decoding step, the decoder attends to a weighted combination of encoder outputs. The attention mechanism [92] allows the network to focus on the most relevant parts of the input sequence for each output step. Despite achieving promising results in terms of MOS scores, Tacotron's performance was still lower than that of WaveNet and concatenative methods [138].

Shortly after, Tacotron 2 [135] was introduced, retaining the encoder-decoder with attention architecture of the first version but introducing several improvements, particularly in the encoder design. The most significant advancement, however, was the replacement of the Griffin-Lim vocoder with WaveNet, resulting in a substantial gain in audio quality. On the same dataset used in the first publication, Tacotron 2 achieved a MOS of 4.53, closely approaching the human speech score of 4.58 and setting a new benchmark in TTS quality. However, the recurrent structure of Tacotron 2 prevents parallelization, making both training and inference slower and less efficient. In addition, because text and speech sequences can be quite long, RNNs often struggle to capture long-range dependencies, which can degrade the overall synthesis quality [114].

The introduction of the Transformer architecture [97] in the NLP field sparked growing interest in non-recurrent alternatives for speech synthesis. One of the earliest attempts to bring Transformers into TTS was TransformerTTS [140], which employs a Transformer-based encoder-decoder framework to generate mel-spectrograms from phoneme inputs. While TransformerTTS achieves speech quality comparable to Tacotron 2, it offers significantly faster training thanks to its parallelizable architecture.

Despite the strong performance of leading models at the time, their reliance on autoregressive generation presented several challenges. First, the inference speed was slow, especially for long speech sequences, as mel-spectrograms were generated step by step in an autoregressive manner. Second, these models often suffered from robustness issues, including word skipping and repetition, which were primarily caused by inaccuracies in

attention alignments [114]. To overcome these limitations, researchers began exploring non-autoregressive approaches to speech synthesis. One of the most notable contributions in this direction is FastSpeech [141], which employs a feed-forward Transformer architecture to generate mel-spectrograms in parallel. This design greatly improves inference speed and eliminates the reliance on attention mechanisms, thereby avoiding the alignment-related issues of skipping and repeating words. To address the mismatch in length between phoneme sequences and mel-spectrograms, FastSpeech introduces a length regulator. This module uses a duration predictor to estimate the duration of each phoneme and expands the phoneme hidden sequence accordingly. As a result, the expanded phoneme sequence aligns with the length of the target mel-spectrogram, enabling parallel generation while preserving temporal consistency. Building on this foundation, FastSpeech 2 [122] further enhanced the architecture, incorporating additional variance information, such as pitch, duration, and energy, as inputs to the decoder. This helps alleviate the one-to-many mapping problem in TTS[1] [114]. FastSpeech 2 achieves better voice quality than its predecessor while maintaining the speed and controllability that made FastSpeech a landmark in non-autoregressive speech synthesis.

Despite impressive advances in naturalness, speaker timbre, and computational efficiency, the TTS systems described so far still face a fundamental limitation. Speech comprises multiple attributes (including pronunciation, rhythm, intonation, and timbre) yet these models attempt to synthesize acoustic features directly from text or phoneme sequences. This remains challenging due to the inherent one-to-many mapping problem, where a single text input can correspond to multiple valid acoustic realizations. While FastSpeech 2 tried to mitigate this issue by incorporating additional information such as pitch and energy, these enhancements alone are not sufficient to fully resolve the ambiguity. To more effectively bridge the information gap between text and speech, recent research has begun exploring the use of SSL speech representations in TTS, reflecting similar trends in VC. The core idea is that richer and semantically informed

[1]One-to-many mapping: in TTS, the one-to-many mapping problem refers to the fact that a single text input can correspond to multiple valid speech variations due to natural differences in prosody, emotion, and speaker style.

representations of text can help neural TTS models generate more natural and expressive speech. As a result, a new component, often referred to as *text-to-semantic* or *phoneme-to-semantic*, has been introduced into the modern TTS pipeline, in addition to the three modules already shown in Figure 2.10. This module transforms the output of the text analysis stage (typically normalized text or phoneme sequences) into SSL-based representations that are structurally and semantically closer to actual speech. These representations often leverage disentanglement techniques (Section 2.1.3), allowing them to capture speaker-independent yet content-rich features. By replacing symbolic input with these learned representations, the acoustic model's task becomes easier, as it now generates acoustic features from inputs more closely aligned with its target domain than raw text or phonemes. Several cutting-edge models have adopted this paradigm, including SpearTTS [7], Hierspeech [6] and its successor Hierspeech++ [69], Make-a-Voice [142], BaseTTS [5], Pheme [143], and Vevo [144]. These systems represent a significant step forward in closing the gap between text and speech, significantly advancing the quality and expressiveness of modern TTS systems.

More recently, another major shift in TTS has emerged through the integration of large language models (LLMs). Instead of predicting spectrograms, these systems operate directly on acoustic speech tokens obtained through neural codecs [145; 146], allowing TTS to be framed as a sequence modeling problem similar to text generation. A landmark contribution in this direction is VALL-E [147], which introduced a unified autoregressive language model trained jointly on text and codec-based speech representations. By learning the statistical relationships between textual prompts and target speech tokens, the system is able to perform zero-shot voice cloning, generating speech that captures both the voice timbre and speaking style of unseen speakers using only a short reference audio sample. The follow-up system VALL-E 2 [148] further improved this paradigm by scaling the model and training data, refining the neural codec, and strengthening the alignment between textual and acoustic token sequences. These enhancements enabled the model to approach human-level naturalness, significantly reducing artifacts and improving prosody, pronunciation, and speaker similarity.

2.4 Evaluation Metrics

Evaluating synthetic speech, whether generated through VC or TTS, requires assessing multiple perceptual and acoustic properties. A successful system must not only sound natural and intelligible, but also preserve the identity of the target speaker and produce prosody consistent with the intended linguistic content. Because no single metric captures all these aspects, the literature typically combines objective and subjective evaluations. Objective metrics offer quantitative measurements that are reproducible and scalable, whereas subjective evaluation directly reflects human perception that cannot be reliably inferred from automatic measures alone.

2.4.1 Objective Metrics

Objective metrics are automatically computed measures that quantify different aspects of synthesized or converted audio. Although they cannot fully replace human judgment, particularly for naturalness and expressiveness, they remain essential for large-scale benchmarking, model development, and ablation studies. In both VC and TTS, objective evaluation typically focuses on four complementary dimensions: *speech quality*, *speaker similarity*, *prosody*, and *intelligibility*.

A first group of metrics quantifies signal-level or spectral distortion between a generated waveform and a corresponding reference. One of the most established metric is the Mel-Cepstral Distortion (MCD) [149], which measures the frame-level difference between the mel-cepstral coefficients of the generated and target signals. Because two utterances rarely align frame by frame, MCD is usually computed after applying Dynamic Time Warping (DTW) [81], which provides an optimal temporal alignment between the sequences. A similar approach is used for F0-based metrics, such as the root mean squared error on fundamental frequency (F0 RMSE) [150] and the correlation of F0 contours [151]. These measures evaluate differences in pitch trajectories and are informative for assessing prosody preservation in VC or prosody generation accuracy in TTS. As with MCD, they require access to a reference signal and reliable alignment; therefore, they

are most suitable for controlled evaluations with parallel data and are less applicable in scenarios such as XVC or multilingual TTS, where direct reference could not exist.

While spectral and prosodic distortion metrics are important indicators of signal fidelity, they do not capture speaker identity, which is essential in both VC and TTS evaluation. To assess whether the system successfully reproduces the characteristics of the target speaker, researchers commonly measure speaker similarity between the generated audio and a reference recording of the target speaker. In practice, a pre-trained speaker verification model [152; 153; 154] is used to extract fixed-dimensional embedding vectors from both signals, and the cosine similarity between these embeddings is computed. Higher similarity scores indicate that the system is more faithfully able to produce the target speaker's timbre and vocal characteristics, whereas lower scores suggest that the generated speech deviates from the desired identity.

Objective evaluation also plays a key role in assessing intelligibility, which reflects the ability of VC systems to preserve the linguistic content of the input speech and of TTS systems to generate speech that correctly corresponds to the input text. A common strategy is to transcribe the generated audio using an ASR model [66] and compare the resulting transcript with the ground-truth text. This yields widely used measures such as the word error rate (WER) and the character error rate (CER), which quantify the differences between the system's transcription and the reference text at the word and character levels, respectively. Lower values indicate better performance, as they reflect a closer match between the linguistic content of the generated waveform and the actual text. A closely related metric is the phoneme error rate (PER), which evaluates intelligibility at the phoneme level and requires a text-to-phoneme converter such as [155]. Although these metrics are informative, they are not perfect, as they inherit the biases and limitations of the underlying ASR and phonemization models. Nonetheless, they remain extremely useful for detecting pronunciation errors or badly generated phonetic sequences, particularly when the VC or TTS system must generalize to new languages in a multilingual setting.

In recent years, automatic mean opinion score (A-MOS) prediction models have

emerged as a practical complement to human MOS evaluations (see below). Among these, UTokyo-SaruLab MOS (UTMOS) [156] has shown particularly strong correlation with human judgments across a wide range of conditions. By estimating audio quality directly from the waveform, UTMOS enables rapid large-scale comparisons without the time and cost requirements of listening tests. While such predicted scores cannot fully replace human evaluations, they provide an efficient proxy during system development and early-stage experimentation.

2.4.2 Subjective Metrics

Despite their usefulness and scalability, objective metrics provide only partial insight into perceptual quality, and none of them correlates perfectly with human judgments. For this reason, they are typically complemented with subjective evaluation methods.

The most widely adopted subjective metric is the MOS [157], in which human listeners rate audio samples on a five-point scale. Depending on the evaluation setup, listeners may assess different characteristics of the tested audio, such as overall quality, speaker similarity, or intelligibility. Overall quality MOS [157] reflects the naturalness of the audio, including fluency, clarity, and the absence of perceptual artifacts. Speaker similarity MOS [158] evaluates whether the synthesized or converted speech is perceived as originating from the intended target speaker. Intelligibility MOS [159] complements WER/CER/PER by assessing how well listeners understand the linguistic content, capturing subtle pronunciation errors that objective intelligibility metrics may either miss or exaggerate. In more specialized scenarios, other MOS variants are used, such as accent-related MOS (ACMOS) [160] for identifying accent leakage in multilingual settings and emotional MOS (EMOS/ECMOS) [161] for evaluating style or emotion rendering.

Alongside the standard MOS, many studies also employ the comparative MOS (CMOS) [162], a relative evaluation method in which listeners are presented with two audio samples (typically generated by a baseline and a system under test) and judge which one sounds better and by how much, assigning a comparative score based on a predefined scale (e.g., from –3 to +3). Positive values indicate preference for the second sample

and negative values indicate preference for the first. Unlike MOS, which measures absolute perceived quality, CMOS emphasizes relative differences between systems and is particularly useful when perceptual gaps between models are small. CMOS is also more sensitive in identifying small improvements in naturalness or similarity that might not be reflected clearly in absolute MOS ratings.

Finally, other popular subjective evaluations are the AB/ABX tests [163]. In an AB test, listeners are presented with two samples and asked to choose which one they prefer according to a specific criterion, such as naturalness or speaker similarity. The ABX test extends this format by adding a third reference sample, X; listeners must then determine whether A or B more closely matches X in the specified dimension (e.g., naturalness, similarity, or emotional expressiveness). Unlike CMOS, which quantifies how much one sample is preferred over another using a continuous comparative scale, AB and ABX tests focus purely on choice-based judgments. This makes them particularly effective for revealing perceptual preferences when differences between systems are really small.

Chapter 3

High-Quality Cross-Lingual Voice Conversion System

Summary

This chapter provides a comprehensive description of the voice conversion system developed during the first phase of the research, covering its three key components, the cross-lingual fine-tuning strategy designed to improve performance in multilingual conversion scenarios, and the experimental setup and results used to validate the method.

3.1 Introduction

The first goal of this research is to generate artificial data to extend in-house voice databases at reduced cost. To achieve this, the artificially generated voices must be nearly indistinguishable from human ones, with results comparable to those obtained through recording sessions with professional voice actors. An additional requirement is that the same voice must be polyglot, meaning it should be able to speak fluently and with the correct accent in languages other than its native tongue. Meeting these requirements is particularly challenging for two main reasons. First, despite the significant progress made in speech synthesis technologies in recent years, achieving a level

of quality that is almost indistinguishable from human speech is not trivial. Second, the requirement of polyglot capability makes the problem even more complex, as it is extremely difficult to obtain high-quality recordings of the same speaker in multiple languages, each with accurate pronunciation and a native-like accent.

In recent years, notable advancements have been made in tasks such XVC, particularly in some A2O VC variants [1; 2]. These developments have led to models increasingly capable of producing high-quality polyglot voices, even without requiring target speaker data in the language to be converted. Given this capability, XVC was identified as the task best suited to achieve the goals described above. For example, an American English speaker could be made capable of speaking an entirely new language, such as Italian, by providing speech from an Italian speaker as the source input to an XVC system.

The implemented A2O XVC system belongs to the family of *modern voice conversion approaches* that do not require parallel data to learn the conversion function (Section 2.2.2). Similarly to [1], the model leverages a Seq2Seq architecture that uses SSL representations as semantic/linguistic features and mel-spectrograms as acoustic features. However, unlike previous works that apply disentanglement techniques to SSL representations (Section 2.1.3), the proposed system uses the representations in their original form and allows the model itself to perform disentanglement during training. In this way, the model decides what information to remove and to what extent, in order to achieve the best possible performance during the training phase. This design choice is motivated by the observation that current disentanglement techniques can yield high-performing VC systems in intra-lingual scenarios but tend to degrade performance in cross-lingual settings, as demonstrated in the experiments reported in Section 3.3. Furthermore, the proposed system is able to both improve cross-lingual generation quality and reduce the amount of training data required. The main contributions of this work are as follows:

- Introduction of a bottleneck information extractor layer, similar to that in [4], which is shown to be sufficient to improve intelligibility and accent in cross-lingual scenarios without compromising speech quality or target speaker identity, and without requiring supplementary steps such as quantization.

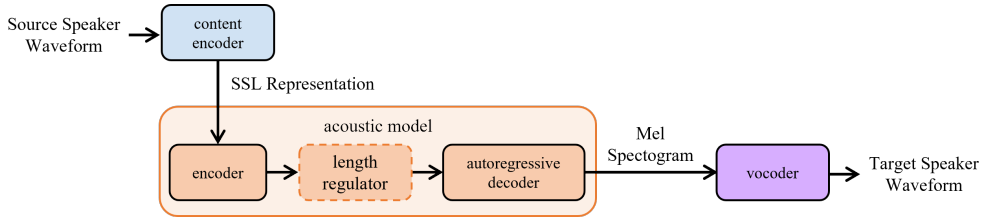


Figure 3.1: Overview of the proposed voice conversion system.

- Proposal of a cross-lingual pre-training fine-tuning strategy (Section 3.2.2) that not only further enhances VC performance in cross-lingual scenarios but also enables the use of low-quality data for data-intensive pre-training.

3.2 Model Architecture

As shown in Figure 3.1, the proposed system consists of three key components: a content encoder, an acoustic model, and a vocoder. The content encoder extracts an SSL speech representation from an audio waveform of any (source) speaker, the acoustic model maps this representation to a mel-spectrogram of the target speaker, and the vocoder converts the resulting mel-spectrogram into a time-domain waveform. The three components are trained independently, each with its own specific strategy and datasets, and then operate together during the inference (or conversion) stage.

3.2.1 Problem Definition

Consider a dataset D of a target speaker containing M utterances in the time domain. The i-th utterance of the target speaker is denoted as u_i, its features extracted by a pre-trained SSL model as $v_i \in \mathbb{R}^s$, and its mel-spectrogram as x_i, where $i \in [1, M]$.

The content encoder $\mathcal{C}$ takes the utterance u_i as input and generates the corresponding SSL representation v_i:

$$v_i = \mathcal{C}(u_i; w_{\mathcal{C}}) \tag{3.1}$$

where $w_{\mathcal{C}}$ represents the content encoder parameters. To remove speaker-specific in-

formation from v_i while minimizing its impact on other components (e.g., linguistic content), the acoustic model $\mathcal{A}$ employs a pre-net $\mathcal{P}$ that projects v_i into a space $\mathbb{R}^d$, where $d \ll s$. This dimensionality reduction imposes an information bottleneck, allowing the acoustic model to discard content-irrelevant information such as noise and speaker identity [4]. The acoustic model $\mathcal{A}$ then predicts the mel-spectrogram $\hat{x}_i$ from v_i:

$$\hat{x}_i = \mathcal{A}(\mathcal{P}(v_i; w_{\mathcal{P}}); w_{\mathcal{A}}) \tag{3.2}$$

where $w_{\mathcal{P}}$ and $w_{\mathcal{A}}$ represent the parameters of the pre-net and the acoustic model, respectively. Finally, the vocoder $\mathcal{V}$ takes $\hat{x}_i$ as input and generates $\hat{u}_i$:

$$\hat{u}_i = \mathcal{V}(\hat{x}_i; w_{\mathcal{V}}) \tag{3.3}$$

where $w_{\mathcal{V}}$ represents the vocoder parameters.

The three components of the system are trained independently. The content encoder $\mathcal{C}$ is an off-the-shelf SSL model (Section 2.1.2) pre-trained on a large multi-speaker dataset, whereas the acoustic model $\mathcal{A}$ and the vocoder $\mathcal{V}$ are trained specifically on target speaker data. After generating pairs (v_i, x_i) for each utterance u_i in the dataset D, the acoustic model $\mathcal{A}$ is trained by optimizing the following objective function:

$$\min_{w_{\mathcal{P}}, w_{\mathcal{A}}} L_{\mathcal{A}}(x_i, \mathcal{A}(\mathcal{P}(v_i; w_{\mathcal{P}}); w_{\mathcal{A}}))) \tag{3.4}$$

where $L_{\mathcal{A}}$ is a time–frequency domain loss function, such as the L_1 distance between target and predicted mel-spectrograms. In this setting, the pre-trained content encoder $\mathcal{C}$ is required to compute the input SSL representation v_i. The vocoder $\mathcal{V}$ can be trained either directly on ground-truth mel-spectrograms or on mel-spectrograms generated by the acoustic model $\mathcal{A}$:

$$\min_{w_{\mathcal{V}}} L_{\mathcal{V}}(u_i, \mathcal{V}(x_i; w_{\mathcal{V}})) \text{ or } \min_{w_{\mathcal{V}}} L_{\mathcal{V}}(u_i, \mathcal{V}(\hat{x}_i; w_{\mathcal{V}})) \tag{3.5}$$

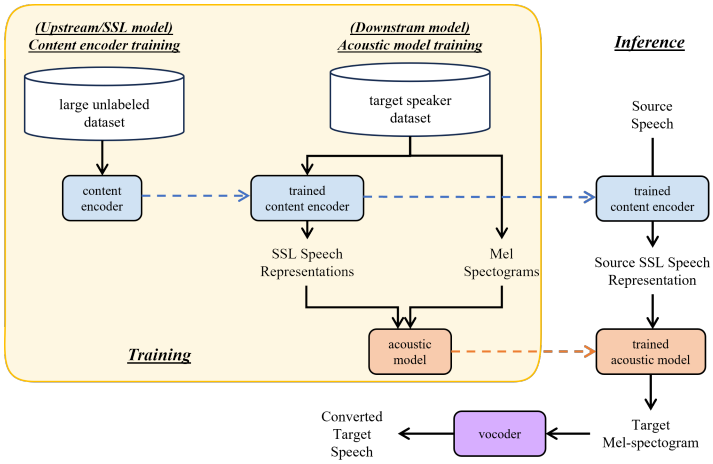


Figure 3.2: Training and inference procedures of the proposed cross-lingual voice conversion approach. Vocoder training (not shown) is performed using the same target speaker dataset and mel-spectrograms generated for the acoustic model.

where $L_{\mathcal{V}}$ is a loss function in the time domain. In this work, the vocoder $\mathcal{V}$ is trained using the ground-truth mel-spectrograms approach, making it completely independent from the other components.

