



HAL
open science

Learning and controlling the source-filter representation of speech with a variational autoencoder

Samir Sadok, Simon Leglaive, Laurent Girin, Xavier Alameda-Pineda, Renaud Séguier

► To cite this version:

Samir Sadok, Simon Leglaive, Laurent Girin, Xavier Alameda-Pineda, Renaud Séguier. Learning and controlling the source-filter representation of speech with a variational autoencoder. *Speech Communication*, 2023, 148, pp.53-65. 10.1016/j.specom.2023.02.005 . hal-03650569v3

HAL Id: hal-03650569

<https://hal.science/hal-03650569v3>

Submitted on 21 Mar 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Learning and controlling the source-filter representation of speech with a variational autoencoder[☆]

Samir Sadok^{a,*}, Simon Leglaive^{a,*}, Laurent Girin^b, Xavier Alameda-Pineda^c, Renaud Séguier^a

^aCentraleSupélec, IETR UMR CNRS 6164, France

^bUniv. Grenoble Alpes, CNRS, Grenoble-INP, GIPSA-lab, France

^cInria, Univ. Grenoble Alpes, CNRS, LJK, France

Abstract

Understanding and controlling latent representations in deep generative models is a challenging yet important problem for analyzing, transforming and generating various types of data. In speech processing, inspired from the anatomical mechanisms of phonation, the source-filter model considers that speech signals are produced from a few independent and physically meaningful continuous latent factors, among which the fundamental frequency f_0 and the formants are of primary importance. In this work, we start from a variational autoencoder (VAE) trained in an unsupervised manner on a large dataset of unlabeled natural speech signals, and we show that the source-filter model of speech production naturally arises as orthogonal subspaces of the VAE latent space. Using only a few seconds of labeled speech signals generated with an artificial speech synthesizer, we propose a method to identify the latent subspaces encoding f_0 and the first three formant frequencies, we show that these subspaces are orthogonal, and based on this orthogonality, we develop a method to accurately and independently control the source-filter speech factors within the latent subspaces. Without requiring additional information such as text or human-labeled data, this results in a deep generative model of speech spectrograms that is conditioned on f_0 and the formant frequencies, and which is applied to the transformation speech signals. Finally, we also propose a robust f_0 estimation method that exploits the projection of a speech signal onto the learned latent subspace associated with f_0 .

Keywords: Representation learning, deep generative models, variational autoencoder, source-filter model

1. Introduction and related work

1.1. Introduction

High-dimensional data such as natural images or speech signals exhibit some form of regularity which prevents their dimensions from varying independently from each other. This suggests that there exists a latent representation of smaller dimension from which the high-dimensional observed data are generated. Discovering such latent representation of complex data is the goal of representation learning, and deep latent-variable generative models have emerged as promising unsupervised approaches (Goodfellow et al., 2014; Kingma & Welling, 2014; Rezende et al., 2014; Chen et al., 2016; Higgins et al., 2017; Kim & Mnih, 2018; Chen et al., 2018). The variational autoencoder (VAE) (Kingma & Welling, 2014; Rezende et al., 2014), which is equipped with both a generative and inference model, can be used not only for data generation but also for analysis and transformation. As an explicit model of a probability density function (pdf), the VAE can also be used as a learned prior for solving inverse problems such as compressed sensing (Bora

[☆]This research was supported by ANR-3IA MIAI (ANR-19-P3IA-0003), ANR-JCJC ML3RI (ANR-19-CE33-0008-01), H2020 SPRING (funded by EC under GA #871245).

*Corresponding authors

Email addresses: samir.sadok@centralesupelec.fr (Samir Sadok), simon.leglaive@centralesupelec.fr (Simon Leglaive)

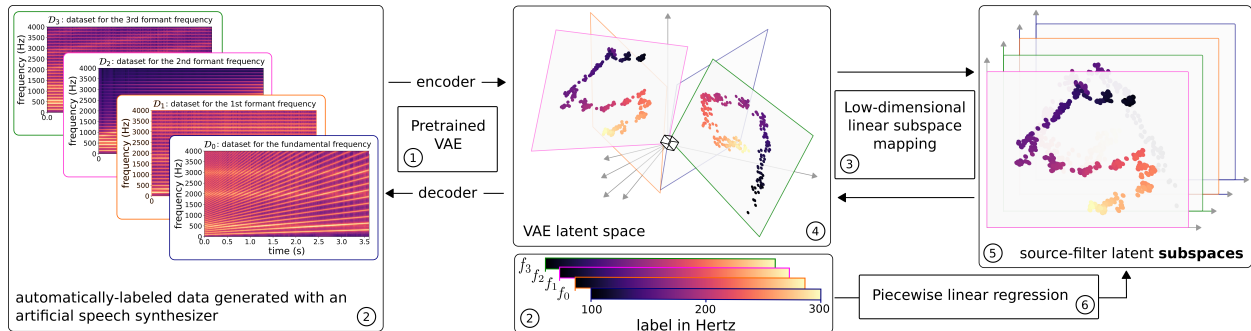


Figure 1: Overview of the proposed method. First, ① a VAE is trained in an unsupervised manner by maximizing a lower bound of the data log-marginal likelihood (see Section 2.1) on a large dataset of unlabeled natural speech signals (not shown on this figure for clarity). Given the pretrained VAE and given ② a few seconds of automatically-labeled speech generated with an artificial speech synthesizer, we then propose ③ a linear subspace identification method to put in evidence that ④ the VAE latent space is structured into ⑤ orthogonal subspaces that encode f_0 and the formant frequencies, thus complying with the source-filter model of speech production. The subspaces are identified by minimizing the L2 norm of the reconstruction error obtained after passing the artificially-generated speech trajectories through the VAE encoder and projecting on the subspaces (see Section 2.2). Finally, we propose ⑥ a piecewise linear regression model to learn how to move into the source-filter latent subspaces, so as to perform speech manipulations in a disentangled manner. This model is also learned using the automatically-labeled artificial speech trajectories, by minimizing the L2 norm of the difference between the output of the regression model and the data coordinates in the previously-learned latent subspaces (see Section 2.4). No supervision is used to constrain the structure of the VAE latent space during its training. Supervision is only used after the training of the VAE, to identify the disentangled latent subspaces encoding the f_0 and formant frequencies, and to learn how to move into these subspaces to perform speech manipulations.

et al., 2017), speech enhancement (Bando et al., 2018; Leglaive et al., 2018), or source separation (Kameoka et al., 2019; Jayaram & Thickstun, 2020). Making sense of the latent representation learned by a VAE and controlling the underlying continuous factors of variation in the data are important challenges to build more expressive and interpretable generative models and probabilistic priors.

A series of previous works on representation learning with deep generative models, in particular VAEs, have focused on images (Higgins et al., 2017; Kim & Mnih, 2018; Chen et al., 2018; Locatello et al., 2019, 2020a). Yet, it is not always easy to define the ground-truth latent factors of variation involved in the generation of natural images. For speech data, the latent factors of variation can be directly related to the anatomical mechanisms of speech production. Indeed, the source-filter model proposed by Fant (1970) considers that the production of speech signals results from the interaction of a source signal with a linear filter. In voiced speech, the source originates from the vibration of the vocal folds, which produces a quasi-periodic glottal sound wave whose fundamental frequency f_0 , loosely referred to as the *pitch*, is a key element of speech prosody. In unvoiced speech, a noise source is produced by a turbulent airflow or an acoustic impulse. The source signal is modified by the vocal tract, which is assumed to act as a linear filter. The cavities of the vocal tract give rise to resonances, which are called the *formants* and are characterized by their frequency, amplitude, and bandwidth. By moving the speech articulators such as the tongue, lips, and jaw, humans modify the shape of their vocal tract, which results in a change of the acoustic filter, the associated resonances, and the resulting speech sounds. For voiced phonemes, humans are able to control the formants independently of the pitch, i.e., to change the filter independently of the source (Fant, 1970), and of each other (MacDonald et al., 2011). The source-filter model thus considers that a speech signal is mainly characterized by a few continuous latent factors of variation corresponding to the source, among which f_0 plays a central role, and to the filter, mostly characterized by the formants. The independence of the source and filter characteristics makes the speech signals an interesting material for disentangled representation learning methods, especially with deep generative latent-variable models such as the VAE.

In this work, we analyze and control the latent space of a VAE from the perspective of the source-filter model of speech production, which can be beneficial for various applications in speech analysis, transformation, and synthesis. An overview of the proposed approach is shown in Figure 1. We first train a VAE on a dataset of about 25 hours of unlabeled speech signals. Then, using only a few seconds of automatically

labeled speech signals generated with an artificial speech synthesizer, we propose a method to identify and independently control the source-filter continuous latent factors of speech variation within the latent space of the VAE. Our contributions are the following: (i) We identify the source-filter model in the VAE latent space by showing experimentally that f_0 and the frequency of the first three formants, f_1 , f_2 , and f_3 , are encoded in different subspaces. We put in evidence the orthogonality of the learned subspaces, which not only shows that the representation learned by the VAE complies with the source-filter model of speech production, but also suggests that we can perform speech transformations in a disentangled manner (i.e., modifying one of the factors would not affect the others) by moving into the learned subspaces. (ii) For each factor f_i , $i \in \{0, 1, 2, 3\}$, we propose to learn a piecewise linear regression model from the factor value in the synthetic speech dataset to the coordinates in the corresponding latent subspace. This method allows us to precisely and independently control the source-filter factors of speech variation within the learned subspaces, as confirmed experimentally on both artificial and natural signals. Without requiring additional information such as text or human-labeled data, the proposed approach leads to a deep generative model of speech spectrograms that is conditioned on f_0 and the formant frequencies. (iii) Finally, to illustrate the interest of the learned representation for downstream tasks, we propose an f_0 estimation method that exploits the projection of a speech signal onto the learned latent subspace associated with f_0 . Experiments show that this approach competes with state-of-the-art methods in terms of precision and robustness to noise.

To the best of our knowledge, this is the first study showing the link between the classical source-filter model of speech production and the representation learned in the latent space of a VAE. Exploiting this link, we propose a principled method applied to the generation and transformation of speech signals controlled with interpretable trajectories of f_0 and the formant frequencies. Regarding this latter application, our objective is not to compete with traditional signal processing methods (these are discussed in the next subsection), which to the best of our knowledge remain the state-of-the-art. The interest of the present paper is rather to advance the understanding of deep generative modeling of speech signals while comparing fairly with traditional signal-model-based systems specifically designed for a given task. Moreover, advancing on the interpretability and control of the VAE latent space is expected to be beneficial for downstream tasks, for instance to develop pitch-informed extensions of VAE-based speech enhancement methods such as those of Bando et al. (2018); Leglaive et al. (2018, 2020); Bie et al. (2022).