By training only on target speaker data, the loss function in Eq. 3.4 makes the acoustic model target-speaker specific, meaning it generates mel-spectrograms corresponding to that speaker. The primary goal is to convert the source speaker's identity into that of the target speaker without altering other elements such as linguistic content, emotions, and accent. Consequently, it is possible to perform XVC by converting a source utterance from a language different from that spoken by the target speaker, thereby enabling the target speaker to become polyglot.

Figure 3.2 illustrates how each component interacts with the others during the training and inference phases. The content encoder is first trained on a large unlabeled dataset containing audio samples from a wide variety of speakers, enabling it to produce SSL representations. As mentioned earlier, in this work the content encoder is an off-the-shelf SSL model that has already been pre-trained and remains frozen throughout the entire pipeline. Once the content encoder is available, the training of the acoustic model can begin. Using a target speaker dataset, SSL speech representations and mel-

spectrograms are generated for all its audio files. For each pair of SSL representation and mel-spectrogram, the acoustic model predicts the target mel-spectrogram from the SSL input, and the L_1 distance between the predicted and ground-truth mel-spectrograms is computed to update the model parameters. Finally, the vocoder is trained on the same target speaker dataset used for the acoustic model, taking the same mel-spectrograms as inputs and the corresponding audio from the dataset as its ground-truth outputs.

During inference, the three trained models are applied in sequence. First, the content encoder extracts an SSL speech representation from the source speaker's audio. Next, the acoustic model predicts the target speaker's mel-spectrogram, and finally, the vocoder generates the converted target-speaker waveform.

3.2.2 Cross-Lingual Fine-Tuning

Although the system is already capable of achieving both intra- and cross-lingual A2O VC with good intelligibility and overall quality, some phonemes may still be mispronounced or influenced by the target speaker's accent, particularly in cross-lingual scenarios. To address this issue, a hybrid approach is proposed, leveraging the phonetic knowledge of an acoustic model trained on a specific language (e.g., French) to improve the pronunciation and linguistic accuracy of the target speaker (e.g., an English speaker) when speaking that language. Consider a *lang1-lang2* XVC scenario, where *lang1* is the language spoken by the source speaker and *lang2* is the language spoken by the target speaker. The objective is to enhance the acoustic model (trained on *lang2*) for *lang1* without degrading performance on the *lang2-lang2* intra-lingual conversion or other cross-lingual tasks (e.g., *lang3-lang2*).

Two strategies can be adopted to achieve this:

1. **Single-speaker pre-training**: Pre-train the acoustic model using data from a single target speaker who speaks *lang1*, then fine-tune the model using the dataset (or a subset) of the target speaker in *lang2*.

2. **Multi-speaker pre-training**: Pre-train the acoustic model on a large multi-

speaker dataset in which all speakers speak *lang1*, then fine-tune the model using the dataset (or a subset) of the target speaker in *lang2*.

Both strategies produce a target-specific acoustic model with improved performance in *lang1* compared to a model trained only on *lang2*, while maintaining the same level of performance in the *lang2*-*lang2* intra-lingual scenario. The two approaches are therefore interchangeable. However, internal experiments showed that the first strategy works well only if sufficient high-quality data (at least 10-12 hours) from a single *lang1* speaker are available; otherwise, the second strategy should be preferred, as it allows the use of larger amounts of lower-quality data (100-150 hours) from publicly available multi-speaker datasets. In this research, both strategies were explored. The first was mainly applied internally within the company, where high-quality recordings from multiple single speakers in different languages are available. The second was used for research purposes, as public datasets are often preferred for reproducibility.

Considering the second strategy with public datasets, the formal definition of the cross-lingual fine-tuning method is now provided. In addition to the target speaker dataset D containing audio in the *lang2* language, consider also a dataset D' consisting of N speakers, each with M' utterances in *lang1*. Let u'_{ji} denote the i-th utterance of the j-th speaker, v'_{ji} its SSL vector, and x'_{ji} its mel-spectrogram, where $j \in [1, N]$ and $i \in [1, M']$. First, the acoustic model $\mathcal{A}$ is pre-trained using the loss in Eq. 3.4, but with pairs (v'_{ji}, x'_{ji}) for each u'_{ji} from the dataset D'. Thus, Eq. 3.4 becomes:

$$\min_{w'_{\mathcal{P}}, w'_{\mathcal{A}}} L_{\mathcal{A}}(x'_{ji}, \mathcal{A}(\mathcal{P}(v'_{ji}; w'_{\mathcal{P}}); w'_{\mathcal{A}}))) \tag{3.6}$$

Second, the acoustic model $\mathcal{A}$ is fine-tuned using the dataset D (or a subset thereof) from the target speaker, as described in the previous paragraph. This procedure enables the acoustic model to acquire phonetic coverage of *lang1* during pre-training, which can then be leveraged during fine-tuning on *lang2*. As a result, the *lang2* target speaker exhibits improved *lang1* pronunciation and linguistic characteristics after conversion.

The cross-lingual fine-tuning technique offers several advantages. It not only im-

proves linguistic accuracy and pronunciation in XVC but also reduces the amount of target speaker training data required. While training the acoustic model without fine-tuning typically demands at least 10-12 hours of high-quality target speaker audio (or at least 100 hours of lower-quality, publicly available data for pre-training), the fine-tuning stage requires only about 2 hours. This makes it possible to obtain a target speaker model with significantly less high-quality data.

3.2.3 Content Encoder

The content encoder (blue box in Figure 3.1) is responsible for extracting semantically rich latent information from raw input audio, producing flexible and meaningful representations that capture the essential characteristics of the speech signal. Specifically, the content encoder embodies the "Any" part of the A2O VC task, serving as the preparation phase of the conversion process. It accepts audio from any source speaker (in any language) and converts it into SSL representations, which are dense embeddings containing comprehensive information from the input audio (Section 2.1).

In the proposed system, the content encoder coincides exactly with a pre-trained SSL model without any additional refinement steps. More specifically, the SSL model adopted is the SOTA WavLM [56], following the trend in recent speech synthesis studies that increasingly make use of it [4; 5; 112; 164]. WavLM, illustrated in Figure 2.5, was pre-trained by Microsoft on 60,000 hours of Libri-Light [165], 10,000 hours of GigaSpeech [166], and 24,000 hours of VoxPopuli [167], for a total of 94,000 hours of audio data. In this work, the publicly available WavLM-Large[1] model is used, with the output of its 15-th Transformer layer serving as the SSL representation for speech audio v_i in all experiments. This choice was motivated by the findings of [168; 169], which demonstrated the effectiveness of this layer in phone discrimination tests. Accordingly, s in Section 3.2.1 is configured to 1024, that is the dimension of the output vectors of WavLM-Large.

[1] `https://huggingface.co/microsoft/wavlm-large`

3.2.4 Acoustic Model

The acoustic model (orange box in Figure 3.1) performs the core conversion process within the VC pipeline and represents the "One" part of the A2O VC task. It is specific to the target speaker and takes as input the SSL representations generated by the content encoder, learning to map these into mel-spectrograms that reflect the target speaker's voice characteristics. Following [1], the model architecture resembles that of Tacotron 2 [135], featuring an encoder and an autoregressive decoder. Both the encoder and decoder are preceded by a feed-forward pre-net, while a final linear layer with n-mels units follows the decoder. Moreover, following [2], no attention module is used. The encoder pre-net is a feed-forward neural network composed of two linear layers with 256 units each, ReLU activations, and dropout [170]. The encoder consists of three 1D-convolutional layers with 512 units, kernel size 5, stride 1, padding 2, and ReLU activations. Following [76], an Instance Normalization (IN) layer is applied after each convolutional layer, as this helps further reduce speaker identity while preserving content information. The decoder predicts each spectrogram frame based on the encoder output and the previously generated frames. First, a decoder pre-net similar to that of the encoder is applied and then three LSTM layers with 768 units followed by a final linear layer with n-mels units are used.

As there is no attention mechanism, an optional duration adjustment strategy is included to mitigate potential length mismatches between the SSL input representations and the target mel-spectrogram sequence. WavLM generates features at a 20ms frame rate for audio sampled at 16 kHz (corresponding to a hop length of 320), which may result in mismatches when generating mel-spectrograms with different time resolutions (e.g., different hop lengths or higher sample rates). This strategy involves integrating a *length regulator* module between the encoder and the decoder, which interpolates the SSL features using PyTorch's *interpolate*[1] function with its default settings.

[1] `https://pytorch.org/docs/stable/generated/torch.nn.functional.interpolate.html`

3.2.5 Vocoder

The vocoder (purple box in Figure 3.1) completes the VC pipeline by converting the mel-spectrograms produced by the acoustic model into waveforms representing the target speaker's speech. It is also target speaker-specific and serves as the final stage of the conversion process, generating synthetic speech that closely resembles the target speaker.

Any model that converts mel-spectrograms into waveforms can serve as the vocoder component of the pipeline, such as the previously mentioned WaveNet [133], WaveRNN [171], or more recent approaches like MelGAN [172], HiFi-GAN [173], BigVGAN [174], or Vocos [175]. In the proposed VC system, HiFi-GAN is chosen for its SOTA real-time performance and its ability to generate high-quality speech from mel-spectrograms.

3.3 Experiments

To evaluate the effectiveness of the proposed XVC approach, experiments across a range of different tasks and datasets were conducted. All the experiments were run on a Linux machine with a single NVIDIA Titan RTX GPU with 24 GB of RAM.

The system was tested in two variants: *Proposed*, the VC method using the conventional training strategy described in Section 3.2.1, and *Proposed-F*, the alternative system employing the cross-lingual fine-tuning strategy described in Section 3.2.2. For comparison, two baseline models were selected: Soft-VC[1] [1] and kNN-VC[2] [112]. Both excel in XVC tasks, demonstrating strong content intelligibility and speech quality.

3.3.1 Experimental Setup

Experiments in both intra- and cross-lingual settings were conducted using LJSpeech [176] as the target speaker dataset for the training of both the acoustic model and the vocoder. LJSpeech is a public domain speech dataset consisting of 13,100 short audio clips of a single female speaker reading passages from 7 non-fiction books. The dataset

[1] `https://github.com/bshall/soft-vc`
[2] `https://github.com/bshall/knn-vc`

have a total length of approximately 24 hours.

Four languages were selected for evaluation: American English, European French, European Spanish, and Mandarin Chinese. For English, the LibriSpeech dataset [46] was used; for French and Spanish, Multilingual LibriSpeech (MLS) [177]; and for Mandarin Chinese, Aishell [178]. To ensure a fair comparison, the training of the LJSpeech acoustic model followed the same configuration as in Soft-VC, including the use of their HiFi-GAN vocoder pre-trained on LJSpeech, unless otherwise stated. More specifically, sampling rate was set to 16 kHz, the number of mel bins n-mels to 128, the SSL representation dimension s to 1024, and the bottleneck dimension d to 256.

For cross-lingual fine-tuning, four separate acoustic models (one per language) were first pre-trained using 150 hours of randomly selected audio from the training sets of LibriSpeech, French MLS, Spanish MLS, and Aishell, respectively. The choice of 150 hours was determined by the size of the Aishell training set, ensuring that all languages had an equal amount of training data. Each pre-trained acoustic model was then fine-tuned with only 2 hours of LJSpeech data for 15,000 steps with a batch size of 8. This value of 2 hours was found to be optimal for fine-tuning based on experiments conducted with a validation set of 100 LJSpeech utterances. It was further observed that using less than 2 hours of data was insufficient to achieve satisfactory overall quality.

3.3.2 Evaluation Metrics

Intelligibility, speaker similarity, accent preservation, and overall speech quality were evaluated using both subjective and objective methods (Section 2.4). For each language, a dedicated test set was created to serve as the source speech for conversion into LJSpeech. Each test set consisted of 60 utterances, obtained by randomly selecting 3 utterances from 20 speakers (10 male and 10 female). Specifically, the test-clean set of LibriSpeech was used for English; the test set of Spanish MLS for Spanish; the combined dev and test sets of French MLS for French; and the combined dev and test sets of Aishell for Chinese.

Objective Evaluations

To evaluate the *intelligibility* of the converted speech, WER and PER were calculated by comparing the transcriptions of the source and converted utterances. For the source utterances, ground-truth transcriptions were available, whereas the converted utterances were transcribed using two pre-trained ASR models: Whisper Medium[1] [66] for English, French, and Spanish, and FunASR[2] [179] for Chinese. In addition, *speaker similarity* (SSIM) was assessed using a trained speaker verification model[3]. Specifically, the cosine similarity was computed between the d-vectors [153] of each converted sample and that of the target speaker. The target speaker embedding is generated by averaging the d-vectors of 50 utterances from LJSpeech.

Subjective Evaluations

Subjective evaluations were carried out to assess both the *overall quality* and the *native accent level* of the converted speech. For each language, all 60 utterances from the corresponding test set were converted into LJSpeech using the two proposed methods and the two baselines, yielding a total of 300 utterances per language (60 ground-truth + 240 generated). For both evaluations, 20 native speakers of the respective languages were recruited to listen to the samples, which were presented in random order, and provide ratings. To assess overall quality, MOS was collected on a 5-point scale, where 1 corresponds to "very poor" and 5 to "excellent". The question asked was: "How do you rate the overall quality of this sample?". Accent evaluation was conducted using an accent comparative MOS (ACMOS). Participants rated each sample on how convincing the accent sounded, again on a 5-point scale, where 1 corresponds to "strong foreign accent" and 5 to "native". The question asked was: "Does the voice sound like that of a native speaker?". In this test, Proposed-F was set as the reference baseline. A negative Δ-ACMOS indicates worse performance compared to the baseline, a value of zero denotes equal performance (or the baseline itself), and a positive value indicates

[1] `https://huggingface.co/openai/whisper-medium`
[2] `https://github.com/modelscope/FunASR`
[3] `https://github.com/resemble-ai/Resemblyzer`

	English					French				
	WER	**PER**	**SSIM**	**MOS**	**Δ-ACMOS**	**WER**	**PER**	**SSIM**	**MOS**	**Δ-ACMOS**
Ground-truth	1.25	0.22	-	3.75 ± 0.04	-	8.74	2.70	-	3.77 ± 0.05	-
Discretization	18.81	10.28	90.86	-	-	70.00	60.54	90.54	-	-
Perturbation	2.82	0.88	**92.42**	-	-	27.17	12.08	**90.84**	-	-
Utterance std	4.76	2.56	88.62	-	-	15.75	5.58	84.03	-	-
Soft-VC	**2.63**	**0.73**	91.12	3.88 ± 0.04	-	19.13	6.91	89.16	3.73 ± 0.05	-0.13
kNN-VC	2.75	0.99	84.70	3.94 ± 0.04	-	16.55	6.24	81.26	3.82 ± 0.05	-0.10
Proposed	3.01	1.09	90.22	3.94 ± 0.05	-	14.16	5.00	88.67	**3.85 ± 0.05**	-0.09
Proposed-F	2.85	0.88	89.70	**3.96 ± 0.04**	-	**12.43**	**4.02**	88.25	3.83 ± 0.04	0.0
	Spanish					**Mandarin Chinese**				
Ground-truth	6.31	1.52	-	3.95 ± 0.04	-	7.72	0.43	-	4.07 ± 0.08	-
Discretization	75.04	66.74	**92.33**	-	-	90.01	77.35	**87.26**	-	-
Perturbation	17.95	6.52	89.64	-	-	47.73	20.03	87.14	-	-
Utterance std	12.51	3.66	85.43	-	-	28.34	5.47	82.27	-	-
Soft-VC	14.78	5.02	89.17	3.80 ± 0.04	-0.08	51.82	21.04	86.15	3.01 ± 0.08	-1.25
kNN-VC	11.91	3.47	80.31	3.83 ± 0.04	-0.06	35.24	9.78	77.50	3.25 ± 0.08	-0.47
Proposed	9.78	2.56	88.13	**3.87 ± 0.03**	-0.05	19.95	3.64	84.78	3.67 ± 0.08	-0.03
Proposed-F	**8.61**	**2.21**	86.53	3.86 ± 0.04	0.0	**18.39**	**3.33**	83.02	**3.73 ± 0.08**	0.0

Table 3.1: Objective and subjective evaluation. (middle) Results (%) in terms of intelligibility (W/PER ↓) and speaker similarity (SSMI ↑) considering three speaker and content disentanglement strategies: discretization, perturbation and per-utterance standardization. (bottom) W/PER, SSIM, overall quality (MOS ↑) with 95% confidence intervals, and the delta of the Accent Comparative MOS (Δ-ACMOS) for the proposed approaches and the baseline systems.

improvement. The intra-lingual English–English scenario was excluded, since accent consistency is inherently preserved after conversion.

3.3.3 Results

In this section, the effectiveness of the proposed approaches is evaluated in both intra- and cross-lingual scenarios. In addition, the ability of the systems to preserve the emotions of the source speakers is also demonstrated.

Speaker and Content Disentanglement

First, it is shown that applying disentanglement steps to the SSL representations, while effective in intra-lingual scenarios, results in a significant performance drop in cross-lingual tasks. A comparison was conducted between the proposed approach and three

disentanglement strategies described in Section 2.1.3, applied on top of the 15th-layer output of WavLM-Large: discretization, perturbation, and per-utterance standardization. These were implemented following the procedures outlined in [1], [68], and [63], respectively. For this test, the soft strategy was excluded, as it is already incorporated in the Soft-VC baseline, which is evaluated in the next experiment.

Table 3.1 (middle) reports intelligibility in terms of WER and PER, and speaker similarity (SSIM) of each model. Compared to the other methods, the proposed systems significantly enhance the intelligibility across all languages in cross-lingual scenarios while also maintaining very good performance for English. Particularly notable is the substantial improvement for Mandarin Chinese, where the proposed models significantly reduce both WER and PER compared with all other methods. Unlike perturbation, which fails to maintain the same performance observed for English, the overall effectiveness of the per-utterance standardization is significant. However, it does not adequately capture the target speaker's identity like the alternative approaches do. On the other hand, the speaker identity of the proposed systems is comparable to that of the other techniques, confirming that in an A2O VC setup, the information bottleneck at the beginning of the acoustic model is sufficient to ensure high overall performance.

Objective and Subjective Evaluation

After the initial assessment, the performance of the two proposed methods was further evaluated against the selected baselines. Table 3.1 (bottom) reports WER/PER and SSIM, along with MOS and ACMOS for each model. As previously observed, the proposed systems achieve consistently higher intelligibility across all cross-lingual tasks, while maintaining performance comparable to the baselines for English. Moreover, MOS results indicate that both proposed approaches surpass the baselines in overall quality. Although kNN-VC shows comparable MOS values, it fails to preserve speaker similarity effectively. An interesting observation is that the ground-truth samples yield lower MOS scores than all models for English and French, likely because VC systems act as denoisers, enhancing noisy inputs and thus influencing listener ratings. Regarding accent

assessment, Soft-VC and kNN-VC consistently produce negative Δ-ACMOS values, indicating that their samples are perceived as "less native-sounding" across all languages. Overall, while both proposed approaches outperform the baselines, Proposed-F provides superior intelligibility, higher MOS, and is also data efficient.

Ablation: Emotions Maintenance

In this section, the system's ability to preserve source emotions during VC is evaluated. To this end, the LJSpeech dataset was extended with the four emotions available in the Emotional Speech Dataset (ESD) [180]: angry, surprise, sad, and happy. Since this experiment focused exclusively on English, all emotional utterances from the 10 English speakers of ESD were converted into LJSpeech using the highest-quality setup. The resulting corpus, referred to as Emo-LJSpeech, contains roughly 35 hours of data, of which 11 hours were generated through VC.

To ensure compatibility with the original LJSpeech sample rate (22050 Hz), the length regulator approach described in Section 3.2.4 was applied to address mismatches between source SSL representation and target mel-spectrogram lengths. In this experiment, the parameters were set as follows: n-mels = 80, n-fft = 1024, window length = 1024, and hop length = 256. The acoustic model was trained on LJSpeech for 200,000 steps with a batch size of 32, using AdamW [181] with an initial learning rate of 0.0001 and a weight decay of 0.01. The learning rate was linearly increased during the first 4,000 steps (warm-up) and subsequently decreased linearly throughout training. These hyper-parameters were determined through an extensive search using a validation set of 100 utterances from LJSpeech. For cross-lingual fine-tuning, the acoustic model was pre-trained on 13 hours of data from the English subset of ESD (including neutral utterances) and then fine-tuned with 2 hours of LJSpeech for 15,000 steps using a batch size of 8. For waveform generation, an internal HiFi-GAN vocoder operating at 22050 Hz and trained on a proprietary multi-speaker dataset was employed.

To assess the quality of the newly generated emotional corpus, a test set was created by randomly selecting, for each of the four emotions, 3 utterances from 5 speakers of

	WER	**PER**	**SSIM**	**MOS**	Δ-ECMOS Angry	Δ-ECMOS Happy	Δ-ECMOS Sad	Δ-ECMOS Surprise
Ground-truth	10.08	4.89	-	3.95 ± 0.04	-	-	-	-
Proposed	11.01	5.53	89.54	3.90 ± 0.05	+0.39	+0.63	+0.41	+0.92
Proposed-F	11.55	5.80	88.37	3.80 ± 0.04	+0.86	+0.74	+0.24	+0.80

Table 3.2: Results (%) measuring intelligibility (W/PER ↓), speaker similarity (SSMI ↑), overall quality (MOS ↑) with 95% confidence intervals, and the delta of the Emotion Comparative Mean Opinion Score (Δ-ECMOS) vs neutral samples for the emotional experiment.

the ESD English subset, resulting in 60 emotional utterances. Objective and subjective evaluations were performed by comparing the ground-truth recordings with the 60 utterances converted into LJSpeech using the *Proposed* and *Proposed-F* models trained as described above. Alongside WER, PER, SSIM, and MOS, the ability of the systems to preserve emotions was analyzed through an emotion comparative MOS (ECMOS). This evaluation was conducted separately for each emotion (denoted as e). 20 native participants rated the samples on a 5-point scale, indicating the extent to which the corresponding emotion e (e.g., happy) was perceived, with 1 denoting "totally not e" and 5 representing "very e". The question was: "Does the voice sound e?". In addition to the 60 emotional utterances, 60 neutral-style utterances were generated with both models to serve as references. The reported Δ-ECMOS reflects the degree to which a given emotion was perceived in comparison to its neutral counterpart.

The results, presented in Table 3.2, show that intelligibility and overall quality are very close to those of the ground-truth recordings. Furthermore, both systems consistently achieved positive Δ-ECMOS values, confirming their ability to effectively preserve source emotions during conversion.

3.4 Conclusion and Limitations

In this work, an A2O XVC system capable of generating human-level, native-sounding polyglot voices without requiring multilingual data from the target speaker was developed. The study showed that several content–speaker disentanglement strategies applied

to SSL features lead to degraded performance in cross-lingual scenarios. To address this limitation, it was demonstrated that introducing a trainable bottleneck layer, combined with cross-lingual fine-tuning, enhances both the intelligibility and accent of the converted voices. In addition, the proposed models were shown to preserve source emotions throughout the conversion process. The experiments highlighted consistent improvements over strong baseline methods across all evaluated metrics. Moreover, MOS tests with multiple native participants confirmed that the outputs of the proposed approaches closely approximate human recordings in terms of overall quality.

However, some limitations remain. To achieve high-quality polyglot voices, an A2O system with a target-specific acoustic model was employed. This approach requires a substantial amount of training data. Although cross-lingual fine-tuning reduces the data requirement, the need for 2 hours of target speaker data can still be considered significant. Developing approaches that further lower the amount of data needed to train target-specific models is left for future research. In addition, from the experiments conducted, the SSIM of the proposed methods is slightly lower, though comparable, to that of other approaches. This discrepancy likely arises from the absence of explicit disentanglement strategies applied to the SSL representations. While the introduction of the bottleneck layer helps, more challenging conversions, such as cross-gender scenarios (male to female or female to male), still require improvement. Although the quality in these cases is acceptable, it does not yet reach the level of intra-gender scenarios (female to female or male to male). Nonetheless, sacrificing a small degree of speaker similarity enables improvements in other aspects such as intelligibility and accent, particularly in cross-lingual tasks. A reasonable hypothesis is that implementing a more efficient disentanglement scheme would allow the model to achieve more balanced, high-quality performance across all dimensions. This research direction has been further explored and is discussed in Chapter 4.

Chapter 4

Eta-WavLM: Efficient Speaker Identity Removal in Self-Supervised Speech Representations

Summary

This chapter provides a comprehensive description of a new, efficient, and training-free disentanglement method designed to generate speaker-independent SSL representations. The method was developed to improve the performance of the VC system presented in Chapter 3, by addressing the limitations of existing disentanglement approaches.

4.1 Introduction

The previous chapter introduced an A2O VC system that demonstrated strong performance in both intra- and cross-lingual settings. The experiments showed that, without applying explicit disentanglement steps to SSL features, the system achieved levels of

speaker similarity comparable to existing methods, while delivering superior intelligibility and accent preservation in cross-lingual tasks. Nevertheless, the speaker identity of the target speaker remains an area with room for improvement, particularly in challenging scenarios such as cross-gender conversions. At the same time, the experiments in Chapter 3 also highlighted a major limitation of current disentanglement strategies: while they can strengthen target speaker identity, they consistently degrade performance in cross-lingual settings. This creates a trade-off that existing approaches fail to resolve.

Motivated by this, the work presented in this chapter proposes a new disentanglement method for SSL representations that is simple to implement, computationally efficient, and effective at improving the overall performance of the VC system. The core idea is to isolate and subtract the speaker-dependent component from SSL features through a simple linear projection, yielding representations that retain linguistic information while minimizing speaker identity. The implemented strategy can be considered part of the family of *perturbation- and normalization-based approaches* described in Section 2.1.3, since SSL representations are adjusted to remove the contribution of speaker identity without requiring quantization or additional training strategies involving neural networks or loss functions. Additionally, in line with the other approaches in this family, the method is designed to be both lightweight and readily implementable. The main contributions of this work are as follows:

- Introduction of an efficient strategy for generating speaker-independent SSL representations by solving a simple linear equation.
- Demonstration that the proposed method actually generates speaker-independent representations, and therefore reduces speaker classification accuracy by nearly 30% compared with standard SSL representations.
- Enhancement of the performance of the VC model described in Chapter 3 through the use of features derived from the proposed approach rather than the original SSL representations. In particular, the method improves target speaker identity, linguistic content preservation, and overall system quality. These findings align

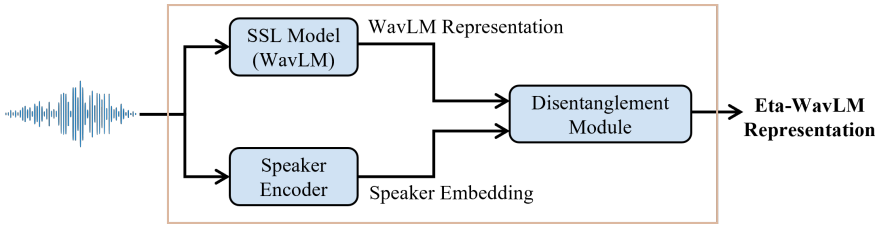


Figure 4.1: High-level overview of the proposed Eta-WavLM approach.

with the hypotheses of prior work [60; 164], indicating that effectively addressing speaker disentanglement can yield significant performance improvements in content-related speech tasks.

4.2 Method

The proposed approach can be viewed as an extension of an SSL model, implemented as an offline module designed to extract disentangled eta representations. As shown in Figure 4.1, the method is composed of three key components: an *SSL model* that extracts an SSL representation from a raw waveform, a *speaker encoder* that generates a speaker embedding from the same waveform, and a *disentanglement module* that produces a speaker-independent eta representation from the input SSL representation, conditioned on the speaker embedding. In this work, both the SSL model and the speaker encoder are off-the-shelf pre-trained models that are not further trained or fine-tuned. The main contribution lies in the design of the disentanglement module. For consistency with Chapter 3, WavLM is used as the SSL model. Accordingly, the SSL representations are referred to as *WavLM representations*, and the output of the proposed system as *Eta-WavLM representations*.

4.2.1 Problem Definition

The primary concept underlying the disentanglement module is to decompose a WavLM representation $\mathbf{s}$ into speaker-dependent $\mathbf{d}$ and speaker-independent $\boldsymbol{\eta}$ components. For a given data point, $\mathbf{s}$ and $\mathbf{d}$ can be easily obtained using a pre-trained WavLM model

and a pre-trained speaker encoder, respectively. Based on the assumption that different factors of speech (e.g., speaker identity, linguistic content, prosody) can be separated into distinct subspaces within SSL representations, and motivated by the goal of exploring whether a simple approach can effectively address the challenging problem of speaker-content disentanglement, an additive relationship between the speaker-dependent and speaker-independent components is assumed:

$$\mathbf{s} = f(\mathbf{d}) + \boldsymbol{\eta} \tag{4.1}$$

This formulation is supported by prior findings showing that SSL models encode different aspects of speech in a structured manner across layers [182], rather than arbitrarily entangling them. If these aspects were entirely interwoven in a highly nonlinear fashion, such structured encoding would likely not be observed. Moreover, recent studies have demonstrated that factors such as phonetic and speaker labels are, to a large extent, linearly separable in SSL representations [183]. Ideally, $\boldsymbol{\eta}$ should capture linguistic, prosodic, and environmental information (e.g., recording conditions), provided that $\mathbf{d}$ effectively represents speaker characteristics. The importance of selecting an appropriate speaker encoder for extracting $\mathbf{d}$ in this context will be discussed in Section 4.3.3. Consequently, the speaker-independent component $\boldsymbol{\eta}$ can be computed as:

$$\boldsymbol{\eta} = \mathbf{s} - f(\mathbf{d}) \tag{4.2}$$

Thus, to extract speaker-independent Eta-WavLM representations, the goal is to model the function $f(\cdot)$ so as to remove as much speaker-dependent information as possible from the original WavLM representations. The next section details how $f(\cdot)$ is modeled.

4.2.2 Computation of Latent Basis and Bias

Based on the hypothesis that large embedding spaces tend to linearize complex nonlinear relationships [183; 184], the function $f(\cdot)$ is approximated using a linear model. Consider a multi-speaker dataset composed of U utterances of raw speech. Let a generic

utterance be denoted as $\mathbf{u}_i$, its speaker embedding extracted by a pre-trained speaker encoder $\mathcal{E}$ as $\mathbf{e}_i \in \mathbb{R}^V$ with $i \in [1, U]$, and its WavLM representation extracted by a pre-trained WavLM model $\mathcal{S}$ as $\mathbf{S}_i = [\mathbf{s}_1, \cdots, \mathbf{s}_M]^T$, where $\mathbf{s}_m \in \mathbb{R}^Q$ represents the m-th frame, and M is the sequence length. Since M can be large, L frames are randomly extracted from each utterance, yielding a fixed-length representation $\mathbf{S}_i \in \mathbb{R}^{L \times Q}$. Consequently, the dataset's WavLM representation, obtained by stacking all $\mathbf{S}_i$, is given by $\mathbf{S} \in \mathbb{R}^{N \times Q}$, where $N = U \times L$ for simplicity. To align $\mathbf{e}$ with the sequence length of $\mathbf{S}$, it is expanded by replicating it L times along the frame axis, resulting in $\mathbf{E}_i \in \mathbb{R}^{V \times L}$. This is justified by the fact that the speaker embedding encodes speaker-level information, which remains constant across all frames of an utterance. Consequently, the dataset's embedding representation, obtained by stacking all $\mathbf{E}_i$, is given by $\mathbf{E} \in \mathbb{R}^{V \times N}$. Additionally, since V can be large, principal component analysis (PCA) is applied to reduce its dimensionality to $P < V$, yielding $\mathbf{D} \in \mathbb{R}^{P \times N}$. This reduction helps eliminate redundancy while retaining the most informative components. The importance of this step will be analyzed in Section 4.3.3.

Given $\mathbf{S} \in \mathbb{R}^{N \times Q}$ and $\mathbf{D} \in \mathbb{R}^{P \times N}$, their relationship can be modeled as:

$$\mathbf{S} = \mathbf{D}^T \mathbf{A} + \mathbf{1}_N \mathbf{b}^T \tag{4.3}$$

where $\mathbf{A} \in \mathbb{R}^{P \times Q}$ and $\mathbf{b} \in \mathbb{R}^{Q \times 1}$ are learnable parameters. For simplicity, this equation can be rewritten as:

$$\mathbf{S} = \tilde{\mathbf{D}}^T \tilde{\mathbf{A}} \tag{4.4}$$

where $\tilde{\mathbf{D}}^T = \begin{bmatrix} \mathbf{D}^T & \mathbf{1} \end{bmatrix}$ and $\tilde{\mathbf{A}}^T = \begin{bmatrix} \mathbf{A}^T & \mathbf{b} \end{bmatrix}$. The optimization problem to solve is then:

$$\tilde{\mathbf{A}}^* = \arg\min_{\tilde{\mathbf{A}}} ||\mathbf{S} - \tilde{\mathbf{D}}^T \tilde{\mathbf{A}}||_F \tag{4.5}$$

which can be solved via the pseudo-inverse:

$$\tilde{\mathbf{A}}^* = (\tilde{\mathbf{D}}^T\tilde{\mathbf{D}})^{-1}\tilde{\mathbf{D}}^T\mathbf{S} \tag{4.6}$$

where $\tilde{\mathbf{A}}^{*T} = \begin{bmatrix}\mathbf{A}^{*T} & \mathbf{b}^*\end{bmatrix}$. From this point onward, $\mathbf{A}^*$ and $\mathbf{b}^*$ will be referred to as the latent basis and bias, respectively. At this stage, the function $f(\cdot)$ has been learned, completing the first step. With $\mathbf{A}^*$ and $\mathbf{b}^*$ known, the disentanglement module can now generate Eta-WavLM representations.

4.2.3 Creation of Eta-WavLM Representations

During inference, the proposed system (Figure 4.1) generates speaker-independent Eta-WavLM representations directly from raw waveforms. Given an utterance $\mathbf{u}'$, first the pre-trained WavLM model $\mathcal{S}$ extracts a WavLM representation $\mathbf{S} \in \mathbb{R}^{K\times Q}$:

$$\mathbf{S} = \mathcal{S}(\mathbf{u}'; \mathbf{W}_{\mathcal{S}}) \tag{4.7}$$

where $\mathbf{W}_{\mathcal{S}}$ represents the frozen parameters of the WavLM model, and K is the sequence length. Next, the pre-trained speaker encoder $\mathcal{E}$ generates a speaker embedding $\mathbf{e} \in \mathbb{R}^{V\times 1}$ from the same utterance $\mathbf{u}'$:

$$\mathbf{e} = \mathcal{E}(\mathbf{u}'; \mathbf{W}_{\mathcal{E}}) \tag{4.8}$$

where $\mathbf{W}_{\mathcal{E}}$ represents the frozen parameters of the speaker encoder. To reduce the dimensionality of $\mathbf{e}$, PCA is applied, producing $\mathbf{d} \in \mathbb{R}^{P\times 1}$:

$$\mathbf{d} = PCA(\mathbf{e}; \mathbf{C_{PCA}}) \tag{4.9}$$

where $\mathbf{C_{PCA}}$ denotes the matrix of principal components obtained during the PCA process executed in the first step (Section 4.2.2). Finally, the disentanglement module

H extracts a speaker-independent Eta-WavLM representation $\boldsymbol{\eta} \in \mathbb{R}^{K \times Q}$:

$$\boldsymbol{\eta} = H(\mathbf{S}; \mathbf{d}, \mathbf{A}^*, \mathbf{b}^*) \tag{4.10}$$

where $\mathbf{A}^*$ and $\mathbf{b}^*$ are the latent basis and bias obtained at the end of the first step (Section 4.2.2), and the disentanglement module $H(\cdot)$ is implemented as:

$$H(\mathbf{S}) = \mathbf{S} - \mathbf{1}_K(\mathbf{d}^T\mathbf{A}^* + \mathbf{b}^*) \tag{4.11}$$

4.3 Experiments

To evaluate the effectiveness of the proposed approach, a speaker-related and a content-related task were selected. The primary objective of the first experiment is to determine whether the Eta-WavLM representations extracted by the proposed method exhibit minimal or no speaker-specific characteristics, thereby confirming the achievement of the desired disentanglement. The second experiment aims to assess whether the disentangled representations provide benefits in real-world tasks such as VC, where preserving linguistic content and achieving high similarity to the target speaker's voice are essential. For all experiments, the following setup was employed:

1. **Framework and hardware**: All experiments were executed on a Linux machine with a single NVIDIA GeForce RTX 3090 GPU with 24 GB of RAM.

2. **Dataset**: The full training set of the multi-speaker LibriSpeech [46] dataset was used for computing $\mathbf{A}^*$ and $\mathbf{b}^*$, as described in Section 4.2.2.

3. **SSL model**: As discussed in Section 4.2, the SOTA WavLM was employed as the pre-trained SSL model $\mathcal{S}$. Following Chapter 3, the official WavLM-Large model was used, with the output of its 15th Transformer layer taken as the representation $\mathbf{S}$. Accordingly, Q in Sections 4.2.2 and 4.2.3 was set to 1024, corresponding to the dimensionality of the WavLM-Large output vectors. Additionally, the fixed number of frames L in Section 4.2.2 was set to 100.

4. **Speaker encoder**: The SOTA ECAPA-TDNN [154] was chosen as the pre-trained speaker encoder model $\mathcal{E}$. ECAPA-TDNN extracts speaker embeddings from input speech by leveraging channel attention, propagation, and aggregation mechanisms to produce robust and discriminative speaker representations $\mathbf{d}$. A publicly available ECAPA-TDNN model[1] pre-trained by SpeechBrain [185] was used. Accordingly, V in Sections 4.2.2 and 4.2.3 was set to 192, corresponding to the dimensionality of the embeddings extracted by the model. The choice of ECAPA-TDNN as the speaker encoder is justified in Section 4.3.3.

5. **Dimensionality reduction**: PCA[2] was used to reduce V to P, as described in Sections 4.2.2 and 4.2.3. Specifically, P was set to 128, and motivation behind this choice is discussed in Section 4.3.3.