1.2. Related work

Time-scale, pitch-scale, and timbre modification of speech signals is a highly covered research problem originally addressed with signal processing methods. Three main groups of approaches exist (Laroche, 2002): time-domain methods such as the pitch-synchronous overlap and add (PSOLA) algorithm (Moulines & Charpentier, 1990), methods that work in the short-time Fourier transform (STFT) domain such as the phase vocoder (Flanagan & Golden, 1966; Laroche & Dolson, 1999), and parametric approaches based for instance on linear predictive coding (LPC) (Makhoul, 1975; Markel & Gray, 1976), sinusoidal modeling (McAulay & Quatieri, 1986; George & Smith, 1997), or sinusoidal plus noise modeling (Serra & Smith, 1990; Laroche et al., 1993). Other signal-processing-based approaches to real-time speech manipulations include the STRAIGHT (Kawahara, 2006; Banno et al., 2007) and WORLD (Morise et al., 2016) vocoders, which exploit a decomposition of the speech signal into f_0 , spectral envelope, and aperiodicity.

Deep learning has recently emerged as a powerful approach to speech signal manipulation. A few methods have investigated combining traditional signal processing models with deep learning (Valin & Skoglund, 2019; Wang et al., 2019; Juvela et al., 2019; Lee et al., 2019; Choi et al., 2021). LPCNet is a successful neural vocoder inspired by the source-filter model (Valin & Skoglund, 2019). It was recently extended to pitch shifting and time stretching of speech signals by Morrison et al. (2021). Yet, the authors showed that time-domain PSOLA (TD-PSOLA) (Moulines & Charpentier, 1990) remains a very strong baseline that is difficult to outperform with deep learning methods.

Regarding the use of deep generative models (in particular VAEs) for speech modeling and transformation, the studies of Blaauw & Bonada (2016); Hsu et al. (2016, 2017b,a); Akuzawa et al. (2018) are pioneering. Of particular interest to the present paper is the work of Hsu et al. (2017a). The authors proposed to use VAEs for modifying the speaker identity and the phonemic content of speech signals by

translations in the latent space of a VAE. Yet, this method requires to know predefined values of the latent representations associated with both the source and target speech attributes to be modified. The performance of the method thus depends on the quality of the estimation of the source attribute (e.g., f_0), which has to be obtained from the input speech signal to be transformed. This differs from the proposed method which relies on projection onto the latent subspace associated with a given attribute, and only requires the target value for this attribute. Moreover, [Hsu et al. \(2017a\)](#) did not address the control of continuous factors of speech variation in the VAE latent space, contrary to the present work.

For deep latent representation learning methods, the challenge is to relate the learned representation to interpretable speech attributes. In [Qian et al. \(2020\)](#) and [Webber et al. \(2020\)](#), this interpretability is enforced by the design of the model. [Qian et al. \(2020\)](#) proposed to use three independent encoder networks to decompose a speech signal into f_0 , timbre and rhythm latent representations. [Webber et al. \(2020\)](#) focused on controlling source-filter parameters in speech signals, where the ability to control a given parameter (e.g., f_0) is enforced explicitly using labeled data and adversarial learning. In this approach, each parameter to be controlled requires dedicated training of the model. Moreover, these methods are speaker-dependent, as speech generation in [Qian et al. \(2020\)](#) is conditioned on the speaker identity and [Webber et al. \(2020\)](#) used a single-speaker training dataset. This contrasts with the proposed method which is speaker-independent, and in which the source-filter representation is shown to emerge as orthogonal subspaces of the latent space of a single unsupervised VAE model.

In the machine learning and computer vision communities, variants of the VAE have recently led to considerable progress in disentangled representation learning ([Kim & Mnih, 2018](#); [Higgins et al., 2017](#); [Chen et al., 2018](#)). From experimental analyses on image data, these methods suggest that a vanilla VAE cannot learn a disentangled representation. Moreover, [Locatello et al. \(2019, 2020a\)](#) recently showed both theoretically and from a large-scale experimental study that the unsupervised learning of disentangled representations is impossible without inductive biases (i.e., implicit or explicit assumptions by which a machine learning algorithm is able to generalize) on both the models and the data. Weakly-supervised ([Hosoya, 2018](#); [Shu et al., 2020](#); [Locatello et al., 2020b](#)) and semi-supervised ([Locatello et al., 2020c](#); [Sorrison et al., 2020](#)) methods have thus been proposed to learn disentangled representations. For example, the semi-supervised approach of [Locatello et al. \(2020c\)](#) exploits a small amount of labelled data to enforce the disentanglement of the representation at training time. This differs from the proposed approach where, after training a VAE on unlabeled natural speech signals, a few examples of artificially-generated labeled speech data are used to identify the disentangled structure of the VAE latent representation in terms of source-filter factors of speech variation. This allows us to experimentally show that learning a disentangled source-filter representation of speech using a simple VAE is possible, complying with the definition of disentanglement proposed in ([Higgins et al., 2018](#)).

Several methods have been recently proposed to control continuous factors of variation in deep generative models ([Jahanian et al., 2019](#); [Plumerault et al., 2020](#); [Goetschalckx et al., 2019](#); [Härkönen et al., 2020](#)), focusing essentially on generative adversarial networks ([Goodfellow et al., 2014](#)). They consist in identifying and then moving onto semantically meaningful directions in the latent space of the model. The present work is inspired by ([Plumerault et al., 2020](#)), which assumes that a factor of variation can be predicted from the projection of the latent vector along a specific axis, learned from artificially generated trajectories. The proposed method is however more generic, thanks to the learning of latent subspaces associated with the latent factors and to the introduction of a general formalism based on the use of “biased aggregated posteriors. Moreover, the previous works on controlling deep generative models only allow for moving “blindly onto semantically meaningful directions in the latent space. In the present study, we are able to generate data conditioned on a specific target value for a given factor of variation (e.g., a given formant frequency value). Finally, previous works focused on image data. To the best of our knowledge, the present paper proposes the first approach to identify and control source-filter factors of speech variation in a VAE.

The rest of this paper is organized as follows: Section 2 presents the proposed method for analyzing and controlling source-filter factors of speech variation in a VAE. The method is evaluated experimentally and compared with traditional signal processing algorithms and with the approach of [Hsu et al. \(2017a\)](#) in Section 3. We finally conclude in Section 4.

2. Analyzing and controlling source-filter factors of speech variation in a VAE

In this section, we first present the VAE model that we build upon, and that is trained on a large dataset of natural unlabeled speech signals. Then, we present (i) the method that we propose to identify latent subspaces encoding source-filter factors of speech variation from a few seconds of artificially-generated labeled speech signals, (ii) a simple strategy to measure the disentanglement of the learned representation, (iii) a method to control the continuous factors of variation in the learned subspaces and generate corresponding speech signals, and (iv) a simple method to estimate the f_0 contour of a speech signal by exploiting its projection onto the corresponding latent subspace.

2.1. Variational autoencoder

Generative modeling consists in learning a probabilistic model of an observable random variable $\mathbf{x} \in \mathcal{X} \subset \mathbb{R}^D$. Let $\mathcal{D} = \{\mathbf{x}_1, \dots, \mathbf{x}_N \in \mathcal{X}\}$ be a dataset of $N = \#\mathcal{D}$ independent and identically distributed (i.i.d.) observations of \mathbf{x} , where $\#\mathcal{D}$ denotes the cardinal of \mathcal{D} . The empirical distribution of \mathbf{x} is defined by $\hat{p}(\mathbf{x}) = \frac{1}{N} \sum_{\mathbf{x}_n \in \mathcal{D}} \delta(\mathbf{x} - \mathbf{x}_n)$, where δ is the Dirac delta function, which is null everywhere except in 0 where it takes the value 1.

The variational autoencoder (VAE) (Kingma & Welling, 2014; Rezende et al., 2014) attempts to approximate $\hat{p}(\mathbf{x})$ with a pdf $p_\theta(\mathbf{x})$ parametrized by θ . High-dimensional data such as natural images or speech signals exhibit some form of regularity which prevents the D dimensions of \mathbf{x} from varying independently from each other. We can thus assume that there exists a latent variable $\mathbf{z} \in \mathbb{R}^L$, with $L \ll D$, from which the observed data were generated. Accordingly, the model distribution in the VAE is defined by marginalizing the joint distribution of the latent and observed data, $p_\theta(\mathbf{x}) = \int p_\theta(\mathbf{x}|\mathbf{z})p(\mathbf{z})d\mathbf{z}$.

In this work, the observed data vector $\mathbf{x} \in \mathcal{X} = \mathbb{R}_+^D$ denotes the power spectrum of a short frame of speech signal (i.e., a column of the short-time Fourier transform (STFT) power spectrogram). Its entries are thus non-negative and its dimension D equals the number of frequency bins. We use the Itakura-Saito VAE (IS-VAE) (Bando et al., 2018; Leglaive et al., 2018; Girin et al., 2019) defined by

$$p(\mathbf{z}) = \mathcal{N}(\mathbf{z}; \mathbf{0}, \mathbf{I}), \quad (1)$$

$$p_\theta(\mathbf{x}|\mathbf{z}) = \prod_{d=1}^D \text{Exp}\left([\mathbf{x}]_d; [\mathbf{v}_\theta(\mathbf{z})]_d^{-1}\right), \quad (2)$$

where \mathcal{N} and Exp denote the densities of the multivariate Gaussian and univariate exponential distributions, respectively, and $[\mathbf{v}]_d$ denotes the d -th entry of \mathbf{v} . The inverse scale parameters of $p_\theta(\mathbf{x}|\mathbf{z})$ are provided by a neural network called the decoder, parametrized by θ and taking \mathbf{z} as input.

The marginal likelihood $p_\theta(\mathbf{x})$ and the posterior distribution $p_\theta(\mathbf{z}|\mathbf{x})$ are intractable due to the nonlinearities of the decoder, so it is necessary to introduce an inference model $q_\phi(\mathbf{z}|\mathbf{x}) \approx p_\theta(\mathbf{z}|\mathbf{x})$, which is defined by

$$q_\phi(\mathbf{z}|\mathbf{x}) = \mathcal{N}(\mathbf{z}; \boldsymbol{\mu}_\phi(\mathbf{x}), \text{diag}\{\mathbf{v}_\phi(\mathbf{x})\}), \quad (3)$$

where the mean and variance parameters are provided by a neural network called the encoder network, parametrized by ϕ and taking \mathbf{x} as input. Then, the VAE training consists in maximizing a lower-bound of $\ln p_\theta(\mathbf{x})$, called the evidence lower-bound (ELBO) and defined by

$$\mathcal{L}(\theta, \phi) = \mathbb{E}_{\hat{p}(\mathbf{x})} \left[\mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})} [p_\theta(\mathbf{x}|\mathbf{z})] - D_{\text{KL}}(q_\phi(\mathbf{z}|\mathbf{x}) \parallel p(\mathbf{z})) \right], \quad (4)$$

where $D_{\text{KL}}(q \parallel p) = \mathbb{E}_q [\ln q - \ln p]$ is the Kullback-Leibler divergence. During training, the generative and inference model parameters θ and ϕ are jointly estimated by maximizing the ELBO, using (variants of) stochastic gradient descent with the so-called reparameterization trick (Kingma & Welling, 2014; Rezende et al., 2014).