4.3.1 Speaker-Related Classification Task

To evaluate whether the proposed approach effectively reduces speaker information in the WavLM representations, thereby creating speaker-independent Eta-WavLMs, a speaker classification task was designed. Intuitively, since this is a speaker-related task, a model can only perform well if the input representations retain a significant amount of speaker-specific information. Conversely, if the input representations are speaker-independent, the model will struggle to achieve high classification accuracy. Thus, the hypothesis is that Eta-WavLM representations will yield lower accuracy than the original WavLMs, which are known to encode speaker-specific characteristics. To test this hypothesis, 10 speakers from the LibriSpeech test-clean set were randomly selected, resulting in a total of 1,285 utterances. Then, for each utterance, both the WavLM representation $\mathbf{S}$ (Equation 4.7) and the Eta-WavLM representation $\boldsymbol{\eta}$ (Equation 4.11) were computed. A multi-class support vector machine (SVM) classifier was trained and evaluated on both representation sets using a 5-fold cross-validation setup, recording the classification accuracy for each fold. In addition, the mean and the standard deviation across the 5

[1] `https://huggingface.co/speechbrain/spkrec-ecapa-voxceleb`
[2] `https://scikit-learn.org/1.6/modules/generated/sklearn.decomposition.PCA.html`

	FOLD1	FOLD2	FOLD3	FOLD4	FOLD5	MEAN ± STD
WavLM	83.46	82.33	80.85	83.30	81.55	82.30 ± 0.01
Eta-WavLM	53.82	55.14	58.77	53.94	56.96	**55.73 ± 0.01**

Table 4.1: Classification accuracy results (%) for WavLM and Eta-WavLM across the 5 folds of cross-validation (ACC ↓). Lower accuracy indicates better performance, as it reflects reduced speaker-related information.

folds were reported. SVM was chosen for its simplicity and well-known robustness in handling high-dimensional feature spaces and small datasets. The results are presented in Table 4.1. As expected, the Eta-WavLM representations achieve significantly lower classification accuracy compared to the original WavLM ones (paired t-test yielded a *T-Statistic* of 18.41 and a *p-value* of 5.12×10^{-5}, rejecting the null hypothesis $p < 0.05$). This accuracy reduction provides clear evidence that the proposed approach is effective in reducing speaker-specific information from the standard WavLM representations.

To further validate the method, WavLM and Eta-WavLM representations were also visualized. Five speakers from the LibriSpeech test-clean set were randomly selected and 10 utterances per speaker were extracted. For each utterance, the WavLM and Eta-WavLM representations were computed and projected into a two-dimensional space using UMAP [186]. As shown in Figure 4.2 (left), the UMAP projection of the WavLM representations cluster to regions corresponding to the individual speakers, suggesting the presence of strong speaker-specific information. In contrast, the projection of the Eta-WavLM representations in Figure 4.2 (right) does not show a discernible cluster of speakers, indicating that the proposed transformation effectively minimizes speaker-specific information. These visualizations reinforce the quantitative results from the speaker classification task by providing an intuitive and qualitative demonstration of the speaker-independence of the Eta-WavLM representations. The absence of speaker-specific clusters in the Eta-WavLM projection aligns with the significantly lower speaker classification accuracy observed in Table 4.1, further reinforcing the conclusion that the proposed approach successfully disentangles speaker-related information.

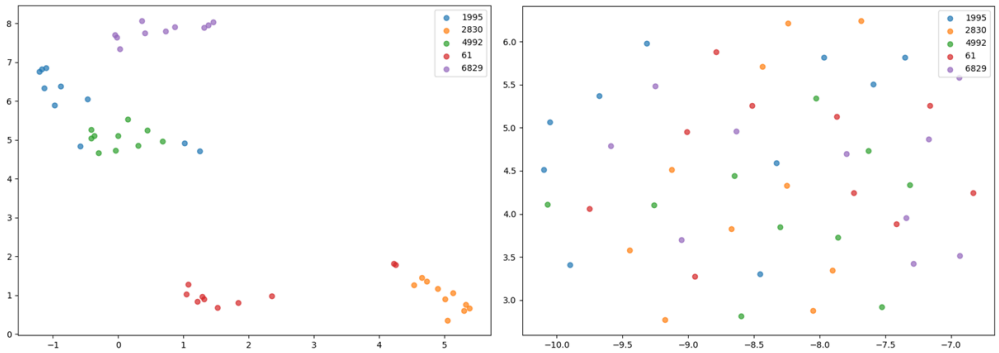


Figure 4.2: UMAP projections of the WavLM (left) and Eta-WavLM (right) representations extracted from 10 utterances of 5 speakers (with ids 1995, 2830, 4992, 61, 6829) from the LibriSpeech test-clean set.

4.3.2 Voice Conversion Task

Despite providing good insights into the reduction of speaker-related information in the Eta-WavLM representations, the speaker classification task does not assess whether this reduction affects other critical components, such as linguistic content. To evaluate this, a content-related VC task was designed, where the preservation of linguistic content and the accurate representation of the target speaker's identity are both essential for creating a high-quality conversion system. This dual requirement makes VC an ideal framework for evaluating whether the proposed approach removes speaker-specific components while preserving other essential features. The hypothesis is that the proposed Eta-WavLM representations will improve VC performance compared to both the original WavLM representations and other SOTA disentanglement methods.

Model Architecture

For this experiment, the same A2O VC system described in Chapter 3 and illustrated in Figure 3.1 was employed. The main difference lies in the content encoder: instead of directly using the WavLM model, this component was replaced with the eta content encoder shown in Figure 4.1. In contrast, the acoustic model remained identical to the implementation in Chapter 3 and was trained from scratch using the different types of representations provided as input. For the vocoder, the multi-speaker and recently

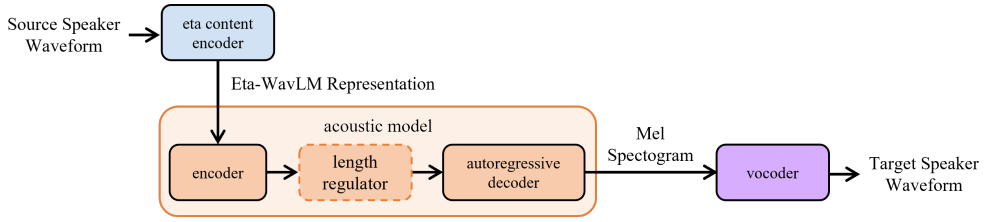


Figure 4.3: Overview of the proposed voice conversion system, where the standard content encoder is replaced by the eta content encoder.

introduced Vocos model[1] [175] was adopted, as it is known for producing high-quality speech in real time. The overall VC system incorporating the eta content encoder is illustrated in Figure 4.3. In addition, the content encoder was adapted to output not only WavLM or Eta-WavLM representations, but also other baseline disentanglement variants selected for experimental comparison, as described in the following section.

Baseline Approaches

The Eta-WavLM representations were evaluated against several baselines, including the direct use of the WavLM model (i.e., the same content encoder employed in Chapter 3), as well as four prominent disentanglement strategies from the literature: perturbation, per-utterance standardization, soft unit creation, and vector quantization (VQ). The first three were selected for consistency with the experiments in Chapter 3, while the earlier discretization approach was replaced with the more recent SOTA quantization method based on VQ. Specifically, for this latter case, the RepCodec model [67] was trained following the official guidelines[2], substituting HuBERT with WavLM. This comparison resulted in 6 distinct content encoders: one based on the unmodified WavLM representations and five derived from the application of the different disentanglement strategies, including the proposed method (Figure 4.3).

Experimental Setup

Since the focus of this work is on cross-gender rather than cross-lingual scenarios, English

[1] `https://huggingface.co/charactr/vocos-mel-24khz`
[2] `https://github.com/mct10/RepCodec`

was chosen as the evaluation language, and two target speakers with distinct characteristics, genders, and recording conditions were selected to train the acoustic model:

- **LJSpeech** [176] (F): The same female single-speaker dataset used in Chapter 3, containing approximately 24 hours of clean, high-quality read English speech.
- **Elliot Miller** (M): A male single-speaker dataset consisting of 38 hours of read English speech, extracted from the multi-speaker and multilingual M-AILABS Speech Dataset[1]. To ensure a fair comparison with LJSpeech, 24 hours were randomly selected.

While LJSpeech provides clean and consistent speech data, Elliot Miller introduces more challenging conditions. This contrast in target speaker profiles was intentionally selected to evaluate the robustness of the VC system under varied conditions. For each target speaker, audio samples were first downsampled to 16 kHz, and SSL representations were extracted using all 6 content encoders. Each audio file was then transformed into its corresponding mel-spectrogram according to the Vocos configuration [175]: sampling_rate = 24 kHz, n_fft = 1024, hop_length = 256, and n-mels = 100. Finally, for each pair of (representation, mel-spectrogram), a target-specific acoustic model was trained, using the representation as input and the mel-spectrogram as the target. In total, 12 distinct acoustic models (6 representation types $\times$ 2 speakers) were trained.

Evaluation Metrics

Both objective and subjective evaluations were carried out to assess the intelligibility, speaker similarity, and overall quality of the converted speech. For this purpose, a test set of 60 utterances was created by randomly selecting 3 utterances from 20 speakers (10 male and 10 female) extracted from the test-clean set of LibriSpeech. All these utterances were then converted into the two target speakers, LJSpeech and Elliot Miller, using the proposed method and the five baseline approaches, resulting in a total of 420 samples per target speaker (60 ground-truth + 360 generated). The evaluation employed

[1] `https://github.com/imdatceleste/m-ailabs-dataset`

Table 4.2: Objective and subjective evaluation of the voice conversion task. Results (%) in terms of intelligibility (W/PER ↓), target speaker similarity (SSIM ↑), and overall quality (MOS ↑) with 95% confidence intervals for the proposed Eta-WavLM and the baseline methods. WavLM was used as the SSL model across all approaches.

	LJSpeech				**Elliot Miller**			
	WER	**PER**	**SSIM**	**MOS**	**WER**	**PER**	**SSIM**	**MOS**
Ground-truth	3.22	5.47	-	3.85 ± 0.04	3.22	5.47	-	3.85 ± 0.04
Perturbation [68]	6.29	7.32	91.69	3.45 ± 0.06	10.76	8.43	87.41	3.13 ± 0.07
Utterance std [63]	4.13	7.32	90.34	3.80 ± 0.04	5.16	6.68	85.91	3.41 ± 0.06
Soft [1]	4.82	5.94	91.81	3.84 ± 0.05	5.50	6.75	86.69	3.32 ± 0.06
Vector quantization [67]	4.79	6.08	90.05	3.90 ± 0.05	7.72	7.56	86.30	3.50 ± 0.06
WavLM (Chapter 3)	4.56	5.84	89.52	3.84 ± 0.05	5.14	6.38	86.18	3.66 ± 0.06
Proposed (Eta-WavLM)	**3.81**	**5.63**	**92.46**	**4.00 ± 0.05**	**4.64**	**6.09**	**89.32**	**3.79 ± 0.05**

the same metrics as in Section 3.3.2: word error rate (WER), phoneme error rate (PER), speaker similarity (SSIM), and mean opinion score (MOS).

Results

In this section, objective and subjective results for the VC experiment are reported. Table 4.2 shows WER/PER, SSIM, and MOS for the two target speakers: LJSpeech (first column) and Elliot Miller (second column). Compared to the other methods, Eta-WavLM significantly enhances conversion intelligibility, achieving the lowest error rates for both target speakers. For LJSpeech, it closely approaches the ground-truth WER and PER values, demonstrating a high level of linguistic content preservation compared to all baselines. A similar pattern is observed for Elliot Miller, where Eta-WavLM outperforms the baselines, confirming its robustness even under more challenging acoustic conditions. Notably, the perturbation approach exhibits the highest error rates, particularly for Elliot Miller, suggesting that this approach excessively distorts linguistic and semantic information in the input speech. In terms of speaker similarity, the proposed method achieves the best SSIM scores for both target speakers, outperforming both the original WavLM representations and all other disentanglement strategies. While the soft approach yields comparable results for LJSpeech, it struggles with the noisier Elliot

Miller speaker, highlighting that some disentanglement methods are more sensitive to challenging acoustic conditions. In contrast, as already seen in Chapter 3, using WavLM directly results in lower speaker similarity, reinforcing the notion that some speaker-dependent information of the source speaker remains embedded also after conversion and this affects the overall performance of the VC system. Finally, regarding overall speech quality, Eta-WavLM achieves the highest MOS scores, surpassing all baselines. Interestingly, the VQ approach demonstrates relatively strong MOS ratings but fails to maintain the same level of linguistic and semantic integrity, as evidenced by its higher WER and PER values, especially for Elliot Miller. Conversely, as with intelligibility, the perturbation yields the lowest MOS values, further indicating that speech modification negatively impacts also naturalness.

These results confirm the hypothesis that Eta-WavLM effectively disentangles speaker information while preserving linguistic content, achieving the best balance between intelligibility, speaker similarity, and speech quality. Moreover, the consistent improvements across both target speakers underline its robustness, demonstrating that the proposed approach not only reduces speaker-related information more effectively than existing methods but also avoids degradation of other features.

4.3.3 Ablation: Speaker Encoder and PCA

This section analyzes the impact of the speaker encoder on the creation of effective speaker embeddings $\mathbf{d}$. Since the proposed approach aims to decompose SSL representations into speaker-dependent and speaker-independent components, it is essential that $\mathbf{d}$ capture speaker-specific characteristics without encoding other critical factors such as linguistic content, prosody, or phonetic details. If the extracted embeddings contain excessive non-speaker-related information, the decomposition process may degrade essential speech content in the SSL representations, resulting in suboptimal eta representations $\boldsymbol{\eta}$. In addition, speaker embeddings are often high-dimensional and may include redundancy. For this reason, it is also important to investigate whether dimensionality reduction techniques such as PCA can yield more compact and effective embeddings,

	WER	PER	SSIM	SPK ACC
Resemblyzer w/o PCA	4.94	6.01	89.02	74.01 ± 0.01
Resemblyzer w PCA-64	4.86	5.92	89.86	73.54 ± 0.02
Resemblyzer w PCA-128	4.48	5.84	90.59	65.87 ± 0.01
WavLM-SV w/o PCA	4.27	5.81	89.29	69.74 ± 0.02
WavLM-SV w PCA-64	4.15	5.75	89.35	68.31 ± 0.02
WavLM-SV w PCA-128	3.91	5.70	89.76	65.83 ± 0.01
ECAPA-TDNN w/o PCA	4.18	5.80	89.90	60.87 ± 0.01
ECAPA-TDNN w PCA-64	3.95	5.63	90.91	58.14 ± 0.02
ECAPA-TDNN w PCA-128	**3.81**	**5.63**	**92.46**	**55.73 ± 0.01**

Table 4.3: Results (%) measuring voice conversion intelligibility (W/PER ↓), target speaker similarity (SSIM ↑), and the speaker classification accuracy (SPK ACC ↓) using different speaker encoders and PCA reductions. The VC target speaker is LJSpeech.

thereby improving the overall performance of the approach. To this end, three different speaker encoders were evaluated:

- **Resemblyzer**[1], a publicly available implementation of [73].
- **WavLM-SV**[2], a WavLM-Large variant designed for speaker verification (SV).
- **ECAPA-TDNN** [154], already introduced in Section 4.3.

For each encoder, both the raw embeddings and two reduced versions obtained with PCA (PCA-64 and PCA-128) were considered. Following the same configurations described in Sections 4.3.1 and 4.3.2, two evaluations were carried out: a Speaker Classification task and a Voice Conversion task (considering only LJSpeech as the target speaker), using eta representations generated from the different embeddings $\mathbf{d}$ produced by each encoder. Table 4.3 reports WER/PER and SSIM of VC and the mean and standard deviation across the 5 folds of the cross-validation of the speaker accuracy for each model. ECAPA-TDNN consistently achieves the best performance across all metrics, demonstrating its superiority in preserving linguistic content, achieving a high level of target

[1] `https://github.com/CorentinJ/Real-Time-Voice-Cloning`
[2] `https://github.com/microsoft/UniSpeech/tree/main/downstreams/speaker_verification`

speaker similarity in the VC task, and reaching the lowest speaker classification accuracy. While WavLM-SV also shows strong intelligibility performance, its VC speaker similarity remains lower than that of ECAPA-TDNN. This highlights the fact that the eta representations created with **d** extracted using WavLM-SV are less speaker-independent than those of ECAPA-TDNN. This is further confirmed by the higher speaker accuracy in the classification task. On the other hand, the performance obtained with Resemblyzer is not comparable to that of the other two approaches, suggesting that the eta representations created with its **d** are too entangled. Interestingly, reducing the dimensionality of the speaker embeddings using PCA actually enhances overall performance in both the VC and speaker classification tasks for all methods. In this case as well, ECAPA-TDNN achieves the best values across all metrics, particularly with the PCA-128 configuration. This aligns with the hypothesis that reducing redundant information from **d** further improves performance. However, excessively reducing the dimensionality of **d** does not appear to provide additional benefits. This is evident from the performance obtained using PCA-64, which is lower than that of PCA-128. This suggests that while PCA can enhance performance, its effectiveness depends on the extent of the dimensionality reduction applied. In conclusion, the results demonstrate that ECAPA-TDNN is the most effective speaker encoder for the proposed approach, and applying PCA to **d** further enhances the decomposition process.

4.4 Conclusion and Limitations

In this work, a novel approach for disentangling speaker-related and speaker-independent components in WavLM representations was introduced, resulting in a new type of speaker-independent SSL representation called Eta-WavLM. By leveraging a simple linear decomposition strategy, the method effectively minimizes speaker information while preserving critical factors such as linguistic content, making it well-suited for speaker-independent speech processing tasks. This allowed the enhancement of the VC system described in Chapter 3, including in challenging scenarios such as cross-gender conver-

sion. The effectiveness of Eta-WavLM was first validated in a speaker-related classification task, confirming its ability to significantly reduce speaker-specific information. It was then further assessed in a content-related VC task, where it achieved a stronger balance between intelligibility, speaker similarity, and overall speech quality compared to existing disentanglement methods. Future research will explore extending this approach to multilingual settings, including low-resource languages, and investigating more advanced disentanglement strategies such as non-linear formulations. Another promising direction is the integration of Eta-WavLM representations into other downstream tasks, such as TTS, a line of research further explored in Chapter 5.

Despite these encouraging results, some limitations remain. The proposed method relies on explicit decomposition using speaker embeddings, which substantially reduces speaker identity leakage but does not fully eliminate it. In particular, performance in the 10-way speaker classification task remains above chance, indicating that traces of speaker information still persist. This residual information may reflect the dependence of the approach on the quality of the chosen speaker encoder. Future work could explore alternative or complementary speaker encoders to achieve a stronger trade-off between content preservation and the removal of speaker-related cues. Additionally, the latent basis $\mathbf{A}^*$ and bias $\mathbf{b}^*$ were derived using LibriSpeech. While LibriSpeech is a large and diverse dataset, incorporating larger-scale or multilingual datasets could further improve generalization across diverse linguistic and acoustic conditions, ultimately enhancing the robustness and flexibility of the proposed method.

Chapter 5

High-Quality Polyglot Text-to-Speech System

Summary

This chapter presents the final contribution of this research: the extension of the VC system described in the previous chapters into a complete polyglot TTS system. It outlines the modifications introduced to the VC architecture to enable text as input instead of speech, and describes the experimental setup and results used to evaluate whether the system's performance closely matches that of the original VC framework.

5.1 Introduction

The final objective of this research was to develop a high-quality polyglot TTS system capable of producing natural-sounding speech across multiple languages with native-level quality. Since the VC system developed in the earlier phases of this work achieved strong results not only in generating high-quality voices but also in producing native-like speech, the natural next step was to extend the original VC framework toward modern TTS synthesis. Inspired by recent TTS approaches that incorporate semantic features predicted from text within the synthesis pipeline [6; 7], a simple yet effective modification

of the VC system was introduced. In the VC framework, the content encoder takes audio as input and outputs semantic SSL (or eta) representations. To enable TTS synthesis within the same architecture, this module was redesigned to take *text features* as input while continuing to output the same type of semantic representations. This allowed the acoustic model and vocoder to remain unchanged, thereby reusing the robust components of the original VC system while adapting it for TTS. In this work, the chosen text features are *phoneme representations* of the input text.

The implemented TTS system belongs to the family of *modern text-to-speech approaches* that employ SSL features as intermediate representations to bridge the gap between text and speech (Section 2.3.2). While still in the early stages of research, preliminary experiments with internal data (Section 5.3) have shown promising results, indicating that the proposed system approaches the performance of the VC framework and has strong potential for advancing the in-house TTS, especially in terms of polyglot capabilities. The main contributions of this work are as follows:

- Introduction of an additional module to the VC pipeline, the TTS content encoder, which takes phonemes as input and generates semantic SSL (or eta) representations. This allowed the other components of the pipeline to remain unchanged.
- Development of a unified system that performs VC when the input is audio and TTS when the input is phonemes.
- Demonstration that the proposed system matches the performance of the VC framework, successfully generating high-quality polyglot voices.

5.2 Model Architecture

As shown in Figure 5.1, the proposed TTS system closely resembles the VC framework illustrated in Figure 3.1, with the same three main components and the same types of intermediate features. The key difference lies in the first module: instead of the VC content encoder (or eta content encoder), the system employs a new TTS content encoder

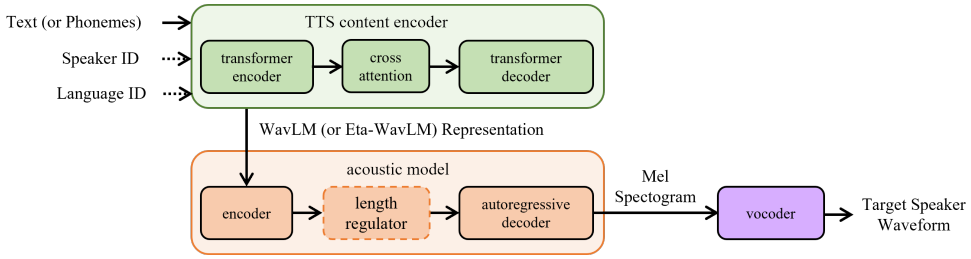


Figure 5.1: Overview of the proposed text-to-speech system.

that takes text as input (phonemes in this work) and generates SSL (or eta) representations. Although not explicitly shown in the figure, the VC content encoder remains part of the system in a secondary role, as it is used to generate the ground-truth representations that the TTS content encoder is trained to reproduce. For consistency with Chapter 3 and Chapter 4, WavLM is again adopted as the SSL model. Consequently, the semantic intermediate representations produced by the TTS content encoder can be either WavLM or Eta-WavLM representations, depending on the setup.

However, unlike VC where audio already contains the full range of information encoded in WavLM features, the mapping from phonemes to WavLM representations is more complex. WavLM features capture not only linguistic content but also speaker, language, and prosodic information. To support the TTS content encoder in learning this mapping, two additional (optional) inputs were introduced: a *speaker ID* and a *language ID*. These inputs serve distinct purposes. The speaker ID provides information about the speaker associated with the target WavLM representation. If the target features are standard WavLM representations, the speaker ID guides the TTS content encoder to generate WavLM features tied to the specified speaker. In this case, the system mirrors the VC framework: the acoustic model receives features with a chosen speaker identity and converts them into those of the target speaker, reproducing the task described in Chapter 3. If the target features are Eta-WavLM representations, which are speaker-independent, the speaker ID instead serves to guide prosodic aspects of the chosen speaker. The acoustic model then applies the target speaker identity on top of these features, as in Chapter 4. The language ID helps the model produce speech in

the correct language, particularly when trained on multilingual datasets. By explicitly providing language information, the system avoids having to infer it only from phoneme sequences, thereby improving accuracy and robustness in multilingual synthesis [187].

In this work, Eta-WavLM representations are employed, given the strong performance demonstrated by the VC system in Chapter 4. Consequently, the speaker ID serves primarily as a *prosody ID* rather than a true speaker identifier.

5.2.1 Problem Definition

Consider a multi-speaker dataset D containing N speakers, each with M utterances in the time domain. The j-th utterance of the i-th speaker is denoted as u_{ij}, its features extracted by the eta content encoder $\mathcal{C}$ (Figure 4.1) as s_{ij}, and the phonetic representation of its transcript as p_{ij}, where $i \in [1, N]$ and $j \in [1, M]$.

The TTS content encoder $\mathcal{T}$ takes as input the phoneme sequence p_{ij} along with two integer labels: the speaker ID sid_i and the language ID lid_{ij}. It then predicts the corresponding Eta-WavLM representation $\hat{s}_{ij}$:

$$\hat{s}_{ij} = \mathcal{T}(p_{ij}, sid_i, lid_{ij}; w_{\mathcal{T}}) \tag{5.1}$$

where $w_{\mathcal{T}}$ represents the TTS content encoder parameters. The predicted representation $\hat{s}_{ij}$ is then passed to the acoustic model $\mathcal{A}$ and the vocoder $\mathcal{V}$, which operate in sequence as described in Eq. 3.2 and Eq. 3.3, with the only difference that the input to $\mathcal{A}$ is now $\hat{s}_{ij}$ instead of the ground-truth s_{ij}.

As in the VC framework, the three components are trained independently. The TTS content encoder $\mathcal{T}$ is trained on a labeled multi-speaker dataset (potentially containing multiple languages), while the acoustic model $\mathcal{A}$ and the vocoder $\mathcal{V}$ remain target-speaker specific. After generating $(s_{ij}, p_{ij}, sid_i, lid_{ij})$ samples for each utterance u_{ij} in the dataset D, the TTS content encoder $\mathcal{T}$ is trained by optimizing the following objective function:

$$\min_{w_{\mathcal{T}}} L_{\mathcal{T}}(s_{ij}, \mathcal{T}(p_{ij}, sid_i, lid_{ij}; w_{\mathcal{T}}))) \tag{5.2}$$

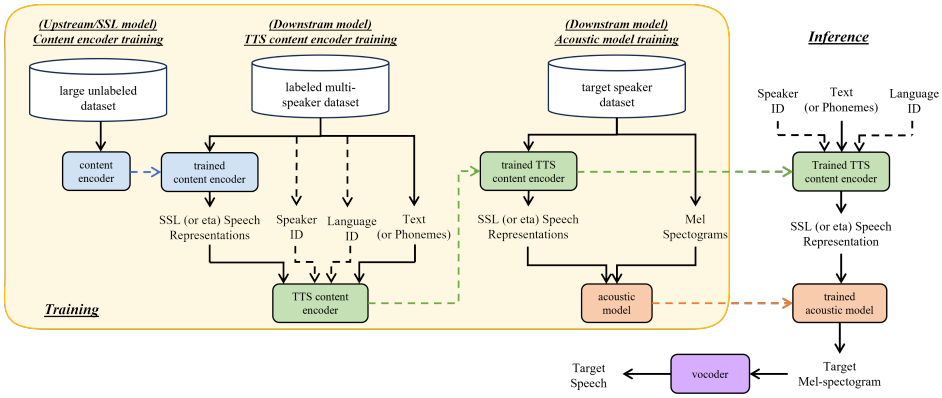


Figure 5.2: Training and inference procedures of the proposed text-to-speech approach. Vocoder training (not shown) is performed using the same target speaker dataset and mel-spectrograms generated for the acoustic model.

where $L_{\mathcal{T}}$ is the L_1 distance between the target and predicted Eta-WavLM representations. The acoustic model $\mathcal{A}$ and the vocoder $\mathcal{V}$ are then trained as described in Eq. 3.4 and Eq. 3.5, respectively.

Figure 5.2 illustrates how each component interacts with the others during the training and inference phases. The figure is largely similar to Figure 3.2 of the VC system, but with one key addition: the TTS content encoder. In this setup, the eta content encoder no longer provides input representations to the acoustic model but instead generates target representations for training the TTS content encoder. Specifically, for each tuple consisting of an Eta-WavLM representation, a phoneme sequence, a speaker ID, and a language ID, the TTS content encoder predicts the target Eta-WavLM representation, and the L_1 distance between the predicted and ground-truth representations is computed to update its parameters. Once trained, the TTS content encoder is used to produce the Eta-WavLM representations that serve as input for training the acoustic model, effectively replacing the role of the eta content encoder from the VC framework.

During inference, the three trained models are applied in sequence. First, the TTS content encoder generates an Eta-WavLM representation from a sequence of phonemes, a speaker ID, and a language ID. Next, the acoustic model predicts the target speaker's mel-spectrogram, and finally, the vocoder synthesizes the target-speaker waveform.

5.2.2 TTS Content Encoder

The TTS content encoder (green box in Figure 5.1) is based on the TransformerTTS architecture [140], but instead of generating mel-spectrograms, it outputs Eta-WavLM representations. The module consists of a Transformer encoder and an autoregressive Transformer decoder with cross-attention. Both the encoder and decoder are preceded by pre-nets, and the decoder is followed by two linear layers: one predicts the final Eta-WavLM representations, and the other predicts a *stop token*. The stop token allows the model to dynamically determine when to stop generating during inference, rather than always producing a fixed number of frames. Specifically, generation stops at the first frame where this probability exceeds a threshold of 0.5.

The input phoneme sequence and the language ID (if provided) are first embedded using two learnable embedding layers with dimensionalities of d_{model} and 16, respectively. The language embedding is then replicated along the frame axis to match the phoneme sequence length, and the two embeddings are concatenated along the feature axis. This is followed by a 3-layer CNN pre-net, where each convolutional layer has d_{model} channels, batch normalization [188], ReLU activation, and dropout. After the final ReLU activation, a linear projection layer with d_{model} units completes the pre-net. Sinusoidal positional encodings [97] are then added before passing the output through a stack of 6 Transformer encoder layers, each with 4 attention heads, forming the encoder. The encoder outputs features of dimensionality d_{model}.

After the encoder, the speaker ID is embedded using a 32-dimensional embedding layer, replicated along the frame axis, and concatenated with the encoder output along the feature axis. This is followed by a decoder pre-net composed of 3 stacked linear layers. Each linear layer (except the last) has 256 units, batch normalization, ReLU activation, and dropout. The final linear layer projects the 256-dimensional features to d_{model}, followed by batch normalization without ReLU activation. After adding sinusoidal positional encodings, the decoder consists of 6 Transformer decoder layers, each with 4 attention heads. Finally, two parallel linear layers produce the outputs: one with 1024 units for predicting the Eta-WavLM representations, and one with a single unit for

predicting the stop token.

5.3 Experiments

To evaluate the effectiveness of the proposed TTS approach, a set of comparative experiments was conducted against the VC system described in the previous chapters, which served as the baseline. All the experiments were run on a Linux machine with a single NVIDIA Titan RTX GPU with 24 GB of RAM.

Both the TTS and VC systems were tested using their standard training strategies, as described in Sections 5.2.1 and 3.2.1, as well as their cross-lingual fine-tuning variants, denoted as *TTS-F* and *VC-F*, since the TTS system naturally inherits this capability.

5.3.1 Experimental Setup

Experiments in both single- and multi-lingual settings were conducted using an internal dataset, denoted as *ENU-J*, as the target speaker for training the acoustic model. The ENU-J dataset consists of 6,260 studio-recorded audio clips from a single American English female speaker, with a total duration of about 10 hours.

For the TTS content encoder, training relied on a multi-speaker and multi-lingual dataset containing 16 internal speakers across 7 languages (French, Italian, Spanish, Mandarin Chinese, American English, British English, and German), with a total duration of about 260 hours. All audio samples in this dataset were first downsampled to 16 kHz, after which ground-truth Eta-WavLM representations were extracted following the procedure described in Section 4.2.3. The model dimension d_{model} in Section 5.2.2 was set to 512, following [140]. For training the acoustic model, each audio file was converted into its corresponding mel-spectrogram using the following setup: sampling_rate = 22050 Hz, n_fft = 1024, hop_length = 256, and n-mels = 80. This configuration follows the requirements of the internal version of HiFi-GAN [173] employed as vocoder. The main difference from the public version[1] is that this internal version was trained with

[1] `https://github.com/jik876/hifi-gan`

an additional speaker ID input to improve the synthesis of the target voice when used as a multi-speaker vocoder model.

Three languages were selected as the source audio for VC and their corresponding transcripts as input to the TTS system:

- **American English**: An internal female speaker dataset, denoted as *ENU-Z*, was selected, consisting of approximately 30,000 studio-recorded English audio clips with a total duration of about 30 hours.
- **European Italian**: An internal female speaker dataset, denoted as *ITI-E*, was selected, consisting of approximately 4,500 studio-recorded Italian audio clips with a total duration of about 15 hours.
- **European French**: An internal female speaker dataset, denoted as *FRF-A*, was selected, consisting of approximately 15,000 studio-recorded French audio clips with a total duration of about 23 hours.

For cross-lingual fine-tuning, the *single-speaker pre-training* strategy (Section 3.2.2) was adopted instead of the *multi-speaker pre-training* used in Chapter 3, as high-quality datasets with a sufficient number of hours were considered for this experiment. Specifically, three separate acoustic models (one for each language) were first pre-trained using the ENU-Z, ITI-E, and FRF-A datasets. Each pre-trained model was then fine-tuned on just 2 hours of ENU-J data for 15,000 steps with a batch size of 8.

5.3.2 Evaluation Metrics

Objective evaluations were conducted to assess the intelligibility and speaker similarity of the synthesized speech across all selected language scenarios. For each language, a dedicated test set consisting of <audio, transcript> pairs was created by randomly selecting 50 utterances from each dataset: ENU-Z for English, ITI-E for Italian, and FRF-A for French. All 50 utterances from each test set were then converted into ENU-J using the two VC baselines, resulting in a total of 450 utterances per language (150

Table 5.1: Objective evaluation. Results (%) in terms of intelligibility (W/PER ↓) and target (ENU-J) speaker similarity (SSIM ↑) for the proposed text-to-speech approaches and the voice conversion baselines.

	English (ENU-Z)			Italian (ITI-E)			French (FRF-A)		
	WER	**PER**	**SSIM**	**WER**	**PER**	**SSIM**	**WER**	**PER**	**SSIM**
Ground-truth	1.62	0.17	-	2.37	0.50	-	5.12	1.03	-
VC	1.86	0.18	94.84	4.98	1.23	87.18	9.22	2.67	86.75
VC-F	1.82	0.18	93.62	3.01	0.69	85.40	6.04	1.29	86.36
TTS	2.04	0.24	95.04	4.93	1.23	87.28	8.93	2.45	87.48
TTS-F	1.91	0.20	94.87	2.89	0.51	86.89	6.58	1.32	87.14

ground-truth + 300 converted). In parallel, the transcripts of these 50 utterances (150 transcripts in total across all languages) were synthesized into ENU-J using the two proposed TTS systems, producing a total of 300 generated utterances. The evaluation employed the same objective metrics used in Sections 3.3.2 and 4.3.2: word error rate (WER), phoneme error rate (PER), and speaker similarity (SSIM).

5.3.3 Results

Table 5.1 reports WER, PER, and SSIM for the proposed TTS systems and the VC baselines across the three selected languages: ENU-Z English (first column), ITI-E Italian (second column), and FRF-A French (third column). Across all languages, the TTS systems achieved intelligibility levels comparable to their VC counterparts. In English, all models performed close to the ground-truth, with very low WER and PER values. Notably, TTS and TTS-F obtained slightly higher SSIM scores than the VC systems, indicating a stronger resemblance to the target speaker. This difference can be attributed to the nature of the task: VC must transform source audio that contains the source speaker's identity, whereas TTS operates directly from phonemes. Although the use of Eta-WavLM features helps mitigate speaker leakage, VC remains inherently more challenging in this respect, especially since Eta-WavLM representations still retain some traces of speaker identity, as discussed in Section 4.4. In cross-lingual settings

(Italian and French), a similar trend emerged. Without cross-lingual fine-tuning, TTS achieved nearly identical results to VC, and the fine-tuned variants (TTS-F and VC-F) substantially improved intelligibility compared to their non-fine-tuned counterparts. In particular, TTS-F achieved the lowest WER and PER values in Italian and competitive results in French. SSIM remained consistently high across all configurations, with TTS systems again showing slightly higher values.

Overall, these results demonstrate that the proposed TTS system can match, and in some cases slightly surpass, the VC framework in both intelligibility and target-speaker similarity, confirming its effectiveness as an extension of the original VC system and its ability to support polyglot synthesis.

5.4 Conclusion and Limitations

In this chapter, a polyglot TTS system was introduced as the final outcome of this research. Building upon the A2O XVC framework detailed in the previous chapters, the proposed approach extends its capabilities from VC to full TTS synthesis. This was achieved by introducing a novel TTS content encoder that maps phoneme sequences, enriched with optional speaker and language identifiers, into semantic Eta-WavLM representations, while reusing the robust acoustic model and vocoder components of the original VC framework. Preliminary internal experiments showed that the proposed TTS system achieves performance closely matching that of the VC across multiple languages, in both standard and cross-lingual fine-tuning settings. The TTS models exhibited comparable intelligibility and target-speaker similarity, while maintaining the high-quality synthesis previously achieved by the VC system. These results confirm that leveraging SSL-based intermediate representations not only enables seamless integration of TTS and VC within a unified architecture but also allows the system to retain the polyglot and high-quality characteristics of its VC predecessor. Future research will extend the evaluation of the proposed TTS system using public datasets, additional languages (including low-resource ones), and complementary metrics such as MOS and Δ-ACMOS

(as in Section 3.3.2 of Chapter 3). Additionally, a promising direction is to enhance the system to achieve greater expressiveness and emotional variability in synthesized speech. The core idea is to evolve the TTS content encoder beyond continuous SSL representations. Inspired by recent advances in large language models (LLMs) in NLP and following the approach in [144], the TTS content encoder is envisioned to be redesigned to produce discrete semantic tokens instead of continuous features, leveraging the SOTA RepCodec model [67] for token generation. This design allows this component to be trained through a next-token prediction task with cross-entropy loss, in alignment with modern discrete generative modeling techniques.

Despite the encouraging results, some limitations remain. Although the approach achieved strong results in terms of intelligibility and speaker similarity, its prosody modeling still has room for improvement. This motivates the planned extension of the system toward more recent and expressive TTS techniques based on semantic discrete tokens rather than continuous ones. Additionally, as with the VC system, the current TTS system remains limited to a single target speaker because the acoustic model is speaker-specific. Future work will address this by exploring a multi-speaker variant of the acoustic model, to be developed alongside the planned improvements to the TTS content encoder, thereby jointly enhancing both prosody modeling and speaker generalization.