2.2. Learning source-filter latent subspaces

In addition to the pre-trained IS-VAE speech spectrogram model introduced in the previous subsection, we also assume the availability of an artificial speech synthesizer allowing for an accurate and independent control of f_0 and the formant frequencies. We use Soundgen (Anikin, 2019), a parametric synthesizer based on the source-filter model of speech production. For a given speech sound, the voiced component of the source signal is generated by a sum of sine waves, the noise component by a filtered white noise, and both components are then summed and passed through a linear filter simulating the effect of the human vocal tract. Importantly, this synthesizer allows us to easily generate artificial speech data labeled with f_0 and formant frequency values.

Formally, let f_i denote the speech factor of variation (in Hz) corresponding to the fundamental frequency, for $i = 0$, and to the formant frequencies, for $i \in \{1, 2, 3\}$. Let \mathcal{D}_i denote a dataset of artificially-generated speech vectors (more precisely short-term power spectra) synthesized by varying only f_i , all other factors $\{f_j, j \neq i\}$ being arbitrarily fixed. All examples in \mathcal{D}_i are labeled with the index and the value of the factor of variation. It would be relatively difficult to build such a dataset from existing corpora of unlabeled natural speech. In contrast, it is a very easy task using an artificial speech synthesizer such as Soundgen (Anikin, 2019), which precisely takes f_0 and the formant parameters as input, and outputs waveforms from which we extract power spectra.

Let $\hat{p}^{(i)}(\mathbf{x})$ denote the empirical distribution associated with \mathcal{D}_i , defined similarly as $\hat{p}(\mathbf{x})$. We also introduce the following marginal distribution over the latent vectors:

$$\hat{q}_\phi^{(i)}(\mathbf{z}) = \int q_\phi(\mathbf{z}|\mathbf{x})\hat{p}^{(i)}(\mathbf{x})d\mathbf{x} = \frac{1}{\#\mathcal{D}_i} \sum_{\mathbf{x}_n \in \mathcal{D}_i} q_\phi(\mathbf{z}|\mathbf{x}_n). \quad (5)$$

In the literature, this quantity is referred to as the aggregated posterior (Makhzani et al., 2016). However, $q_\phi(\mathbf{z}|\mathbf{x})$ is usually aggregated over the empirical distribution $\hat{p}(\mathbf{x})$ such that the aggregated posterior is expected to match with the prior $p(\mathbf{z})$ (Chen et al., 2018; Dai & Wipf, 2018). In contrast, in Equation (5) we aggregate over the “biased” data distribution $\hat{p}^{(i)}(\mathbf{x})$, where we know only one latent factor varies. This defines the explicit inductive bias (Locatello et al., 2019) that we exploit to learn the latent source-filter representation of speech in the VAE.

In the following of the paper, without loss of generality, we assume that, for each data vector in \mathcal{D}_i , the associated latent vector \mathbf{z} has been centered by subtracting the mean vector

$$\boldsymbol{\mu}_\phi(\mathcal{D}_i) = \mathbb{E}_{\hat{q}_\phi^{(i)}(\mathbf{z})}[\mathbf{z}] = \frac{1}{\#\mathcal{D}_i} \sum_{\mathbf{x}_n \in \mathcal{D}_i} \boldsymbol{\mu}_\phi(\mathbf{x}_n). \quad (6)$$

Because only one factor varies in \mathcal{D}_i , we expect latent vectors drawn from the “biased” aggregated posterior in Equation (5) to live on a low-dimensional manifold embedded in the original latent space \mathbb{R}^L . We assume this manifold to be a subspace characterized by its semi-orthogonal basis matrix $\mathbf{U}_i \in \mathbb{R}^{L \times M_i}$, $1 \leq M_i < L$. This matrix is computed by solving the following optimization problem:

$$\min_{\mathbf{U} \in \mathbb{R}^{L \times M_i}} \mathbb{E}_{\hat{q}_\phi^{(i)}(\mathbf{z})} \left[\|\mathbf{z} - \mathbf{U}\mathbf{U}^\top \mathbf{z}\|_2^2 \right], \quad s.t. \mathbf{U}^\top \mathbf{U} = \mathbf{I}. \quad (7)$$

The space spanned by the columns of \mathbf{U}_i is a subspace of the original latent space \mathbb{R}^L in which the latent vectors associated with the variation of the factor f_i in \mathcal{D}_i are expected to live. In Appendix A, we show that, similarly to the principal component analysis (PCA) (Pearson, 1901), the solution to the optimization problem (7) is given by the M_i eigenvectors corresponding to the M_i largest eigenvalues of

$$\mathbf{S}_\phi(\mathcal{D}_i) = \frac{1}{\#\mathcal{D}_i} \sum_{\mathbf{x}_n \in \mathcal{D}_i} \left[\boldsymbol{\mu}_\phi(\mathbf{x}_n)\boldsymbol{\mu}_\phi(\mathbf{x}_n)^\top + \text{diag}\{\mathbf{v}_\phi(\mathbf{x}_n)\} \right] - \boldsymbol{\mu}_\phi(\mathcal{D}_i)\boldsymbol{\mu}_\phi(\mathcal{D}_i)^\top. \quad (8)$$

The dimension M_i of the subspace can be chosen such as to retain a certain percentage of the data variance in the latent space. Note that the only source of supervision used here is the knowledge that only the factor f_i varies in the dataset \mathcal{D}_i .

2.3. Disentanglement analysis of the latent representation

As defined by Higgins et al. (2018), a representation is disentangled if it is possible to learn orthogonal latent subspaces associated with each factor of variation, whether they are single- or multi-dimensional. The approach presented in the previous subsection exactly follows this definition and offers a natural and straightforward way to objectively measure if the unsupervised VAE managed to learn a disentangled representation of the factors of variation under consideration. First, by simply looking at the eigenvalues associated with the columns of $\mathbf{U}_i \in \mathbb{R}^{L \times M_i}$, we can measure the amount of variance that is retained by the projection $\mathbf{U}_i \mathbf{U}_i^\top$. If a small number of components M_i represents most of the variance, it indicates that only a few intrinsic dimensions of the latent space are dedicated to the factor of variation f_i and varying this factor can be done by affine transformations. Second, if for two different factors of variation f_i and f_j , with $i \neq j$, the columns of \mathbf{U}_i are orthogonal to those of \mathbf{U}_j , this indicates that the two factors are encoded in orthogonal subspaces and therefore disentangled. It should however be verified experimentally that applying transformations by moving onto the subspace associated with f_i generalizes to values of $\{f_j, j \neq i\}$ different than the ones used in \mathcal{D}_i .

2.4. Controlling the source-filter factors of variation

So far, for each factor f_i , we have defined a methodology to learn a latent subspace $\mathbf{U}_i \in \mathbb{R}^{L \times M_i}$ that encodes its variations in the dataset \mathcal{D}_i , containing a few examples of speech data generated by an artificial synthesizer. Making now use of the values of the factor f_i for the data in \mathcal{D}_i , we learn a regression model $\mathbf{g}_{\eta_i} : \mathbb{R}_+ \mapsto \mathbb{R}^{M_i}$ from f_i , whose value is denoted by $y \in \mathbb{R}_+$, to the data coordinates in the latent subspace defined by \mathbf{U}_i . The parameters η_i are thus defined as the solution of the following optimization problem:

$$\min_{\eta} \left\{ \mathbb{E}_{\hat{q}_{\phi}^{(i)}(\mathbf{z}, y)} \left[\left\| \mathbf{g}_{\eta}(y) - \mathbf{U}_i^\top \mathbf{z} \right\|_2^2 \right] \stackrel{c}{=} \frac{1}{\#\mathcal{D}_i} \sum_{(\mathbf{x}_n, y_n) \in \mathcal{D}_i} \left\| \mathbf{g}_{\eta}(y_n) - \mathbf{U}_i^\top (\boldsymbol{\mu}_{\phi}(\mathbf{x}_n) - \boldsymbol{\mu}_{\phi}(\mathcal{D}_i)) \right\|_2^2 \right\}, \quad (9)$$

where $\hat{q}_{\phi}^{(i)}(\mathbf{z}, y) = \int q_{\phi}(\mathbf{z}|\mathbf{x}) \hat{p}^{(i)}(\mathbf{x}, y) d\mathbf{x}$, $\hat{p}^{(i)}(\mathbf{x}, y)$ is the empirical distribution associated with \mathcal{D}_i , considering now both the speech data vector \mathbf{x} and the value y of f_i , and $\stackrel{c}{=}$ denotes equality up to an additive constant w.r.t. η . This approach can be seen as a probabilistic extension of principal component regression (Hotelling, 1957; Kendall, 1957). For simplicity and because it revealed efficient for this task, \mathbf{g}_{η_i} is chosen as a piecewise linear regression model learned independently for each output coordinate $m \in \{1, \dots, M_i\}$. This choice is supported by the fact that the semi-orthogonal matrix \mathbf{U}_i decorrelates the data (Bengio et al., 2013). Solving the optimization problem (9) then consists in solving a linear system of equations. In this work, we used the Python library of Jekel & Venter (2019). The learning of the regression models is summarized in Algorithm 1 of Appendix B. It is important to remind that even if the regression model is supervised, the labeled dataset \mathcal{D}_i is very small with only a few seconds of speech signals (see experimental setup details in Appendix C), it is synthetic, and the values of f_i are automatically obtained during the generation of the data with an artificial speech synthesizer, so no manual annotation effort is required.

We can now transform a speech spectrogram by analyzing it with the VAE encoder, then linearly moving in the learned subspaces using the above regression model, and finally resynthesizing it with the VAE decoder. Given a source latent vector \mathbf{z} and a target value y for the factor f_i , we apply the following affine transformation:

$$\tilde{\mathbf{z}} = \mathbf{z} - \mathbf{U}_i \mathbf{U}_i^\top \mathbf{z} + \mathbf{U}_i \mathbf{g}_{\eta_i}(y). \quad (10)$$

This transformation consists in (i) subtracting the projection of \mathbf{z} onto the subspace associated with the factor of variation f_i ; and (ii) adding the target component provided by the regression model \mathbf{g}_{η_i} mapped from the learned subspace to the original latent space by the matrix \mathbf{U}_i . This operation allows us to move only in the latent subspace associated with the factor f_i . If this subspace is orthogonal to the latent subspaces associated with the other factors $\{f_j, j \neq i\}$, the latter should remain the same between \mathbf{z} and $\tilde{\mathbf{z}}$, only f_i should be modified. This process can be straightforwardly generalized to multiple factors, by subtracting and adding terms corresponding to each one of them. Contrary to Hsu et al. (2017a), the operation in Equation (10) does not require the knowledge of the factor f_i value associated with the source vector \mathbf{z} , it

only requires the value y of the factor f_i corresponding to the target vector $\tilde{\mathbf{z}}$ (this value y being used as input to the regression model.)

Finally, as the prior $p(\mathbf{z})$ and inference model $q_\phi(\mathbf{z}|\mathbf{x})$ are Gaussian (see Equations (2) and (3)), the transformation in Equation (10) has the following probabilistic formulation (using $\mathbf{U}_i^\top \mathbf{U}_i = \mathbf{I}$):

$$p(\tilde{\mathbf{z}}; f_i = y) = \mathcal{N}\left(\tilde{\mathbf{z}}; \mathbf{U}_i \mathbf{g}_{\eta_i}(y), \mathbf{M}_i\right) \quad (11)$$

$$q_\phi(\tilde{\mathbf{z}}|\mathbf{x}; f_i = y) = \mathcal{N}\left(\tilde{\mathbf{z}}; \mathbf{U}_i \mathbf{g}_{\eta_i}(y) + \mathbf{M}_i \boldsymbol{\mu}_\phi(\mathbf{x}), \mathbf{M}_i \text{diag}\{\mathbf{v}_\phi(\mathbf{x})\}\right), \quad (12)$$

where $\mathbf{M}_i = \mathbf{I} - \mathbf{U}_i \mathbf{U}_i^\top$. The prior in Equation (11) is now conditioned on the factor f_i and can be used to generate speech data given input trajectories of f_0 and formant frequencies. As we assumed centered latent data, the mean vector $\boldsymbol{\mu}_\phi(\mathcal{D}_i)$ defined in Equation (6) must be added to $\tilde{\mathbf{z}}$ before mapping this vector through the generative model $p_\theta(\mathbf{x}|\mathbf{z})$. The proposed method to manipulate the f_0 and formant frequencies in the latent source-filter subspaces is summarized in Algorithm 2 of Appendix B.

2.5. Estimating the fundamental frequency using the learned latent representation

To illustrate the interest of the learned representation on an analysis task, we propose to estimate the f_0 contour of a speech signal using its projection onto the corresponding latent subspace characterized by the estimated matrix $\mathbf{U}_0 \in \mathbb{R}^{L \times M_0}$ (cf. Section 2.2). As we focus on the analysis of f_0 , in this subsection we assume that the latent vectors are centered by subtracting the mean vector $\boldsymbol{\mu}_\phi(\mathcal{D}_0)$ defined in Equation (6): $\mathbf{z} \leftarrow \mathbf{z} - \boldsymbol{\mu}_\phi(\mathcal{D}_0)$. Let $\mathbf{p} = \mathbf{U}_0^\top \mathbf{z} \in \mathbb{R}^{M_0}$ denote the projection of \mathbf{z} onto the f_0 latent subspace.¹ Because \mathbf{p} results from a linear transformation of \mathbf{z} and the approximate posterior distribution $q_\phi(\mathbf{z}|\mathbf{x})$ defined in equation (3) is Gaussian, we have:

$$q_\phi(\mathbf{p}|\mathbf{x}) = \mathcal{N}\left(\mathbf{p}; \mathbf{U}_0^\top \boldsymbol{\mu}_\phi(\mathbf{x}), \mathbf{U}_0^\top \text{diag}\{\mathbf{v}_\phi(\mathbf{x})\} \mathbf{U}_0\right). \quad (13)$$

As will be confirmed experimentally in Section 3, the subspace generated by \mathbf{U}_0 encodes the fundamental frequency information and the formant frequencies are encoded in other orthogonal subspaces. Therefore, the projection of \mathbf{z} onto \mathbf{U}_0 is expected to provide invariance to a change of the formant frequencies, which is an appealing feature for estimating the fundamental frequency. The method we propose is simple but effective as will be shown experimentally. For an input speech power spectrum \mathbf{x}^{test} assumed to be voiced, the estimated fundamental frequency is given by the value y of f_0 associated with $\mathbf{x}^* \in \mathcal{D}_0$ defined by

$$\mathbf{x}^* = \arg \min_{\mathbf{x} \in \mathcal{D}_0} D_{\text{KL}}\left(q_\phi(\mathbf{p}|\mathbf{x}^{\text{test}}) \parallel q_\phi(\mathbf{p}|\mathbf{x})\right). \quad (14)$$

Using the KL divergence allows us to base the estimation of f_0 on the full distribution of the projection, i.e. taking not only the mean of the projection in Equation (13) into account, but also the covariance. The proposed method requires computing the KL divergence between two multivariate Gaussians, which admits a closed-form expression. We can thus simply compute the KL divergence in Equation (14) numerically for all the examples \mathbf{x} in the synthetic labeled dataset \mathcal{D}_0 and return the value y of f_0 associated with the minimum.

The f_0 estimation is done independently for each frame of the power spectrogram of an input speech signal. The resulting “raw” estimated f_0 trajectory is then smoothed by applying a median filter with a window size of 5 frames. Above, we assumed \mathbf{x}^{test} was a voiced speech spectrum. The voiced/unvoiced detection can be made automatically by setting a threshold on the minimum value of the above KL divergence, i.e. $D_{\text{KL}}(q_\phi(\mathbf{p}|\mathbf{x}^{\text{test}}) \parallel q_\phi(\mathbf{p}|\mathbf{x}^*))$.

¹The term projection used to refer to $\mathbf{p} \in \mathbb{R}^{M_0}$ is a misuse of language. Strictly speaking, the projection of $\mathbf{z} \in \mathbb{R}^L$ onto the subspace characterized by $\mathbf{U}_0 \in \mathbb{R}^{L \times M_0}$ is given by $\mathbf{U}_0 \mathbf{U}_0^\top \mathbf{z} \in \mathbb{R}^L$.

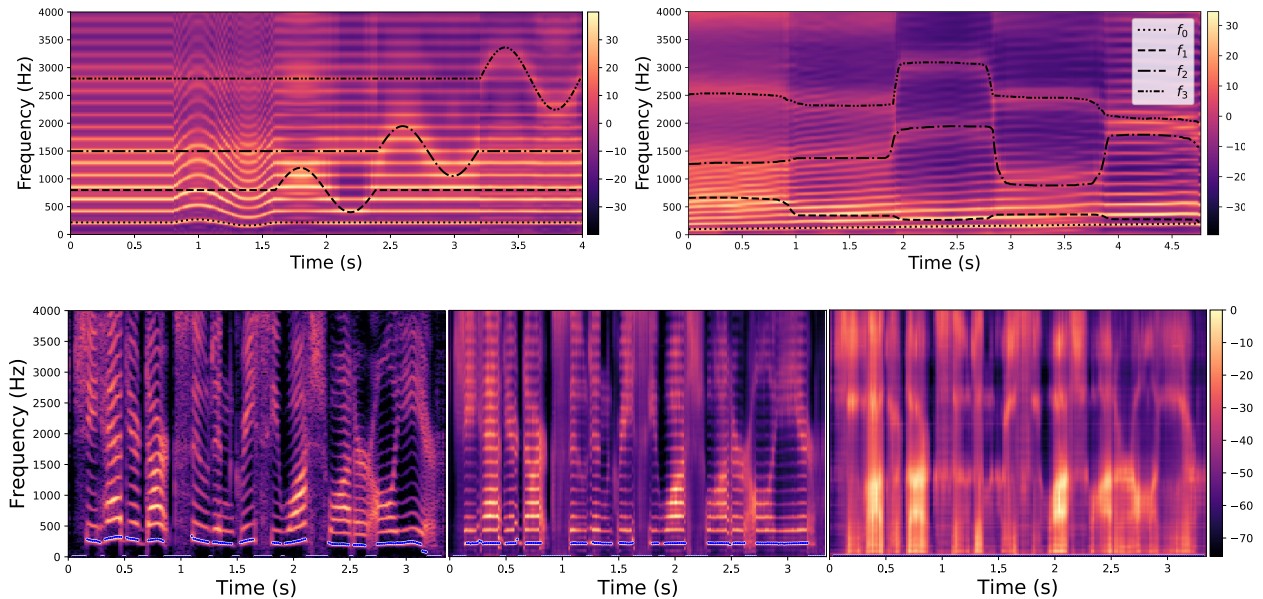


Figure 2: Examples of spectrograms modified and generated with the proposed method. The color bar indicates the power in dB. Top left: f_0 and formant transformations of a vowel /a/ uttered by a female speaker. Top right: Spectrogram generated from input trajectories of f_0 and formant frequencies. The target values of the factors f_i are indicated by the black lines. Bottom left: Original spectrogram of a speech signal uttered by a female speaker; Bottom middle: Transformed spectrogram with f_0 (blue line) set constant over time; Bottom right: Transformed spectrogram where the original voiced speech signal (bottom left) is converted into a whispered speech signal (i.e., the pitch is removed).

3. Experiments

This section presents qualitative and quantitative experimental results obtained with the proposed VAE-based method for controlling f_0 and the formant frequencies of speech signals. The VAE is trained on about 25 hours of multi-speaker speech data from the Wall Street Journal (WSJ0) dataset (Garofalo et al., 1993). The data space dimension is 513 and the latent space dimension is set to 16. This dimension was chosen based on previous work showing it is optimal for the modeling of speech power spectra in the context of speech enhancement (Leglaive et al., 2018, 2019b; Sekiguchi et al., 2019). For a given factor of variation, the corresponding latent subspace is learned (see Section 2.2) using short trajectories of speech power spectra (corresponding to a few seconds of speech) generated with Soundgen (Anikin, 2019), all other factors being arbitrarily fixed. When solving the optimization problem (7), the latent subspace dimension M_i of each factor of variation is chosen such that 80% of the data variance is retained. This leads $M_0 = 4$, $M_1 = 1$ and $M_2 = M_3 = 3$. The regression models used to control the speech factors of variation in the latent space (see Section 2.4) are learned on the same trajectories, but using the values of the Soundgen input control parameters (i.e., f_0 and formant frequencies values). More details on the experimental set-up can be found in Appendix C. Given a generated or transformed spectrogram, we use Waveglow (Prenger et al., 2019) to reconstruct the time-domain signal.