Chapter 6

Conclusion

This thesis presented a progressive research path aimed at developing high-quality artificial polyglot voices, addressing concrete industrial needs within the framework of a collaboration between the University of Turin and Cerence AI. The work advanced the SOTA in generative speech modeling by moving from VC to TTS synthesis through the introduction of innovative techniques for speaker-content disentanglement and cross-lingual adaptation.

The first major contribution was the design of a high-quality A2O XVC system. This framework leveraged SSL representations to capture linguistic and semantic content while employing an acoustic model and a vocoder to reconstruct target-speaker speech. To overcome the limitations of existing VC models, especially the degradation observed in cross-lingual and cross-gender settings, this thesis proposed two key solutions:

1. a cross-lingual fine-tuning strategy that effectively mitigates the native-language bias of target speakers, enabling them to speak unseen languages with native-like pronunciation, and

2. a novel speaker identity removal technique that transforms SSL features into speaker-independent representations, referred to as Eta-WavLM, allowing more accurate speaker identity conversion.

Together, these innovations allowed the VC system to achieve SOTA performance in both intra-lingual and cross-lingual conversions, producing highly natural and intelligible

speech with strong preservation of source prosody and emotion.

Building upon this success, the second major contribution of this thesis was the extension of the VC framework into a modern TTS system. This was achieved by introducing a novel TTS content encoder module that generates the same semantic representations used by the VC system directly from phonemes. This design enabled the reuse of the robust acoustic model and vocoder components from the VC framework, resulting in a unified architecture capable of performing both VC and TTS. Preliminary experiments showed that the proposed TTS system achieves intelligibility and target-speaker similarity on par with the VC framework, while retaining its ability to produce high-quality polyglot speech. These results confirm that SSL-based semantic intermediate representations can effectively bridge the gap between text and speech, enabling seamless integration of VC and TTS within a single unified architecture.

Appendix A

Mel-spectrogram

A mel-spectrogram visualizes the frequency content of an audio signal over time while incorporating principles of human auditory perception, known as *psychoacoustics*. This representation is widely used in speech processing and machine learning tasks. It results from analyzing a signal in the frequency domain, followed by a compression process aimed at reducing dimensionality. In this research, mel-spectrogram parameters have been used as the target spectral-domain representation for the acoustic models of both the VC and TTS systems developed. This appendix describes the steps required to extract these parameters from waveforms.

First, time-domain waveforms are transformed into their frequency-domain representation. This is achieved through the *Fourier transform*[1], or more specifically, the *discrete Fourier transform* (DFT), since sampled signals are used in practice. For example, if the *sampling rate* is 22.05 kHz, the analog waveform is sampled 22050 times per second at equally spaced intervals. Frequency analysis is typically performed using a *sliding window* of fixed duration (e.g., 50 ms), shifted by a predefined step (e.g., 10 ms). This means that every 10 ms, the DFT is computed over a 50 ms segment of the signal. Window functions are used to smooth the boundaries of each segment, reducing high-frequency artifacts caused by abrupt edges. Common examples include the *Hamming*

[1]Fourier transform: the Fourier transform decomposes a function (often a time-domain signal) into its constituent frequencies.

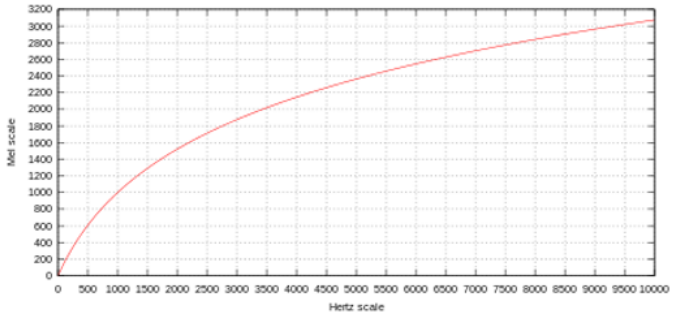


Figure A.1: Mel-scale frequency mapping.

and *Hanning* windows. Each windowed segment is referred to as a *frame* ($x[n]$):

$$x[n] = w[n]s[n] \tag{A.1}$$

where $w[n]$ is the window function and $s[n]$ is the original signal. The DFT is then calculated on each frame. For example, the DFT of frame $x[n]$ is given by:

$$X[k] = \sum_{n=0}^{N-1} x[n]e^{\left(-j\frac{2\pi}{N}kn\right)} \quad k = 0, \ldots, N-1 \tag{A.2}$$

Here, k denotes the frequency bin[1] and N is the window length in samples.

The next step is to produce a more compact representation of the DFT, a process known as *feature extraction*. This is achieved by applying a set of *triangular band-pass filters* that simulate human auditory perception. Humans perceive loudness differently across frequencies and have lower resolution at higher frequencies. This perceptual property is captured by the *Mel frequency scale*, which maps measured frequencies to perceived frequencies (see Figure A.1). For each frame, the feature extraction process consists of the following steps (Figure A.2):

1. Calculation of the square of the output of the DFT. This reflects the power of the speech at each frequency ($X[k]^2$), known as *DFT power spectrum*;

2. Application of the triangular Mel-scale filter banks to transform it to *mel-scale*

[1]Bin: term to indicate the frequency variable with discrete and non-continuous values.

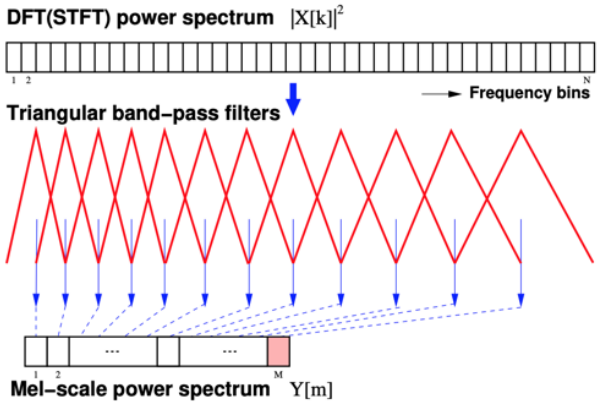


Figure A.2: Mel spectral coefficient extraction [189].

power spectrum Y. This is done by multiplying the DFT bins by the corresponding triangular filter functions and summing up values:

$$Y[m] = \sum_{k=1}^{N} W_m[k]|X[k]|^2 \tag{A.3}$$

where k is the DFT bin number $(1, \ldots, N)$, m is the mel-filter bank number $(1, \ldots, M)$, and $W[m]$ is the m-th triangular filter function.

Each $Y[m]$ thus represents the energy from a specific frequency band (or mel channel).

Appendix B

Publications and Patents

List of Publications:

- **Enhancing Polyglot Voices by Leveraging Cross-Lingual Fine-Tuning in Any-to-One Voice Conversion**[1]: The 2024 Conference on Empirical Methods in Natural Language Processing (EMNLP 2024).
- **Eta-WavLM: Efficient Speaker Identity Removal in Self-Supervised Speech Representations Using a Simple Linear Equation**[2]: The 63rd Annual Meeting of the Association for Computational Linguistics (ACL 2025).

List of Patents:

- **Cross-Lingual Any-to-One Voice Conversion**[3]: International patent application (WO2025184148), published in September 2025.
- **Generative Speech System having Speaker Independent Encoder for Generating Speaker-independent Continuous Semantic Vectors**: U.S. provisional patent application (25-0001-US-PRV), filed in February 2025.

[1] https://aclanthology.org/2024.findings-emnlp.122/
[2] https://aclanthology.org/2025.findings-acl.127/
[3] https://patentscope.wipo.int/search/en/WO2025184148

List of Figures

1.1 High-level overview of the voice conversion system. 3
1.2 Eta content encoder. 4
1.3 High-level overview of the text-to-speech system. 5

2.1 Self-supervised representation learning framework. Figure reproduced from [15] © 2022 IEEE. 8
2.2 Generative, contrastive and predictive SSL models. Figure reproduced from [15] © 2022 IEEE. 10
2.3 The Wav2Vec 2.0 SSL model. Figure reproduced from [44]. 13
2.4 The HuBERT SSL model. Figure reproduced from [54] © 2021 IEEE. . . 15
2.5 The WavLM SSL model. Figure reproduced from [56] © 2022 IEEE. . . . 16
2.6 The typical flow of a voice conversion system. Figure reproduced from [80] (CC-BY-4.0). 22
2.7 Encoder-decoder with attention for voice conversion. Figure reproduced from [80] (CC-BY-4.0). 24
2.8 The CycleGAN-VC architecture. Figure reproduced from [80] (CC-BY-4.0). 25
2.9 A typical auto-encoding network for voice conversion. Figure reproduced from [80] (CC-BY-4.0). 26
2.10 The three key components in a text-to-speech system. 28
2.11 Exaple of concatenative approach to synthesize the sentence: *my voice sounds natural*. 29

2.12 Mean opinion scores (MOS) for Concatenative, Parametric, and WaveNet TTS systems in US English and Mandarin Chinese, compared with human speech [133] 31

3.1 Overview of the proposed voice conversion system. 41
3.2 Training and inference procedures of the proposed cross-lingual voice conversion approach. Vocoder training (not shown) is performed using the same target speaker dataset and mel-spectrograms generated for the acoustic model. 43

4.1 High-level overview of the proposed Eta-WavLM approach. 58
4.2 UMAP projections of the WavLM (left) and Eta-WavLM (right) representations extracted from 10 utterances of 5 speakers (with ids 1995, 2830, 4992, 61, 6829) from the LibriSpeech test-clean set. 65
4.3 Overview of the proposed voice conversion system, where the standard content encoder is replaced by the eta content encoder. 66

5.1 Overview of the proposed text-to-speech system. 75
5.2 Training and inference procedures of the proposed text-to-speech approach. Vocoder training (not shown) is performed using the same target speaker dataset and mel-spectrograms generated for the acoustic model. . 77

A.1 Mel-scale frequency mapping. 87
A.2 Mel spectral coefficient extraction [189]. 88

List of Tables

3.1 Objective and subjective evaluation. (middle) Results (%) in terms of intelligibility (W/PER ↓) and speaker similarity (SSMI ↑) considering three speaker and content disentanglement strategies: discretization, perturbation and per-utterance standardization. (bottom) W/PER, SSIM, overall quality (MOS ↑) with 95% confidence intervals, and the delta of the Accent Comparative MOS (Δ-ACMOS) for the proposed approaches and the baseline systems. 51

3.2 Results (%) measuring intelligibility (W/PER ↓), speaker similarity (SSMI ↑), overall quality (MOS ↑) with 95% confidence intervals, and the delta of the Emotion Comparative Mean Opinion Score (Δ-ECMOS) vs neutral samples for the emotional experiment. 54

4.1 Classification accuracy results (%) for WavLM and Eta-WavLM across the 5 folds of cross-validation (ACC ↓). Lower accuracy indicates better performance, as it reflects reduced speaker-related information. 64

4.2 Objective and subjective evaluation of the voice conversion task. Results (%) in terms of intelligibility (W/PER ↓), target speaker similarity (SSIM ↑), and overall quality (MOS ↑) with 95% confidence intervals for the proposed Eta-WavLM and the baseline methods. WavLM was used as the SSL model across all approaches. 68

4.3 Results (%) measuring voice conversion intelligibility (W/PER ↓), target speaker similarity (SSIM ↑), and the speaker classification accuracy (SPK ACC ↓) using different speaker encoders and PCA reductions. The VC target speaker is LJSpeech. 70

5.1 Objective evaluation. Results (%) in terms of intelligibility (W/PER ↓) and target (ENU-J) speaker similarity (SSIM ↑) for the proposed text-to-speech approaches and the voice conversion baselines. 81

References

[1] B. van Niekerk, M.-A. Carbonneau, J. Zaïdi, M. Baas, H. Seuté, and H. Kamper, "A comparison of discrete and soft speech units for improved voice conversion," *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 6562–6566, 2021. 3, 17, 19, 27, 40, 47, 48, 52, 68

[2] W.-C. Huang, S. wen Yang, T. Hayashi, H. yi Lee, S. Watanabe, and T. Toda, "S3prl-vc: Open-source voice conversion framework with self-supervised speech representations," *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 6552–6556, 2021. 4, 40, 47

[3] A. Polyak, Y. Adi, J. Copet, E. Kharitonov, K. Lakhotia, W.-N. Hsu, A. rahman Mohamed, and E. Dupoux, "Speech resynthesis from discrete disentangled self-supervised representations," in *Interspeech*, 2021. 4, 17

[4] J. Li, W. Tu, and L. Xiao, "Freevc: Towards high-quality text-free one-shot voice conversion," *ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1–5, 2022. 4, 21, 27, 40, 42, 46

[5] M. Lajszczak, G. Cámbara, Y. Li, F. Beyhan, A. van Korlaar, F. Yang, A. Joly, Á. Martín-Cortinas, A. Abbas, A. Michalski, A. Moinet, S. Karlapati, E. Muszy'nska, H. Guo, B. Putrycz, S. L. Gambino, K. Yoo, E. Sokolova, and T. Drugman, "Base tts: Lessons from building a billion-parameter text-to-speech model on 100k hours of data," *ArXiv*, vol. abs/2402.08093, 2024. 4, 21, 34, 46

[6] S.-H. Lee, S.-B. Kim, J.-H. Lee, E. Song, M.-J. Hwang, and S.-W. Lee, "Hierspeech:

Bridging the gap between text and speech by hierarchical variational inference using self-supervised representations for speech synthesis," in *Advances in Neural Information Processing Systems* (S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, eds.), vol. 35, pp. 16624–16636, Curran Associates, Inc., 2022. 5, 34, 73

[7] E. Kharitonov, D. Vincent, Z. Borsos, R. Marinier, S. Girgin, O. Pietquin, M. Sharifi, M. Tagliasacchi, and N. Zeghidour, "Speak, read and prompt: High-fidelity text-to-speech with minimal supervision," *Transactions of the Association for Computational Linguistics*, vol. 11, pp. 1703–1718, 2023. 5, 34, 73

[8] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, pp. 436–44, 05 2015. 7

[9] G. Hinton, L. Deng, D. Yu, G. E. Dahl, A.-r. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. N. Sainath, and B. Kingsbury, "Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups," *IEEE Signal Processing Magazine*, vol. 29, no. 6, pp. 82–97, 2012. 7

[10] H. A. Bourlard and N. Morgan, *Connectionist Speech Recognition: A Hybrid Approach*. USA: Kluwer Academic Publishers, 1993. 7

[11] T. Kemp and A. Waibel, "Unsupervised training of a speech recognizer: Recent experiments," *Proc Eurospeech Budapest*, vol. 6, 02 1970. 8

[12] L. Lamel, J.-L. Gauvain, and G. Adda, "Lightly supervised and unsupervised acoustic model training," *Computer Speech & Language*, vol. 16, no. 1, pp. 115–129, 2002. 8

[13] J. Ma, S. Matsoukas, O. Kimball, and R. Schwartz, "Unsupervised training on large amounts of broadcast news data," in *2006 IEEE International Conference on Acoustics Speech and Signal Processing Proceedings*, vol. 3, pp. III–III, 2006. 8

[14] M. I. Jordan and T. M. Mitchell, "Machine learning: Trends, perspectives, and prospects," *Science*, vol. 349, no. 6245, pp. 255–260, 2015. 8

[15] A. rahman Mohamed, H. yi Lee, L. Borgholt, J. D. Havtorn, J. Edin, C. Igel, K. Kirchhoff, S.-W. Li, K. Livescu, L. Maaløe, T. N. Sainath, and S. Watanabe, "Self-supervised speech representation learning: A review," *IEEE Journal of Selected Topics in Signal Processing*, vol. 16, pp. 1179–1210, 2022. 8, 9, 10, 12, 13, 14, 17, 90

[16] L. Rabiner and J. Wilpon, "Considerations in applying clustering techniques to speaker independent word recognition," in *ICASSP '79. IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 4, pp. 578–581, 1979. 9

[17] J. Wilpon and L. Rabiner, "A modified k-means clustering algorithm for use in isolated work recognition," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 33, no. 3, pp. 587–594, 1985. 9

[18] J.-L. Gauvain and C.-H. Lee, "Maximum a posteriori estimation for multivariate gaussian mixture observations of markov chains," *IEEE Transactions on Speech and Audio Processing*, vol. 2, no. 2, pp. 291–298, 1994. 9

[19] L. Bahl, P. Brown, P. de Souza, and R. Mercer, "Maximum mutual information estimation of hidden markov model parameters for speech recognition," in *ICASSP '86. IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 11, pp. 49–52, 1986. 9

[20] S. Latif, R. K. Rana, S. Khalifa, R. Jurdak, J. Qadir, and B. W. Schuller, "Deep representation learning in speech processing: Challenges, recent advances, and future trends," *ArXiv*, vol. abs/2001.00378, 2020. 9

[21] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, no. 5786, pp. 504–507, 2006. 9

[22] M. Awad and R. Khanna, *Deep Neural Networks*, pp. 127–147. Berkeley, CA: Apress, 2015. 9

[23] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, pp. 1735–1780, 1997. 9, 24

[24] K. Cho, B. van Merriënboer, D. Bahdanau, and Y. Bengio, "On the properties of neural machine translation: Encoder–decoder approaches," in *Proceedings of SSST-8, Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation* (D. Wu, M. Carpuat, X. Carreras, and E. M. Vecchi, eds.), (Doha, Qatar), pp. 103–111, Association for Computational Linguistics, Oct. 2014. 9

[25] G. E. Hinton and R. S. Zemel, "Autoencoders, minimum description length and helmholtz free energy," in *Neural Information Processing Systems*, 1993. 10

[26] P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, and P.-A. Manzagol, "Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion," *J. Mach. Learn. Res.*, vol. 11, p. 3371–3408, Dec. 2010. 10

[27] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," *CoRR*, vol. abs/1312.6114, 2013. 10

[28] J. Chung, K. Kastner, L. Dinh, K. Goel, A. C. Courville, and Y. Bengio, "A recurrent latent variable model for sequential data," in *Neural Information Processing Systems*, 2015. 11, 26

[29] M. Fraccaro, S. K. Sønderby, U. Paquet, and O. Winther, "Sequential neural models with stochastic layers," in *Neural Information Processing Systems*, 2016. 11

[30] W.-N. Hsu, Y. Zhang, and J. R. Glass, "Learning latent representations for speech generation and transformation," in *Interspeech*, 2017. 11

[31] E. Aksan and O. Hilliges, "STCN: Stochastic temporal convolutional networks," in *International Conference on Learning Representations*, 2019. 11

[32] A. van den Oord, O. Vinyals, and K. Kavukcuoglu, "Neural discrete representation learning," in *Neural Information Processing Systems*, 2017. 11

[33] J. Chorowski, R. J. Weiss, S. Bengio, and A. van den Oord, "Unsupervised speech representation learning using wavenet autoencoders," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 27, pp. 2041–2053, 2019. 11

[34] W. Wang, Q. Tang, and K. Livescu, "Unsupervised pre-training of bidirectional speech encoders via masked reconstruction," *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 6889–6893, 2020. 11

[35] D. Jiang, X. Lei, W. Li, N. Luo, Y. Hu, W. Zou, and X. Li, "Improving transformer-based speech recognition using unsupervised pre-training," *ArXiv*, vol. abs/1910.09932, 2019. 11

[36] L. Liu and Y. Huang, "Masked pre-trained encoder base on joint ctc-transformer," *arXiv: Audio and Speech Processing*, 2020. 11

[37] X. Yue and H. Li, "Phonetically motivated self-supervised speech representation learning," in *Interspeech*, 2021. 11