3.1. Qualitative results

In Figure 2, we illustrate the ability of the proposed method to modify f_0 and the formant frequencies in an accurate and independent manner. The top-left spectrogram contains five segments of equal length. The first segment corresponds to the original spectrogram of the steady vowel /a/ uttered by a female speaker. In the following segments, we vary successively each individual factor f_i , for $i = 0$ to 3, as indicated by the black lines in the figure. Variations of f_0 modify the harmonic structure of the signal while keeping the

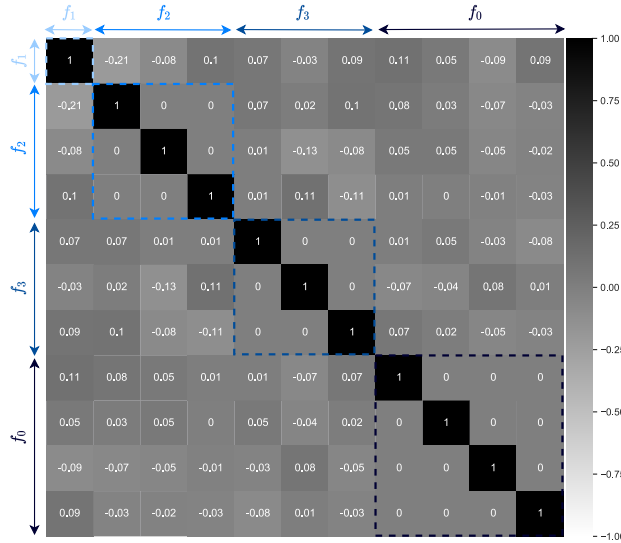


Figure 3: Correlation matrix of the learned latent subspaces basis vectors.

formant structure unaltered. Variations of f_i , $i \in \{1, 2, 3\}$, modify the formant frequencies, as indicated by the color map, while keeping f_0 unaltered.

The top-right spectrogram in Figure 2 was generated by using the conditional prior in Equation (11) (generalized to conditioning on multiple factors). We can see that the characteristics of the generated speech spectrogram match well with the input trajectories represented by the lines in the figure.

In the second row of Figure 2, from left to right we show the original spectrogram of a speech signal uttered by a female speaker (left), the transformed spectrogram where f_0 is set constant over time (middle), and the transformed spectrogram where the pitch has been removed (i.e., the original voiced speech signal is converted into a whispered speech signal) (right). This last spectrogram is obtained by subtracting to \mathbf{z} its projection onto the latent subspace corresponding to f_0 (i.e., by considering only the two first terms in the right-hand side of Equation (10)). This results in a spectrogram where the harmonic component is neutralized, while preserving the original formant structure. This is remarkable considering that the VAE was not trained on whispered speech signals, and it further confirms that the proposed method dissociates the source and the filter contributions in the VAE latent space.

Audio examples and additional examples of generated and transformed speech spectrograms can be found online.² In Appendix D, we provide plots of trajectories in the learned latent subspaces, illustrating that, according to each factor, the proximity of two speech spectra is preserved in the corresponding latent subspace.

3.2. Quantitative results

3.2.1. Orthogonality of the latent subspaces

In this experiment, we quantitatively evaluate the proposed method in terms of disentanglement of the learned source-filter latent representation. Following the discussion in Section 2.3, we compute the dot product between all pairs of unit vectors in the matrices $\mathbf{U}_i \in \mathbb{R}^{L \times M_i}$, $i \in \{0, 1, 2, 3\}$. Figure 3 shows that the resulting correlation matrix is mainly diagonal. Except for a correlation value of -0.21 across f_1 and the first component of f_2 , all other values are below 0.13 (in absolute value), confirming the orthogonality of the learned subspaces and thus the disentanglement of the learned source-filter representation of speech. The methodology presented in Section 2.2 was applied to try to identify orthogonal source-filter subspaces

² <https://samsad35.github.io/site-sfvae/>

Factor	Min (in Hz)	Max (in Hz)	Step size (in Hz)		Relative variation (%)
			English vowels	TIMIT	
f_0	100	300	1	10	± 50
f_1	300	900	10	50	± 50
f_2	1100	2700	20	100	± 42
f_3	2200	3200	20	50	± 18

Table 1: Variation range (min and max values) and step size used for the transformation of each test signal in the English vowels and TIMIT datasets, for each factor of variation f_i , $i \in \{0, 1, 2, 3\}$. The last column indicates by how much a factor varies relative to the center value of its variation range. Its entries are computed as $\pm (\max - \min)/(\max + \min) \times 100\%$.

directly in the space of speech power spectra and Mel-frequency cepstral coefficients (MFCCs). The results (see Appendix E) show that an organization into orthogonal source-filter subspaces does not exist when working on such raw signal representations. Indeed, there exist strong correlations between the learned subspaces. This confirms that the disentanglement in terms of source-filter factors is achieved during the unsupervised learning of the VAE model. We remind that in the proposed method, the synthetic labeled data are only used to identify the disentangled subspaces of the learned representation, after the VAE unsupervised training.

3.2.2. Pitch and formant transformations

In this experiment, we quantitatively evaluate the performance of the proposed method regarding the modification of f_0 and the formant frequencies in speech signals (see Section 2.4).

Experimental set-up. We use a corpus of 12 English vowels uttered by 50 male and 50 female speakers (Hillenbrand et al., 1995), which is labeled with the value of f_0 and the formant frequencies. We also use the TIMIT dataset (Garofolo et al., 1993), a corpus of phonemically and lexically transcribed speech of American English speakers of different sexes and dialects. We used the test corpus containing 1680 utterances. Because we are interested in studying the interaction between modifications of f_0 and the formant frequencies, we only evaluate the method on the voiced phonemes (40 phonemes over a total of 52), which are identified using the annotations. We transform each test signal in the English vowels and TIMIT datasets by varying one single factor f_i at a time, for $i \in \{0, 1, 2, 3\}$, according to the ranges and step sizes given in Table 1. For instance, when performing transformations of f_0 , for each test signal in the English vowels dataset, we vary linearly the target f_0 value between 100 and 300 Hz, with a step size of 1 Hz, thus resulting in 200 transformations.

Metrics. For the modification of each factor f_i , we measure the performance regarding three aspects: First, in terms of *accuracy* by comparing the target value for the factor (see Equation (10)) and its estimation computed from the modified output speech signal. Second, in terms of *disentanglement*, by comparing the values of f_j for $j \neq i$, before and after modification of the factor f_i . Third, in terms of speech *naturalness* of the transformed signal.

Accuracy and disentanglement are measured in terms of relative magnitude error (in percent, the lower the better). For a given factor f_i , it is defined by $\delta f_i = 100\% \times |\hat{y} - y|/y$ where y is the target value of f_i and \hat{y} its estimation from the output transformed signal. Let us take the example of a modification of f_0 : δf_0 measures the accuracy of the transformation on f_0 while δf_1 , δf_2 and δf_3 are used to assess if the other factors of variation f_1 , f_2 and f_3 remained unchanged after modifying f_0 . We use CREPE (Kim et al., 2018) to estimate f_0 and Parselmouth (Jadoul et al., 2018), which is based on PRAAT (Boersma & Weenink, 2021), to estimate the formant frequencies. Regarding speech naturalness, we use the NISQA objective measure (Mittag & Möller, 2020). This metric (the higher the better) was developed in the context of speech transformation algorithms and it was shown to highly correlate with subjective mean opinion scores (MOS) (i.e., human ratings). As a reference, the NISQA score on the original dataset of English vowels (i.e., without any processing) is equal to 2.60 ± 0.53 .

Factor	Method	English vowels dataset					TIMIT dataset				
		NISQA (†)	δf_0 (% , ↓)	δf_1 (% , ↓)	δf_2 (% , ↓)	δf_3 (% , ↓)	NISQA (†)	δf_0 (% , ↓)	δf_1 (% , ↓)	δf_2 (% , ↓)	δf_3 (% , ↓)
f_0	TD-PSOLA	2.32 ± 0.55	3.8 ± 2.5	6.3 ± 2.8	3.7 ± 0.9	2.1 ± 0.5	2.36 ± 0.50	2.4 ± 1.9	7.9 ± 0.6	4.5 ± 0.3	3.9 ± 0.2
	WORLD	2.49 ± 0.60	4.5 ± 0.6	3.7 ± 1.8	2.3 ± 0.7	1.2 ± 0.2	2.45 ± 0.47	0.3 ± 0.1	7.1 ± 1.2	6.2 ± 0.4	4.2 ± 0.2
	VAE baseline	1.94 ± 0.43	6.21 ± 2.8	10.4 ± 2.4	6.2 ± 0.9	4.5 ± 0.2	1.59 ± 0.43	16.1 ± 6.3	17.0 ± 3.0	12.1 ± 0.2	10.9 ± 1.3
	Proposed	2.08 ± 0.48	0.8 ± 0.2	7.2 ± 1.3	3.6 ± 1.2	3.8 ± 0.3	2.28 ± 0.57	0.8 ± 0.6	9.1 ± 1.1	8.3 ± 0.9	6.0 ± 1.8
f_1	VAE baseline	1.84 ± 0.5	11.3 ± 4.2	15.1 ± 3.5	6.0 ± 1.2	4.2 ± 0.4	1.42 ± 0.34	10.1 ± 2.8	16.4 ± 1.4	12.4 ± 0.9	11.2 ± 2.6
	Proposed	1.85 ± 0.4	6.0 ± 1.6	8.4 ± 3.2	5.7 ± 0.4	4.4 ± 0.3	1.66 ± 0.31	7.1 ± 3.6	9.2 ± 0.8	9.0 ± 1.3	7.8 ± 1.1
f_2	VAE baseline	2.01 ± 0.4	19.5 ± 3.2	10.7 ± 0.5	10.9 ± 1.9	5.8 ± 0.6	1.46 ± 0.30	19.3 ± 5.0	16.4 ± 0.8	20.3 ± 6.3	11.5 ± 0.5
	Proposed	2.03 ± 0.43	8.5 ± 1.1	8.7 ± 1.1	6.2 ± 1.5	5.8 ± 0.2	1.49 ± 0.30	9.1 ± 2.2	8.3 ± 1.3	4.3 ± 1.3	8.1 ± 0.2
f_3	VAE baseline	1.82 ± 0.14	27.0 ± 1.5	13.0 ± 1.3	12.0 ± 1.8	7.3 ± 1.5	1.40 ± 0.48	20.4 ± 1.0	17.4 ± 0.2	14.4 ± 0.2	11.7 ± 2.3
	Proposed	1.94 ± 0.48	8.3 ± 1.0	8.6 ± 0.7	4.9 ± 0.9	2.0 ± 0.4	1.48 ± 0.42	8.5 ± 1.9	8.7 ± 0.9	5.7 ± 2.1	2.5 ± 1.8

Table 2: Performance (mean and standard deviation) for the transformation of f_0 and the formant frequencies (f_1 , f_2 and f_3) on the English vowel and TIMIT datasets.

TIMIT is phonemically richer than the English vowels dataset, however, it is not labeled with f_0 and formant frequency values. Therefore, we do not have the ground truth values which makes the evaluation in terms of disentanglement more difficult than with the English vowels labeled dataset. Instead of the ground truth, we use the formant frequencies and f_0 values computed on the original speech utterances (i.e., before transformation). This makes the evaluation on TIMIT less reliable than on the English vowels dataset, but it allows us to test the methods on a larger variety of phonemes.

Methods. We compare the proposed approach with several methods from the literature: (i) TD-PSOLA (Moulines & Charpentier, 1990) performs f_0 modification through the decomposition of the signal into pitch-synchronized overlapping frames. (ii) WORLD (Morise et al., 2016) is a vocoder also used for f_0 modification. It decomposes the speech signal into three components characterizing f_0 , the aperiodicity, and the spectral envelope. (iii) The method proposed by Hsu et al. (2017a) (here referred to as “VAE baseline”) consists in applying translations directly in the latent space of the VAE. Unlike the proposed approach, this method requires predefined latent attribute representations μ_{src} and μ_{trgt} associated with the source and target values of the factor to be modified, respectively. In particular, computing μ_{src} requires analyzing the input speech signal, for instance, to estimate f_0 , which is not the case for the proposed method. The source and target latent attribute representations are then used to perform the translation $\tilde{\mathbf{z}} = \mathbf{z} - \mu_{\text{src}} + \mu_{\text{trgt}}$, where \mathbf{z} and $\tilde{\mathbf{z}}$ are respectively the original and modified latent vectors. To ensure a fair comparison, we build dictionaries of predefined latent attribute representations using the same artificially-generated speech data that were used in the training stage of the proposed method. All the methods we compare with require a pre-processing of the input speech signal to compute the input trajectory of the factor to be modified, which is not the case of the proposed method.

Discussion. Experimental results (mean and standard deviation) are shown in Table 2. Compared to the VAE baseline, the proposed method obtains better performance in terms of accuracy, disentanglement, and naturalness, for both test datasets. These results confirm the effectiveness of performing the transformations in the learned latent subspaces and not directly in the latent space, as well as the advantage of using regression models instead of predefined latent attribute representations. To analyze the disentanglement results, when performing the transformation for a given factor f_i , one must compare the movement of other factors $\{f_j, j \neq i\}$ relative to their fixed targets (as given by the metrics $\{\delta f_j, j \neq i\}$) to the movement of the factor f_i that is varied relative to the center value of its variation range (as given in the last column of Table 1). For instance, when f_0 is varied in the experiments on the English vowels dataset, Table 2 shows that the proposed method makes the other factors move from their fixed targets by 7.2%, 3.6%, and 3.8% for f_1 , f_2 , and f_3 respectively. These values are much smaller than the relative variation of the factor f_0 itself, as indicated in the last column of Table 1 is equal to 50%. We can thus conclude that modifying f_0 has little effect on the other factors. Similar conclusions can be drawn by analyzing the disentanglement results

Dataset	NISQA (\uparrow)	δf_0 (% , \downarrow)	δf_1 (% , \downarrow)	δf_2 (% , \downarrow)	δf_3 (% , \downarrow)
WSJ	2.08 \pm 0.48	0.8 \pm 0.2	7.2 \pm 1.3	3.6 \pm 1.2	3.8 \pm 0.3
SIWIS	1.93 \pm 0.43	1.2 \pm 0.5	10.0 \pm 4.2	8.3 \pm 1.1	14.0 \pm 0.2
TESS	1.98 \pm 0.50	2.7 \pm 2.3	9.3 \pm 3.5	9.0 \pm 0.8	7.0 \pm 0.2
LJspeech	1.96 \pm 0.40	1.2 \pm 0.6	9.3 \pm 1.2	5.6 \pm 0.6	4.6 \pm 0.1

Table 3: Performance (mean and standard deviation) of f_0 transformation with the proposed method, on the English vowels test dataset, using different training datasets for the unsupervised VAE model.

for the variation of other factors, confirming the disentanglement of the learned representation. Regarding f_0 transformation, WORLD obtains the best performance in terms of disentanglement, which is because the source and filter contributions are decoupled in the architecture of the vocoder. In terms of naturalness, WORLD and then TD-PSOLA obtain the best performance. This may be explained by the fact that these methods operate directly in the time domain, therefore they do not suffer from phase reconstruction artifacts, unlike the proposed and VAE baseline methods. Naturalness is indeed greatly affected by phase reconstruction artifacts, even from an unaltered speech spectrogram (i.e., without transformation). Phase reconstruction in a multi-speaker setting is still an open problem in speech processing. We want to emphasize that the objective of this study is not to compete with traditional signal processing methods such as TD-PSOLA and WORLD. It is rather to advance on the understanding of deep generative modeling of speech signals and to compare honestly with highly-specialized traditional systems. TD-PSOLA and WORLD exploit signal models specifically designed for the task at hand, which for instance prevents them to be used for modifying formant frequencies. In contrast, the proposed method is fully based on learning and the same methodology applies to modifying f_0 or the formant frequencies.

3.2.3. Robustness with respect to the VAE training dataset

In this Section, we investigate the robustness of the proposed method with respect to different datasets used to train the VAE model. We considered three training datasets in addition to the WSJ0 dataset used in the previous experiments: (i) the SIWIS French speech synthesis dataset (Honnet et al., 2017), which contains more than ten hours of French speech recordings; (ii) the Toronto emotional speech (TESS) dataset (Dupuis & Pichora-Fuller, 2010), which contains 2,800 utterances spoken by two actresses using different emotions (anger, disgust, fear, happiness, pleasant surprise, sadness, and neutral); and (iii) the LJspeech dataset (Ito & Johnson, 2017), which contains 13,100 short audio clips of a single speaker reading passages from 7 non-fiction books. The artificially-generated speech dataset used for learning the latent subspaces and the regression models remain the same.

Table 3 presents the results for the modification of f_0 only, applied to the English vowels dataset. It can be seen in Table 3 that the performance remains quite stable with different VAE training datasets. WSJ0 is the largest dataset and therefore leads to the best performance. Interestingly, the results obtained with the SIWIS dataset of French speech signals remain satisfactory, even if there is a mismatch between the training (French) and testing (English) datasets.

3.2.4. Robust f_0 estimation

Experimental set-up. For the experiments on fundamental frequency estimation, we used the Pitch Tracking Database from Graz University of Technology (PTDB-TUG) Pirker et al. (2011). This dataset provides microphone and laryngograph signals from 20 native English speakers. The ground truth f_0 is extracted from the laryngograph signals. To evaluate the robustness of the f_0 estimation methods, we corrupted each clean speech test signal by adding real-world cafeteria noise extracted from the DEMAND dataset (Thiemann et al., 2013), where the signal-to-noise ratio (SNR) is varied from -10 to 40 dB. In this experiment, for both the proposed and reference methods, we only considered the problem of estimating f_0 given the ground-truth voicing labels, i.e., we did not consider the problem of detecting which frames are voiced.

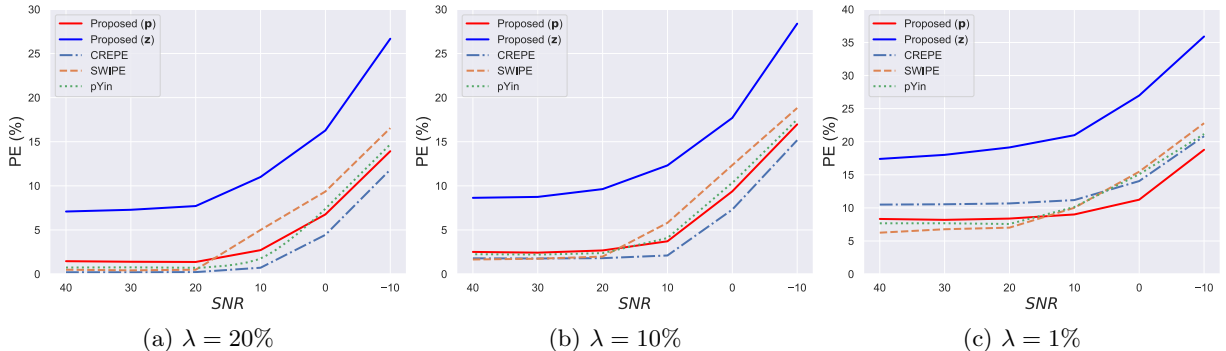


Figure 4: Results of the f_0 tracking experiment: Pitch error (PE, in %) as a function of the SNR (in dB), for different values of the threshold λ (in %). “Proposed(**p**)” and “Proposed(**z**)” denote the proposed approach for f_0 estimation using the projection of \mathbf{z} into the learned subspace of the pitch and using \mathbf{z} directly without the projection, respectively.

Metric. Following Rabiner et al. (1976), the performance is measured in terms of pitch error (PE) defined as the proportion of frames considered as voiced by both the estimation algorithm and the ground truth for which the relative f_0 error is higher than a certain threshold λ in %:

$$PE = \frac{N_{est}}{N_v} \times 100\%, \quad (15)$$

where N_v is the number of voiced frames and N_{est} is the number of the frames for which $|\hat{y} - y|/y \geq \lambda$ with \hat{y} and y the estimated and ground-truth f_0 values, respectively. The parameter λ can be interpreted as the tolerance for which the f_0 prediction is considered as correct.

Methods. We compared the proposed approach described in Section 2.5 with several state-of-the-art methods from the literature: (i) pYin (Mauch & Dixon, 2014) improvement of the Yin-based autocorrelation algorithm in the time domain using probabilistic threshold distribution. We used the one implemented in the Librosa package (McFee et al., 2015); (ii) SWIPE (Camacho & Harris, 2008) is a method for pitch detection based on the autocorrelation of the speech signal in the frequency domain. We used the one implemented in the pysptk toolkit (Yamamoto et al., 2019); (iii) CREPE (Kim et al., 2018) is a supervised method based on CNN already used in Section 3.2.2. We used the one implemented in the torchcrepe github of Morrison (2020). To confirm the interest of estimating the f_0 in the corresponding latent subspace and not directly in the VAE latent space, we also apply the proposed method described in 2.5 by directly working with $q_\phi(\mathbf{z}|\mathbf{x})$ instead of $q_\phi(\mathbf{p}|\mathbf{x})$ to compute the KL divergence in equation (14). This aims to show that projecting the speech signal in the latent subspace encoding f_0 indeed helps discarding information that is not related to this factor and thus improves the estimation accuracy.

Discussion. The results are shown in Figure 4. Figures 4a, 4b, and 4c display the PE metric (%) as a function of the SNR for λ equal to 20%, 10%, and 1%, respectively. The proposed pitch estimation method based on the projection of the latent variable \mathbf{z} on the f_0 subspace outperforms the same approach using the latent variable \mathbf{z} directly. This result confirms the fact that the f_0 subspace is less sensitive to formant variations (invariance property), and projecting \mathbf{z} into this subspace globally preserves well the pitch information, which makes the detection more robust to variations that are independent of f_0 . For large λ values (i.e., high precision tolerance), the proposed method gives results that are competitive with the state-of-the-art methods for moderate noise levels (high SNRs) and that outperform SWIPE for a high level of noise (low SNRs). For small λ values (i.e., low precision tolerance), the proposed approach shows the best trade-off between robustness to noise and precision. Indeed, the proposed method is about 2% PE below CREPE, consistently over the whole SNR range. It also outperforms SWIPE and pYin below 14dB SNR.