[38] A. T. Liu, S.-W. Yang, P.-H. Chi, P.-C. Hsu, and H. yi Lee, "Mockingjay: Unsupervised speech representation learning with deep bidirectional transformer encoders," *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 6419–6423, 2019. 11

[39] A. T. Liu, S.-W. Li, and H. yi Lee, "Tera: Self-supervised learning of transformer encoder representation for speech," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 29, pp. 2351–2366, 2020. 11

[40] A. H. Liu, Y.-A. Chung, and J. R. Glass, "Non-autoregressive predictive coding for learning speech representations from local dependencies," in *Interspeech*, 2020. 11

[41] M. Schultz and T. Joachims, "Learning a distance metric from relative comparisons," in *Neural Information Processing Systems*, 2003. 12

[42] A. van den Oord, Y. Li, and O. Vinyals, "Representation learning with contrastive predictive coding," *ArXiv*, vol. abs/1807.03748, 2018. 12

[43] S. Schneider, A. Baevski, R. Collobert, and M. Auli, "wav2vec: Unsupervised pre-training for speech recognition," in *Interspeech*, 2019. 12

[44] A. Baevski, H. Zhou, A. Mohamed, and M. Auli, "wav2vec 2.0: a framework for self-supervised learning of speech representations," in *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS '20, (Red Hook, NY, USA), Curran Associates Inc., 2020. 12, 13, 90

[45] E. Jang, S. Gu, and B. Poole, "Categorical reparameterization with gumbel-softmax," in *International Conference on Learning Representations*, 2017. 13

[46] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: An asr corpus based on public domain audio books," in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 5206–5210, 2015. 13, 49, 62

[47] J. Kahn, M. Rivière, W. Zheng, E. Kharitonov, Q. Xu, P. E. Mazaré, J. Karadayi, V. Liptchinsky, R. Collobert, C. Fuegen, T. Likhomanenko, G. Synnaeve, A. Joulin, A. Mohamed, and E. Dupoux, "Libri-light: A benchmark for asr with limited or no supervision," in *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 7669–7673, 2020. `https://github.com/facebookresearch/libri-light`. 13

[48] A. Conneau, A. Baevski, R. Collobert, A. Mohamed, and M. Auli, "Unsupervised cross-lingual representation learning for speech recognition," in *Interspeech*, pp. 2426–2430, 08 2021. 13

[49] T. Arun Babu, C. Wang, A. Tjandra, K. Lakhotia, Q. Xu, N. Goyal, K. Singh, P. Platen, Y. Saraf, J. Pino, A. Baevski, A. Conneau, and M. Auli, "Xls-r: Self-supervised cross-lingual speech representation learning at scale," in *Interspeech*, pp. 2278–2282, 09 2022. 13

[50] V. Pratap, A. Tjandra, B. Shi, P. Tomasello, A. Babu, S. Kundu, A. Elkahky, Z. Ni, A. Vyas, M. Fazel-Zarandi, A. Baevski, Y. Adi, X. Zhang, W.-N. Hsu, A. Conneau, and M. Auli, "Scaling speech technology to 1,000+ languages," *J. Mach. Learn. Res.*, vol. 25, Jan. 2024. 13

[51] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)* (J. Burstein, C. Doran, and T. Solorio, eds.), (Minneapolis, Minnesota), pp. 4171–4186, Association for Computational Linguistics, June 2019. 14

[52] A. Baevski, M. Auli, and A. rahman Mohamed, "Effectiveness of self-supervised pre-training for speech recognition," *ArXiv*, vol. abs/1911.03912, 2019. 14

[53] A. Baevski, S. Schneider, and M. Auli, "vq-wav2vec: Self-supervised learning of discrete speech representations," in *International Conference on Learning Representations*, 2020. 14

[54] W.-N. Hsu, B. Bolte, Y.-H. H. Tsai, K. Lakhotia, R. Salakhutdinov, and A. rahman Mohamed, "Hubert: Self-supervised speech representation learning by masked prediction of hidden units," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 29, pp. 3451–3460, 2021. 14, 15, 90

[55] T. Fukada, K. Tokuda, T. Kobayashi, and S. Imai, "An adaptive algorithm for mel-cepstral analysis of speech," in *[Proceedings] ICASSP-92: 1992 IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 1, pp. 137–140 vol.1, 1992. 15, 28

[56] S. Chen, C. Wang, Z. Chen, Y. Wu, S. Liu, Z. Chen, J. Li, N. Kanda, T. Yoshioka, X. Xiao, J. Wu, L. Zhou, S. Ren, Y. Qian, Y. Qian, M. Zeng, and F. Wei, "Wavlm: Large-scale self-supervised pre-training for full stack speech processing," *IEEE Journal of Selected Topics in Signal Processing*, vol. 16, pp. 1505–1518, 2021. 15, 16, 46, 90

[57] Z. Chi, S. Huang, L. Dong, S. Ma, B. Zheng, S. Singhal, P. Bajaj, X. Song, X.-L. Mao, H. Huang, and F. Wei, "XLM-E: Cross-lingual language model pre-training via ELECTRA," in *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)* (S. Muresan, P. Nakov, and A. Villavicencio, eds.), (Dublin, Ireland), pp. 6170–6182, Association for Computational Linguistics, May 2022. 15

[58] W.-C. Huang, S.-W. Yang, T. Hayashi, and T. Toda, "A comparative study of self-supervised speech representation based voice conversion," *IEEE Journal of Selected Topics in Signal Processing*, vol. 16, pp. 1308–1318, 2022. 16, 27

[59] S. S. Hussain, P. Neekhara, J. Huang, J. Li, and B. Ginsburg, "Ace-vc: Adaptive and controllable voice conversion using explicitly disentangled self-supervised speech representations," *ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1–5, 2023. 16, 27

[60] K. Qian, Y. Zhang, H. Gao, J. Ni, C.-I. Lai, D. Cox, M. A. Hasegawa-Johnson, and S. Chang, "Contentvec: An improved self-supervised speech representation by disentangling speakers," in *International Conference on Machine Learning*, 2022. 16, 19, 20, 58

[61] G. Ruggiero, M. Testa, J. V. de Walle, and L. D. Caro, "Eta-wavlm: Efficient

speaker identity removal in self-supervised speech representations using a simple linear equation," *ArXiv*, vol. abs/2505.19273, 2025. 17

[62] W.-C. Huang, Y.-C. Wu, T. Hayashi, and T. Toda, "Any-to-one sequence-to-sequence voice conversion using self-supervised discrete speech representations," *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 5944–5948, 2020. 17

[63] B. van Niekerk, L. Nortje, M. Baas, and H. Kamper, "Analyzing speaker information in self-supervised models to improve zero-resource speech processing," in *Interspeech*, pp. 1554–1558, 08 2021. 17, 18, 52, 68

[64] A. Lee, H. Gong, P.-A. Duquenne, H. Schwenk, P.-J. Chen, C. Wang, S. Popuri, Y. Adi, J. Pino, J. Gu, and W.-N. Hsu, "Textless speech-to-speech translation on real data," in *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies* (M. Carpuat, M.-C. de Marneffe, and I. V. Meza Ruiz, eds.), (Seattle, United States), pp. 860–872, Association for Computational Linguistics, July 2022. 17

[65] Z. Borsos, R. Marinier, D. Vincent, E. Kharitonov, O. Pietquin, M. Sharifi, D. Roblek, O. Teboul, D. Grangier, M. Tagliasacchi, and N. Zeghidour, "Audiolm: A language modeling approach to audio generation," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 31, pp. 2523–2533, 2022. 17

[66] A. Radford, J. W. Kim, T. Xu, G. Brockman, C. McLeavey, and I. Sutskever, "Robust speech recognition via large-scale weak supervision," in *International Conference on Machine Learning*, 2022. 17, 36, 50

[67] Z. Huang, C. Meng, and T. Ko, "RepCodec: A speech representation codec for speech tokenization," in *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)* (L.-W. Ku, A. Martins, and V. Srikumar, eds.), (Bangkok, Thailand), pp. 5777–5790, Association for Computational Linguistics, Aug. 2024. 17, 66, 68, 83

[68] H.-S. Choi, J. Lee, W. S. Kim, J. H. Lee, H. Heo, and K. Lee, "Neural analysis and synthesis: Reconstructing speech from self-supervised representations," *ArXiv*, vol. abs/2110.14513, 2021. 18, 52, 68

[69] S.-H. Lee, H.-Y. Choi, S.-B. Kim, and S.-W. Lee, "Hierspeech++: Bridging the gap between semantic and acoustic representation of speech by hierarchical variational inference for zero-shot speech synthesis," *IEEE transactions on neural networks and learning systems*, vol. PP, 2023. 18, 34

[70] X. Zhu, Y. Lv, Y. Lei, T. Li, W. He, H. Zhou, H. Lu, and L. Xie, "Vec-tok speech: Speech vectorization and tokenization for neural speech generation," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 33, pp. 1243–1254, 2023. 19

[71] G. Ruggiero, M. Testa, J. V. de Walle, and L. D. Caro, "Enhancing polyglot voices by leveraging cross-lingual fine-tuning in any-to-one voice conversion," in *Conference on Empirical Methods in Natural Language Processing*, 2024. 19

[72] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," in *Proceedings of the 37th International Conference on Machine Learning*, ICML'20, JMLR.org, 2020. 20

[73] L. Wan, Q. Wang, A. Papir, and I. López-Moreno, "Generalized end-to-end loss for speaker verification," *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 4879–4883, 2017. 20, 70

[74] J.-h. Lin, Y. Lin, C.-M. Chien, and H.-y. Lee, "S2vc: A framework for any-to-any voice conversion with self-supervised pretrained representations," in *Interspeech*, pp. 836–840, 08 2021. 20

[75] D. Ulyanov, A. Vedaldi, and V. S. Lempitsky, "Instance normalization: The missing ingredient for fast stylization," *ArXiv*, vol. abs/1607.08022, 2016. 20

[76] J.-c. Chou and H.-y. Lee, "One-shot voice conversion by separating speaker and content representations with instance normalization," in *Interspeech*, pp. 664–668, 09 2019. 20, 47

[77] A. Tjandra, R. Pang, Y. Zhang, and S. Karita, "Unsupervised learning of disentangled speech content and style representation," in *Interspeech*, 2020. 21

[78] X. Zhu, Y. Lei, K. Song, Y. Zhang, T. Li, and L. Xie, "Multi-speaker expressive speech synthesis via multiple factors decoupling," *ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1–5, 2022. 21

[79] P. Cheng, W. Hao, S. Dai, J. Liu, Z. Gan, and L. Carin, "Club: A contrastive log-ratio upper bound of mutual information," in *International Conference on Machine Learning*, 2020. 21

[80] B. Sisman, J. Yamagishi, S. King, and H. Li, "An overview of voice conversion and its challenges: From statistical modeling to deep learning," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 29, pp. 132–157, 2020. 21, 22, 23, 24, 25, 26, 90

[81] M. Müller, "Dynamic time warping," *Information Retrieval for Music and Motion*, vol. 2, pp. 69–84, 01 2007. 23, 35

[82] Y. Stylianou, O. Cappe, and E. Moulines, "Continuous probabilistic transform for voice conversion," *IEEE Transactions on Speech and Audio Processing*, vol. 6, pp. 131–142, March 1998. 23

[83] T. Toda, A. W. Black, and K. Tokuda, "Spectral conversion based on maximum likelihood estimation considering global variance of converted parameter," in *Proceedings. (ICASSP '05). IEEE International Conference on Acoustics, Speech, and Signal Processing, 2005.*, vol. 1, pp. I/9–I12 Vol. 1, March 2005. 23

[84] T. Toda, H. Saruwatari, and K. Shikano, "Voice conversion algorithm based on gaussian mixture model with dynamic frequency warping of straight spectrum," in *2001 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings (Cat. No.01CH37221)*, vol. 2, pp. 841–844 vol.2, 2001. 24

[85] T. Toda, A. Black, and K. Tokuda, "Spectral conversion based on maximum likelihood estimation considering global variance of converted parameter," in *Proceedings. (ICASSP '05). IEEE International Conference on Acoustics, Speech, and Signal Processing, 2005.*, vol. 1, pp. I/9–I12 Vol. 1, 2005. 24

[86] T. Toda, A. W. Black, and K. Tokuda, "Voice conversion based on maximum-likelihood estimation of spectral parameter trajectory," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 8, pp. 2222–2235, 2007. 24

[87] A. Kain and M. Macon, "Spectral voice conversion for text-to-speech synthesis," in *Proceedings of the 1998 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP '98 (Cat. No.98CH36181)*, vol. 1, pp. 285–288 vol.1, 1998. 24

[88] S. H. Mohammadi and A. Kain, "Voice conversion using deep neural networks with speaker-independent pre-training," in *2014 IEEE Spoken Language Technology Workshop (SLT)*, pp. 19–23, 2014. 24

[89] F.-L. Xie, Y. Qian, Y. Fan, F. Soong, and H. Li, "Sequence error (se) minimization training of neural network for voice conversion," *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, pp. 2283–2287, 01 2014. 24

[90] L. Sun, S. Kang, K. Li, and H. Meng, "Voice conversion using deep bidirectional long short-term memory based recurrent neural networks," in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 4869–4873, 2015. 24

[91] H. Ming, D.-Y. Huang, L. Xie, J. Wu, M. Dong, and H. Li, "Deep bidirectional lstm modeling of timbre and prosody for emotional voice conversion," in *Interspeech*, pp. 2453–2457, 09 2016. 24

[92] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," *CoRR*, vol. abs/1409.0473, 2014. 24, 32

[93] J.-X. Zhang, Z. Ling, L.-J. Liu, Y. Jiang, and L. Dai, "Sequence-to-sequence acoustic modeling for voice conversion," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 27, pp. 631–644, 2018. 25

[94] K. Tanaka, H. Kameoka, T. Kaneko, and N. Hojo, "Atts2s-vc: Sequence-to-sequence voice conversion with attention and context preservation mechanisms," *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 6805–6809, 2018. 25

[95] H. Kameoka, K. Tanaka, D. Kwaśny, T. Kaneko, and N. Hojo, "Convs2s-vc: Fully convolutional sequence-to-sequence voice conversion," *IEEE/ACM Trans. Audio, Speech and Lang. Proc.*, vol. 28, p. 1849–1863, June 2020. 25

[96] W.-C. Huang, T. Hayashi, Y.-C. Wu, H. Kameoka, and T. Toda, "Voice transformer network: Sequence-to-sequence voice conversion using transformer with text-to-speech pretraining," in *Interspeech*, 2019. 25

[97] A. Vaswani, N. M. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *Neural Information Processing Systems*, 2017. 25, 32, 78

[98] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. C. Courville, and Y. Bengio, "Generative adversarial nets," in *Neural Information Processing Systems*, 2014. 25

[99] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with

conditional adversarial networks," *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5967–5976, 2016. 25

[100] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," *2017 IEEE International Conference on Computer Vision (ICCV)*, pp. 2242–2251, 2017. 25

[101] T. Kaneko and H. Kameoka, "Parallel-data-free voice conversion using cycle-consistent adversarial networks," *ArXiv*, vol. abs/1711.11293, 2017. 25

[102] Y. Taigman, A. Polyak, and L. Wolf, "Unsupervised cross-domain image generation," *ArXiv*, vol. abs/1611.02200, 2016. 25

[103] T. Kaneko, H. Kameoka, K. Tanaka, and N. Hojo, "Cyclegan-vc2: Improved cyclegan-based non-parallel voice conversion," *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 6820–6824, 2019. 25

[104] T. Kaneko, H. Kameoka, K. Tanaka, and N. Hojo, "Cyclegan-vc3: Examining and improving cyclegan-vcs for mel-spectrogram conversion," in *Interspeech*, 2020. 25

[105] H. Kameoka, T. Kaneko, K. Tanaka, and N. Hojo, "Stargan-vc: non-parallel many-to-many voice conversion using star generative adversarial networks," *2018 IEEE Spoken Language Technology Workshop (SLT)*, pp. 266–273, 2018. 26

[106] T. Kaneko, H. Kameoka, K. Tanaka, and N. Hojo, "Stargan-vc2: Rethinking conditional methods for stargan-based voice conversion," in *Interspeech*, pp. 679–683, 09 2019. 26

[107] T. Kaneko, H. Kameoka, K. Tanaka, and N. Hojo, "Maskcyclegan-vc: Learning non-parallel voice conversion with filling in frames," *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 5919–5923, 2021. 26

[108] C.-C. Hsu, H.-T. Hwang, Y.-C. Wu, Y. Tsao, and H. Wang, "Voice conversion from non-parallel corpora using variational auto-encoder," *2016 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA)*, pp. 1–6, 2016. 26

[109] N. Dehak, P. J. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, "Front-end factor analysis for speaker verification," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 4, pp. 788–798, 2011. 26

[110] Y. Saito, Y. Ijima, K. Nishida, and S. Takamichi, "Non-parallel voice conversion using variational autoencoders conditioned by phonetic posteriorgrams and d-vectors," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 5274–5278, 2018. 26

[111] W.-C. Huang, H. Luo, H.-T. Hwang, C.-C. Lo, Y.-H. Peng, Y. Tsao, and H.-M. Wang, "Unsupervised representation disentanglement using cross domain features and adversarial learning in variational autoencoder based voice conversion," *IEEE Transactions on Emerging Topics in Computational Intelligence*, vol. 4, pp. 468–479, 2020. 26

[112] M. Baas, B. van Niekerk, and H. Kamper, "Voice Conversion With Just Nearest Neighbors," in *Proc. INTERSPEECH 2023*, pp. 2053–2057, 2023. 27, 46, 48

[113] M. Abe, K. Shikano, and H. Kuwabara, "Cross-language voice conversion," *International Conference on Acoustics, Speech, and Signal Processing*, pp. 345–348 vol.1, 1990. 27

[114] X. Tan, T. Qin, F. K. Soong, and T.-Y. Liu, "A survey on neural speech synthesis," *ArXiv*, vol. abs/2106.15561, 2021. 28, 30, 32, 33

[115] R. Sproat and N. Jaitly, "Rnn approaches to text normalization: A challenge," *ArXiv*, vol. abs/1611.00068, 2016. 28

[116] N. Xue, "Chinese word segmentation as character tagging," in *International Journal of Computational Linguistics & Chinese Language Processing, Volume 8, Number 1, February 2003: Special Issue on Word Formation and Chinese Language Processing*, pp. 29–48, Feb. 2003. 28

[117] G. I. Schlünz, *The effects of part–of–speech tagging on text–to–speech synthesis for resource–scarce languages*. Phd thesis, North-West University, 2010. 28

[118] K. Yao and G. Zweig, "Sequence-to-sequence neural net models for grapheme-to-phoneme conversion," in *Interspeech*, 2015. 28

[119] K. Tokuda, T. Kobayashi, T. Masuko, and S. Imai, "Mel-generalized cepstral analysis - a unified approach to speech spectral estimation," in *ICSLP*, 1994. 28

[120] H. Kawahara, I. Masuda-Katsuse, and A. de Cheveigné, "Restructuring speech representations using a pitch-adaptive time–frequency smoothing and an instantaneous-frequency-based f0 extraction: Possible role of a repetitive structure in sounds1speech files available. see http://www.elsevier.nl/locate/specom1," *Speech Communication*, vol. 27, no. 3, pp. 187–207, 1999. 28

[121] J. Kim, J. Kong, and J. Son, "Conditional variational autoencoder with adversarial learning for end-to-end text-to-speech," *ArXiv*, vol. abs/2106.06103, 2021. 29