4. Conclusion

The source-filter model of speech production is a fundamental concept in speech processing. In this work, using only a few seconds of artificially generated labeled speech data, we showed that the fundamental frequency and formant frequencies are encoded in orthogonal latent subspaces of an unsupervised VAE. We proposed to exploit this disentangled source-filter latent representation for the transformation and analysis of speech spectrograms. Using a regression model trained on the artificially-generated labeled speech data, we proposed a method to control f_0 and the formant frequencies in speech spectrograms by applying affine transformations in the learned latent subspaces. We also proposed to exploit the projection of the speech signal in the latent subspace associated with f_0 to robustly estimate the latter from speech signals corrupted by noise. Even if the identification of the latent source-filter subspaces, the learning of the regression models, and the f_0 estimation method were designed using a very limited set of artificially-generated signals, we showed experimentally that the speech transformations and analysis are effective on natural signals. To the best of our knowledge, this is the first approach that, with a single methodology, is able to extract, identify and control the source and filter low-level speech attributes within a VAE latent space. This is an important step towards a better understanding of deep latent-variable generative modeling of speech signals.

Future work includes improving the quality of the generated speech waveforms, by addressing the phase reconstruction issue or by directly modeling the speech waveform (Caillon & Esling, 2021). It also includes extending the proposed method to dynamical VAEs (Girin et al., 2021), to hierarchical latent spaces (Vahdat & Kautz, 2020), and to audio-visual speech processing. The proposed model could also be applied to pitch-informed speech enhancement. Indeed, several recent weakly-supervised speech enhancement methods consist in estimating the VAE latent representation of a clean speech signal given a noisy speech signal (Bando et al., 2018; Leglaive et al., 2018; Sekiguchi et al., 2018; Leglaive et al., 2019b,a; Pariente et al., 2019; Leglaive et al., 2020; Richter et al., 2020; Carbajal et al., 2021; Fang et al., 2021). Using the proposed conditional deep generative speech model, this estimation could be constrained given the f_0 contour of the signal.

Appendix A. Solution to the latent subspace learning problem

In this Appendix, we show that the solution to the optimization problem (7) is given by the principal eigenvectors of $\mathbf{S}_\phi(\mathcal{D}_i)$ in Equation (8). Without loss of generality, we formulate the problem for a centered version of the latent data:

$$\mathbf{z} \leftarrow \mathbf{z} - \boldsymbol{\mu}_\phi(\mathcal{D}_i), \quad (\text{A.1})$$

where $\boldsymbol{\mu}_\phi(\mathcal{D}_i)$ is defined in Equation (6). This centering also affects the inference model originally defined in Equation (3), as follows:

$$q_\phi(\mathbf{z}|\mathbf{x}) = \mathcal{N}(\mathbf{z}; \boldsymbol{\mu}_\phi(\mathbf{x}) - \boldsymbol{\mu}_\phi(\mathcal{D}_i), \text{diag}\{\mathbf{v}_\phi(\mathbf{x})\}). \quad (\text{A.2})$$

Using Equation (5), the fact that $\mathbf{U}^\top \mathbf{U} = \mathbf{I}$, and Equation (A.2), the cost function in the optimization problem (7) can be rewritten as follows:

$$\begin{aligned} \mathbb{E}_{q_\phi^{(i)}(\mathbf{z})} \left[\|\mathbf{z} - \mathbf{U}\mathbf{U}^\top \mathbf{z}\|_2^2 \right] &= \frac{1}{\#\mathcal{D}_i} \sum_{\mathbf{x}_n \in \mathcal{D}_i} \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x}_n)} \left[\|\mathbf{z} - \mathbf{U}\mathbf{U}^\top \mathbf{z}\|_2^2 \right] \\ &= \text{tr} \left\{ (\mathbf{I} - \mathbf{U}\mathbf{U}^\top) \frac{1}{\#\mathcal{D}_i} \sum_{\mathbf{x}_n \in \mathcal{D}_i} \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x}_n)} [\mathbf{z}\mathbf{z}^\top] \right\} \\ &= \text{tr} \{ (\mathbf{I} - \mathbf{U}\mathbf{U}^\top) \mathbf{S}_\phi(\mathcal{D}_i) \}, \end{aligned} \quad (\text{A.3})$$

where $\mathbf{S}_\phi(\mathcal{D}_i)$ is defined in Equation (8). From this last equality, we see that the optimization problem (7) is equivalent to

$$\max_{\mathbf{U} \in \mathbb{R}^{L \times M_i}} \text{tr} \{ \mathbf{U}^\top \mathbf{S}_\phi(\mathcal{D}_i) \mathbf{U} \}, \quad \text{s.t. } \mathbf{U}^\top \mathbf{U} = \mathbf{I}. \quad (\text{A.4})$$

Very similarly to PCA (Pearson, 1901), the solution is given by the M_i dominant eigenvectors of $\mathbf{S}_\phi(\mathcal{D}_i)$ (i.e., associated to the M_i largest eigenvalues) (Bishop, 2006, Section 12.1).

Appendix B. Algorithms

Algorithm 1 Learning of the regression model.

Input:

- $\mathcal{D}_i = \{\mathbf{x}_n \in \mathbb{R}_+^D, y_n \in \mathbb{R}_+\}_{n=1}^N$: Dataset of artificially-generated speech power spectra \mathbf{x}_n and the corresponding values y_n for the factor f_i
- $\mathbf{U}_i \in \mathbb{R}^{L \times M_i}$: Latent subspace matrix associated with f_i obtained by solving (7)
- Encoder_ϕ : Pre-trained VAE encoder associated with the inference model $q_\phi(\mathbf{z}|\mathbf{x})$ in Equation (3)
- $\boldsymbol{\mu}_\phi(\mathcal{D}_i)$: Empirical posterior mean vector over \mathcal{D}_i , defined in Equation (6)
- $\text{Solver}(\text{inputs}, \text{labels})$: Continuous piecewise linear regression solver implemented in the `pwl` Python library of [Jekel & Venter \(2019\)](#)

Algorithm:

- 1: $\mathbf{p} = \{\}$ # placeholder to store the coordinates of the speech power spectra in the latent subspace
- 2: $y = \{\}$ # placeholder to store the f_i value associated with \mathbf{p}
- 3: **for** (\mathbf{x}_n, y_n) in \mathcal{D}_i **do**
- 4: $\boldsymbol{\mu}_\phi(\mathbf{x}_n) = \text{Encoder}_\phi(\mathbf{x}_n)$
- 5: $\mathbf{p}_n = \mathbf{U}_i^\top (\boldsymbol{\mu}_\phi(\mathbf{x}_n) - \boldsymbol{\mu}_\phi(\mathcal{D}_i))$
- 6: $\mathbf{p}.\text{append}(\mathbf{p}_n)$
- 7: $y.\text{append}(y_n)$
- 8: **end for**
- 9: $\mathbf{g}_{\eta_i} = \text{Solver}(y, \mathbf{p})$

Output:

$\mathbf{g}_{\eta_i} : \mathbb{R}_+ \mapsto \mathbb{R}^{M_i}$: Estimated piecewise linear regression model to map an f_i value to the coordinates of the speech power spectra in the latent subspace characterized by \mathbf{U}_i

Algorithm 2 Fundamental and formant frequency manipulation in the learned latent subspaces.

Input:

- $\mathbf{x} \in \mathbb{R}_+^D$: Speech power spectrum
- $y \in \mathbb{R}_+$: Target value for the factor f_i to be modified
- $\mathbf{U}_i \in \mathbb{R}^{L \times M_i}$: Latent subspace matrix associated with f_i obtained by solving (7)
- Encoder_ϕ : Pre-trained VAE encoder associated with the inference model $q_\phi(\mathbf{z}|\mathbf{x})$ in Equation (3)
- Decoder_θ : Pre-trained VAE decoder associated with the generative model $p_\theta(\mathbf{x}|\mathbf{z})$ in Equation (2)
- $\mathbf{g}_{\eta_i} : \mathbb{R}_+ \mapsto \mathbb{R}^{M_i}$: Regression model obtained with Algorithm 1
- $\text{randn}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$: Function to sample from a multivariate Gaussian distribution with mean vector $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$

Algorithm:

- 1: $\boldsymbol{\mu}_\phi(\mathbf{x}), \mathbf{v}_\phi(\mathbf{x}) = \text{Encoder}_\phi(\mathbf{x})$
- 2: $\mathbf{z} = \text{randn}(\boldsymbol{\mu}_\phi(\mathbf{x}_n), \text{diag}\{\mathbf{v}_\phi(\mathbf{x})\})$
- 3: $\tilde{\mathbf{z}} = \mathbf{z} - \mathbf{U}_i \mathbf{U}_i^\top \mathbf{z} + \mathbf{U}_i \mathbf{g}_{\eta_i}(y)$
- 4: $\tilde{\mathbf{x}} = \text{Decoder}_\theta(\tilde{\mathbf{z}})$

Output:

$\tilde{\mathbf{x}} \in \mathbb{R}_+^D$: Transformed speech power spectrum corresponding to the input target value y for the factor f_i

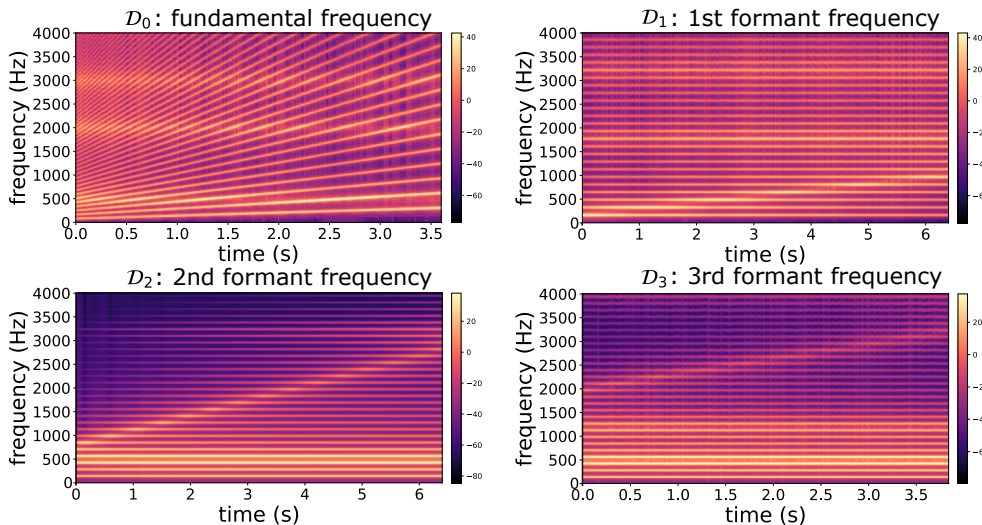


Figure C.5: Trajectories of speech power spectra generated with Soundgen (Anikin, 2019), where only one factor of variation globally varies in each trajectory. From top to bottom and from left to right: the trajectory of the fundamental frequency f_0 , the trajectory of the formant f_1 , the trajectory of the formant f_2 and the trajectory of the formant f_3 .