[122] Y. Ren, C. Hu, X. Tan, T. Qin, S. Zhao, Z. Zhao, and T.-Y. Liu, "Fastspeech 2: Fast and high-quality end-to-end text to speech," *ArXiv*, vol. abs/2006.04558, 2020. 29, 33

[123] M. Rashad, H. El-Bakry, I. il, and M. Mastor, "An overview of text-to-speech synthesis techniques," *International Conference on Communications and Information Technology - Proceedings*, 07 2010. 29

[124] K. Tokuda, Y. Nankaku, T. Toda, H. Zen, J. Yamagishi, and K. Oura, "Speech synthesis based on hidden markov models," *Proceedings of the IEEE*, vol. 101, pp. 1234–1252, May 2013. 30

[125] T. Yoshimura, K. Tokuda, T. Masuko, T. Kobayashi, and T. Kitamura, "Simultaneous modeling of spectrum, pitch and duration in hmm-based speech synthesis," in *EUROSPEECH*, 1999. 30

[126] K. Tokuda, T. Yoshimura, T. Masuko, T. Kobayashi, and T. Kitamura, "Speech parameter generation algorithms for hmm-based speech synthesis," *Acoustics, Speech, and Signal Processing, IEEE International Conference on*, vol. 3, pp. 1315–1318, 06 2000. 30

[127] S. Imai, "Cepstral analysis synthesis on the mel frequency scale," in *ICASSP '83. IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 8, pp. 93–96, April 1983. 30

[128] H. Zen, A. Senior, and M. Schuster, "Statistical parametric speech synthesis using deep neural networks," in *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 7962–7966, 2013. 30

[129] Y. Qian, Y. Fan, W. Hu, and F. K. Soong, "On the training aspects of deep neural network (dnn) for parametric tts synthesis," *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 3829–3833, 2014. 30

[130] K. Hashimoto, K. Oura, Y. Nankaku, and K. Tokuda, "The effect of neural networks in statistical parametric speech synthesis," in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 4455–4459, 2015. 30

[131] Y. Fan, Y. Qian, F.-L. Xie, and F. K. Soong, "Tts synthesis with bidirectional lstm based recurrent neural networks," in *INTERSPEECH*, 2014. 30

[132] W. Wang, S. Xu, and B. Xu, "First step towards end-to-end parametric tts synthesis: Generating spectral parameters with neural attention," in *INTERSPEECH*, pp. 2243–2247, 09 2016. 31

[133] A. van den Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. W. Senior, and K. Kavukcuoglu, "Wavenet: A generative model for raw audio," in *Speech Synthesis Workshop*, 2016. 31, 48, 91

[134] S. O. Arık, G. Diamos, A. Gibiansky, J. Miller, K. Peng, W. Ping, J. Raiman, and Y. Zhou, "Deep voice 2: multi-speaker neural text-to-speech," in *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, (Red Hook, NY, USA), p. 2966–2974, Curran Associates Inc., 2017. 31

[135] J. Shen, R. Pang, R. J. Weiss, M. Schuster, N. Jaitly, Z. Yang, Z. Chen, Y. Zhang, Y. Wang, R. J. Skerry-Ryan, R. A. Saurous, Y. Agiomyrgiannakis, and Y. Wu, "Natural tts synthesis by conditioning wavenet on mel spectrogram predictions," *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 4779–4783, 2017. 31, 32, 47

[136] W. Ping, K. Peng, A. Gibiansky, S. O. Arik, A. Kannan, S. Narang, J. Raiman, and J. Miller, "Deep voice 3: 2000-speaker neural text-to-speech," in *International Conference on Learning Representations*, 2018. 31

[137] A. Tamamori, T. Hayashi, K. Kobayashi, K. Takeda, and T. Toda, "Speaker-dependent wavenet vocoder," in *INTERSPEECH*, pp. 1118–1122, 08 2017. 31

[138] Y. Wang, R. J. Skerry-Ryan, D. Stanton, Y. Wu, R. J. Weiss, N. Jaitly, Z. Yang, Y. Xiao, Z. Chen, S. Bengio, Q. V. Le, Y. Agiomyrgiannakis, R. A. J. Clark, and R. A. Saurous, "Tacotron: Towards end-to-end speech synthesis," in *Interspeech*, 2017. 32

[139] D. Griffin and J. Lim, "Signal estimation from modified short-time fourier transform," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 32, no. 2, pp. 236–243, 1984. 32

[140] N. Li, S. Liu, Y. Liu, S. Zhao, M. Liu, and M. Zhou, "Close to human quality tts with transformer," *ArXiv*, vol. abs/1809.08895, 2018. 32, 78, 79

[141] Y. Ren, Y. Ruan, X. Tan, T. Qin, S. Zhao, Z. Zhao, and T.-Y. Liu, "Fastspeech: fast, robust and controllable text to speech," in *Proceedings of the 33rd International Conference on Neural Information Processing Systems*, (Red Hook, NY, USA), Curran Associates Inc., 2019. 33

[142] R. Huang, C. Zhang, Y. Wang, D. Yang, L. Liu, Z. Ye, Z. Jiang, C. Weng, Z. Zhao, and D. Yu, "Make-a-voice: Unified voice synthesis with discrete representation," *ArXiv*, vol. abs/2305.19269, 2023. 34

[143] P. Budzianowski, T. Sereda, T. Cichy, and I. Vuli'c, "Pheme: Efficient and conversational speech generation," *ArXiv*, vol. abs/2401.02839, 2024. 34

[144] X. Zhang, X. Zhang, K. Peng, Z. Tang, V. Manohar, Y. Liu, J. Hwang, D. Li, Y. Wang, J. Chan, Y. Huang, Z. Wu, and M. Ma, "Vevo: Controllable zero-shot voice imitation with self-supervised disentanglement," in *The Thirteenth International Conference on Learning Representations*, 2025. 34, 83

[145] N. Zeghidour, A. Luebs, A. Omran, J. Skoglund, and M. Tagliasacchi, "Soundstream: An end-to-end neural audio codec," *IEEE/ACM Trans. Audio, Speech and Lang. Proc.*, vol. 30, p. 495–507, Nov. 2021. 34

[146] A. Défossez, J. Copet, G. Synnaeve, and Y. Adi, "High fidelity neural audio compression," *Transactions on Machine Learning Research*, 2023. 34

[147] C. Wang, S. Chen, Y. Wu, Z.-H. Zhang, L. Zhou, S. Liu, Z. Chen, Y. Liu, H. Wang, J. Li, L. He, S. Zhao, and F. Wei, "Neural codec language models are zero-shot text to speech synthesizers," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 33, pp. 705–718, 2023. 34

[148] S. Chen, S. Liu, L. Zhou, Y. Liu, X. Tan, J. Li, S. Zhao, Y. Qian, and F. Wei, "Vall-e 2: Neural codec language models are human parity zero-shot text to speech synthesizers," *ArXiv*, vol. abs/2406.05370, 2024. 34

[149] R. Kubichek, "Mel-cepstral distance measure for objective speech quality assessment," in *Proceedings of IEEE Pacific Rim Conference on Communications Computers and Signal Processing*, vol. 1, pp. 125–128 vol.1, 1993. 35

[150] R. A. J. Clark and K. E. Dusterhoff, "Objective methods for evaluating synthetic intonation," in *Proceedings of the 6th European Conference on Speech Communication and Technology (Eurospeech 1999)*, 1999. 35

[151] D. J. Hermes, "Measuring the perceptual similarity of pitch contours," *Journal of Speech, Language, and Hearing Research*, vol. 41, no. 1, pp. 73–82, 1998. 35

[152] C. Jemine, "Master thesis: Automatic multispeaker voice cloning," 2019. Unpublished master's thesis, Université de Liège, Liège, Belgique. 36

[153] G. Ruggiero, E. Zovato, L. D. Caro, and V. Pollet, "Voice cloning: a multispeaker text-to-speech synthesis approach based on transfer learning," *ArXiv*, vol. abs/2102.05630, 2021. 36, 50

[154] B. Desplanques, J. Thienpondt, and K. Demuynck, "Ecapa-tdnn: Emphasized channel attention, propagation and aggregation in tdnn based speaker verification," in *Interspeech*, 2020. 36, 63, 70

[155] M. Bernard and H. Titeux, "Phonemizer: Text to phones transcription for multiple languages in python," *Journal of Open Source Software*, vol. 6, no. 68, p. 3958, 2021. 36

[156] T. Saeki, D. Xin, W. Nakata, T. Koriyama, S. Takamichi, and H. Saruwatari, "Utmos: Utokyo-sarulab system for voicemos challenge 2022," *ArXiv*, vol. abs/2204.02152, 2022. 37

[157] "Methods for subjective determination of transmission quality," recommendation p.800, International Telecommunication Union (ITU-T), Geneva, Switzerland, 1996. 37

[158] Y. Jia, Y. Zhang, R. J. Weiss, Q. Wang, J. Shen, F. Ren, Z. Chen, P. Nguyen, R. Pang, I. L. Moreno, and Y. Wu, "Transfer learning from speaker verification to multispeaker text-to-speech synthesis," in *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, NIPS'18, (Red Hook, NY, USA), p. 4485–4495, Curran Associates Inc., 2018. 37

[159] Y. Tang, C. Arnold, and T. J. Cox, "A study on the relationship between the intelligibility and quality of algorithmically-modified speech for normal hearing listeners," *Journal of Otorhinolaryngology, Hearing and Balance Medicine*, vol. 1, no. 1, 2018. 37

[160] R. Chung and B. Mak, "Synthesizing near native-accented speech for a non-native speaker by imitating the pronunciation and prosody of a native speaker," in *Interspeech*, pp. 4302–4306, 09 2022. 37

[161] D. Diatlova and V. Shutov, "Emospeech: Guiding fastspeech2 towards emotional text to speech," *ArXiv*, vol. abs/2307.00024, 2023. 37

[162] X. Tan, J. Chen, H. Liu, J. Cong, C. Zhang, Y. Liu, X. Wang, Y. Leng, Y. Yi, L. He, F. K. Soong, T. Qin, S. Zhao, and T.-Y. Liu, "Naturalspeech: End-to-end text-to-speech synthesis with human-level quality," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 46, pp. 4234–4245, 2022. 37

[163] A. Kain and M. Macon, "Spectral voice conversion for text-to-speech synthesis," in *Proceedings of the 1998 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP '98 (Cat. No.98CH36181)*, vol. 1, pp. 285–288 vol.1, 1998. 38

[164] Á. Martín-Cortinas, D. Sáez-Trigueros, I. Vall'es-P'erez, B. T. Vecino, P. Bilinski, M. Lajszczak, G. Beringer, R. Barra-Chicote, and J. Lorenzo-Trueba, "Enhancing the stability of llm-based speech generation systems through self-supervised representations," *ArXiv*, vol. abs/2402.03407, 2024. 46, 58

[165] J. Kahn, M. Rivière, W. Zheng, E. Kharitonov, Q. Xu, P.-E. Mazar'e, J. Karadayi, V. Liptchinsky, R. Collobert, C. Fuegen, T. Likhomanenko, G. Synnaeve, A. Joulin, A. rahman Mohamed, and E. Dupoux, "Libri-light: A benchmark for asr with limited or no supervision," *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 7669–7673, 2019. 46

[166] G. Chen, S. Chai, G.-B. Wang, J. Du, W.-Q. Zhang, C. Weng, D. Su, D. Povey, J. Trmal, J. Zhang, M. Jin, S. Khudanpur, S. Watanabe, S. Zhao, W. Zou, X. Li, X. Yao, Y. Wang, Z. You, and Z. Yan, "Gigaspeech: An evolving, multi-domain asr corpus with 10,000 hours of transcribed audio," in *Proc. Interspeech 2021*, 2021. 46

[167] C. Wang, M. Riviere, A. Lee, A. Wu, C. Talnikar, D. Haziza, M. Williamson, J. Pino, and E. Dupoux, "VoxPopuli: A large-scale multilingual speech corpus for representation learning, semi-supervised learning and interpretation," in *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)* (C. Zong, F. Xia, W. Li, and R. Navigli, eds.), (Online), pp. 993–1003, Association for Computational Linguistics, Aug. 2021. 46

[168] W.-N. Hsu, Y.-H. H. Tsai, B. Bolte, R. Salakhutdinov, and A. Mohamed, "Hubert: How much can a bad teacher benefit asr pre-training?," in *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 6533–6537, 2021. 46

[169] A. Baevski, W.-N. Hsu, A. Conneau, and M. Auli, "Unsupervised speech recognition," *Advances in Neural Information Processing Systems*, vol. 34, pp. 27826–27839, 2021. 46

[170] N. Srivastava, G. E. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov,

"Dropout: a simple way to prevent neural networks from overfitting," *J. Mach. Learn. Res.*, vol. 15, pp. 1929–1958, 2014. 47

[171] N. Kalchbrenner, E. Elsen, K. Simonyan, S. Noury, N. Casagrande, E. Lockhart, F. Stimberg, A. van den Oord, S. Dieleman, and K. Kavukcuoglu, "Efficient neural audio synthesis," in *International Conference on Machine Learning*, 2018. 48

[172] K. Kumar, R. Kumar, T. de Boissiere, L. Gestin, W. Z. Teoh, J. M. R. Sotelo, A. de Brébisson, Y. Bengio, and A. C. Courville, "Melgan: Generative adversarial networks for conditional waveform synthesis," in *Neural Information Processing Systems*, 2019. 48

[173] J. Kong, J. Kim, and J. Bae, "Hifi-gan: Generative adversarial networks for efficient and high fidelity speech synthesis," *ArXiv*, vol. abs/2010.05646, 2020. 48, 79

[174] S. gil Lee, W. Ping, B. Ginsburg, B. Catanzaro, and S. Yoon, "BigVGAN: A universal neural vocoder with large-scale training," in *The Eleventh International Conference on Learning Representations*, 2023. 48

[175] H. Siuzdak, "Vocos: Closing the gap between time-domain and fourier-based neural vocoders for high-quality audio synthesis," in *The Twelfth International Conference on Learning Representations*, 2024. 48, 66, 67

[176] K. Ito and L. Johnson, "The lj speech dataset." `https://keithito.com/LJ-Speech-Dataset/`, 2017. 48, 67

[177] V. Pratap, Q. Xu, A. Sriram, G. Synnaeve, and R. Collobert, "Mls: A large-scale multilingual dataset for speech research," in *Interspeech 2020, 21st Annual Conference of the International Speech Communication Association, Virtual Event, Shanghai, China, 25-29 October 2020* (H. Meng, B. X. 0011, and T. F. Zheng, eds.), pp. 2757–2761, ISCA, 2020. 49

[178] H. Bu, J. Du, X. Na, B. Wu, and H. Zheng, "Aishell-1: An open-source mandarin speech corpus and a speech recognition baseline," *2017 20th Conference of the Oriental Chapter of the International Coordinating Committee on Speech Databases and Speech I/O Systems and Assessment (O-COCOSDA)*, pp. 1–5, 2017. 49

[179] Z. Gao, Z. Li, J. Wang, H. Luo, X. Shi, M. Chen, Y. Li, L. Zuo, Z. Du, Z. Xiao, and S. Zhang, "Funasr: A fundamental end-to-end speech recognition toolkit," in *INTERSPEECH*, 2023. 50

[180] K. Zhou, B. Sisman, R. Liu, and H. Li, "Emotional voice conversion: Theory, databases and esd," *Speech Communication*, vol. 137, pp. 1–18, 2022. 53

[181] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," in *International Conference on Learning Representations*, 2019. 53

[182] A. Pasad, J.-C. Chou, and K. Livescu, "Layer-wise analysis of a self-supervised speech representation model," *2021 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pp. 914–921, 2021. 59

[183] M. Mohamed, O. Liu, H. Tang, and S. Goldwater, "Orthogonality and isotropy of speaker and phonetic information in self-supervised speech representations," in *Interspeech*, pp. 3625–3629, 09 2024. 59

[184] K. Ethayarajh, D. K. Duvenaud, and G. Hirst, "Towards understanding linear word analogies," in *Annual Meeting of the Association for Computational Linguistics*, 2018. 59

[185] M. Ravanelli, T. Parcollet, P. Plantinga, A. Rouhe, S. Cornell, L. Lugosch, C. Subakan, N. Dawalatabad, A. Heba, J. Zhong, J.-C. Chou, S.-L. Yeh, S.-W. Fu, C.-F. Liao, E. Rastorgueva, F. Grondin, W. Aris, H. Na, Y. Gao, R. D. Mori, and Y. Bengio, "SpeechBrain: A general-purpose speech toolkit," 2021. arXiv:2106.04624. 63

[186] L. McInnes, J. Healy, N. Saul, and L. Grossberger, "Umap: Uniform manifold ap-

proximation and projection," *The Journal of Open Source Software*, vol. 3, no. 29, p. 861, 2018. 64

[187] Z.-H. Zhang, L. Zhou, C. Wang, S. Chen, Y. Wu, S. Liu, Z. Chen, Y. Liu, H. Wang, J. Li, L. He, S. Zhao, and F. Wei, "Speak foreign languages with your own voice: Cross-lingual neural codec language modeling," *ArXiv*, vol. abs/2303.03926, 2023. 76

[188] S. Ioffe and C. Szegedy, "Batch normalization: accelerating deep network training by reducing internal covariate shift," in *Proceedings of the 32nd International Conference on International Conference on Machine Learning - Volume 37*, ICML'15, p. 448–456, JMLR.org, 2015. 78

[189] H. Hermansky, "Perceptual linear predictive (plp) analysis of speech," *The Journal of the Acoustical Society of America*, vol. 87, no. 4, pp. 1738–1752, 1990. 88, 91

[190] S. Young and P. Woodland, "State clustering in hidden markov model-based continuous speech recognition," *Computer Speech & Language*, vol. 8, no. 4, pp. 369–383, 1994.

[191] M. U. Gutmann and A. Hyvärinen, "Noise-contrastive estimation of unnormalized statistical models, with applications to natural image statistics," *J. Mach. Learn. Res.*, vol. 13, p. 307–361, Feb. 2012.

[192] B. A. Olshausen and D. J. Field, "Emergence of simple-cell receptive field properties by learning a sparse code for natural images," *Nature*, vol. 381, pp. 607–609, 1996.

[193] H. Lee, A. Battle, R. Raina, and A. Ng, "Efficient sparse coding algorithms," in *Neural Information Processing Systems*, 2006.

[194] G. S. V. S. Sivaram, S. K. Nemala, M. Elhilali, T. D. Tran, and H. Hermansky, "Sparse coding for speech recognition," *2010 IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 4346–4349, 2010.

[195] M. Ranzato, Y.-L. Boureau, S. Chopra, and Y. LeCun, "A unified energy-based framework for unsupervised learning," in *International Conference on Artificial Intelligence and Statistics*, 2007.

[196] G. E. Hinton, "Training products of experts by minimizing contrastive divergence," *Neural Computation*, vol. 14, pp. 1771–1800, 2002.

[197] S. won Park, D. young Kim, and M. chul Joe, "Cotatron: Transcription-guided speech encoder for any-to-many voice conversion without parallel data," in *Interspeech*, 2020.

[198] F. Biadsy, R. Weiss, P. Moreno, D. Kanevsky, and Y. Jia, "Parrotron: An end-to-end speech-to-speech conversion model and its applications to hearing-impaired speech and speech separation," in *Interspeech*, pp. 4115–4119, 09 2019.