Appendix C. Experimental setup details

VAE training. To train the IS-VAE model (Bando et al., 2018; Leglaive et al., 2018; Girin et al., 2019), we use the Wall Street Journal (WSJ0) dataset (Garofalo et al., 1993), which contains 25 hours of speech signals sampled at 16 kHz, including 52 female and 49 male speakers. The time-domain speech signals are converted to power spectrograms using the short-time Fourier transform (STFT) with a Hann analysis window of length 64 ms (1,024 samples) and an overlap of 75%. The VAE input/output dimension is $D = 513$ (we only keep the non-redundant part of the power spectrogram corresponding to positive frequencies) and the latent vector dimension is set to $L = 16$. The VAE encoder and decoder networks each have three dense layers. Their dimensions (input dimension, output dimension) are (513, 256), (256, 64) and (64, 2×16) for the encoder, and (16, 64), (64, 256) and (256, 513) for the decoder³. A hyperbolic tangent (tanh) activation function is used at each layer, except for the output layers of the encoder and decoder where we use the identity function. We train the model using the Adam optimizer (Kingma & Ba, 2015) with a learning rate equal to 0.001.

Artificially generated speech data. For a given factor of variation, the corresponding latent subspace is learned using trajectories of speech power spectra generated with Soundgen (Anikin, 2019), all other factors being arbitrarily fixed (see Section 2.2). For f_0 , the trajectory contains 226 points (which corresponds to 3.6 seconds of speech) evenly spaced in the range [85, 310] Hz, f_1 , f_2 and f_3 being fixed to 600 Hz, 2000 Hz, and 3000 Hz, respectively. For f_1 , the trajectory contains 401 points (which corresponds to 6.4 seconds of speech) evenly spaced in the range [200, 1000] Hz, f_0 , f_2 and f_3 being fixed to 140 Hz, 1600 Hz, and 3200 Hz, respectively. For f_2 , the trajectory contains 401 points evenly spaced in the range [800, 2800] Hz, f_0 , f_1 and f_3 being fixed to 140 Hz, 500 Hz, and 3200 Hz, respectively. For f_3 , the trajectory contains 241 points (which corresponds to 3.9 seconds of speech) evenly spaced in the range [2000, 3200] Hz, f_0 , f_1 and f_2 are fixed to 140 Hz, 500 Hz, and 1200 Hz, respectively. These four trajectories are illustrated in Figure C.5. The amplitude of the formants is fixed at 30dB, and their bandwidth is automatically calculated from the formant frequencies using a formula derived from phonetics studies. Quoting the documentation of Soundgen (Anikin, 2019), “above 500 Hz [the bandwidth] follows the original formula known as “TMF-1963”

³We have two parameter vectors (mean and variance) for the distribution of \mathbf{z} at the encoder, whereas we have only one single parameter vector (scale) for the distribution of \mathbf{x} at the decoder, see Equation (2) and Equation (3).

(Tappert et al., 1963), and below 500 Hz it applies a correction to allow for energy losses at low frequencies (Khodai-Joopari & Clermont, 2002). Below 250 Hz the bandwidth starts to decrease again, in a purely empirical attempt to achieve reasonable values even for formant frequencies below ordinary human range. See the internal function `soundgen:::getBandwidth()`. The regression models used to control the speech factors of variation in the latent space (see Section 2.4) are learned on the same trajectories, but using the values of the Soundgen input parameters.

Appendix D. Visualization of the learned latent subspaces

Table D.4: Cumulative variance (in %) retained by the projection $\mathbf{U}_i \mathbf{U}_i^\top$, $\mathbf{U}_i \in \mathbb{R}^{L \times M_i}$, as a function of the number of components M_i . We keep as much components as needed to retain at least 80 % of the data variance, as indicated by the underlined numbers.

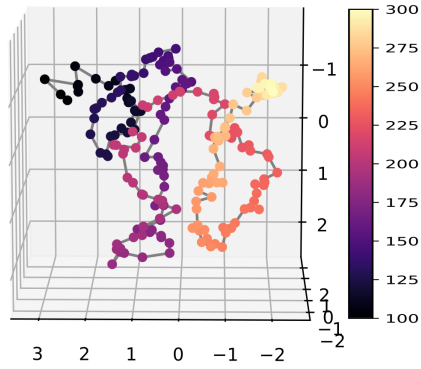
	f_0	f_1	f_2	f_3
$M_i = 1$ component	33	<u>81</u>	39	48
$M_i = 2$ components	59	87	70	73
$M_i = 3$ components	78	90	<u>88</u>	<u>83</u>
$M_i = 4$ components	<u>90</u>	92	93	87

For $i = 0, 1, 2$ and 3 , Figures E.6a, E.6b, E.6c and E.6d are respectively obtained by projecting the latent mean vectors $\boldsymbol{\mu}_\phi(\mathbf{x}) \in \mathbb{R}^L$, for all data vectors $\mathbf{x} \in \mathcal{D}_i$, within the latent subspace characterized by $\mathbf{U}_i \in \mathbb{R}^{L \times M_i}$ (i.e., we perform dimensionality reduction). In the reported experiments, the latent subspace dimension M_i for each factor of variation was chosen such that 80% of the data variance was retained in the latent space. As indicated in Table D.4, this resulted in $M_0 = 4$, $M_1 = 1$ and $M_2 = M_3 = 3$. In this section, for visualization purposes, we set $M_i = 3$ for all $i \in \{0, 1, 2, 3\}$. However, we can see that the f_1 trajectory (Figure E.6b) is mainly concentrated along a single axis, as indicated by the amount of variance retained by this axis 81% (see Table D.4). Regarding f_0 (Figure E.6a), setting $M_0 = 3$ retained 78% of the variance of \mathcal{D}_0 in the latent space. From these figures, we see that two data vectors \mathbf{x} and \mathbf{x}' corresponding to two close values of a given factor have projections of $\boldsymbol{\mu}_\phi(\mathbf{x})$ and $\boldsymbol{\mu}_\phi(\mathbf{x}')$ that are also close in the learned latent subspaces. This can be seen from the color bars which indicate the values of the factors of variation. The learned representation thus preserves the notion of proximity in terms of f_0 and formant frequencies.

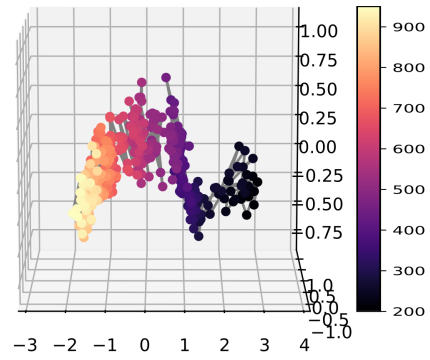
In Figure E.6e, we project three different datasets \mathcal{D}_1 , defined for three different values of f_2 . Similarly, in Figure E.6f we show the trajectories associated with the projection of three datasets \mathcal{D}_2 , defined for three different values of f_1 . We notice that as expected, the trajectories are very similar and mostly differ by a translation.

Appendix E. Correlation matrices obtained from MFCCs and short-term magnitude spectra

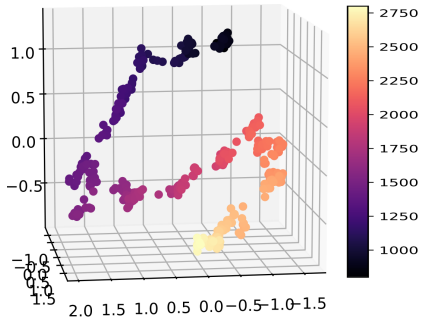
We conducted experiments with the proposed method (i.e., learning the subspace for each factor of variation and then learning a regression model to move in the learned subspace) on the following representations: MFCC and short-term magnitude spectrum (i.e., columns of the STFT magnitude spectrogram). We used the same artificial dataset as for the VAE latent space representation. Figures E.7a and E.7b below show the correlation matrix of the latent subspace basis vectors learned for the MFCC and short-term magnitude spectrum, respectively. Similarly to what we did with the VAE, the dimension of the subspaces is determined by applying a threshold on the data variance ($\geq 80\%$). For the dimension of the f_0 subspace with the MFCCs, we actually need 15 components to keep 80% of the data variance, so here we take only the principal one to facilitate the reading.



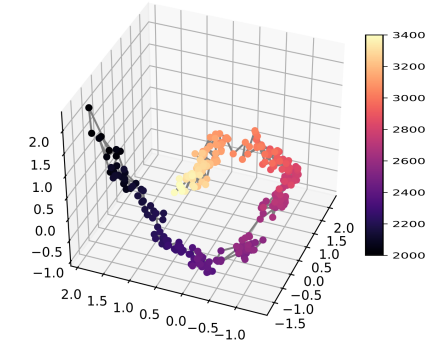
(a) f_0 trajectory corresponding to \mathcal{D}_0 .



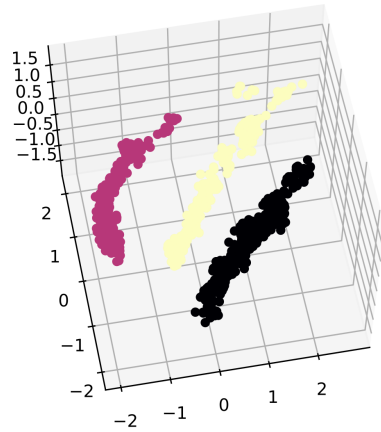
(b) f_1 trajectory corresponding to \mathcal{D}_1 .



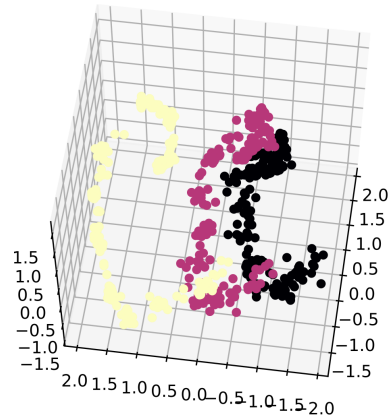
(c) f_2 trajectory corresponding to \mathcal{D}_2 .



(d) f_3 trajectory corresponding to \mathcal{D}_3 .



(e) Three trajectories for f_1 , associated with different values of f_2 .



(f) Three trajectories for f_2 , associated with different values of f_1 .

Figure E.6: Visualization of trajectories in the learned latent subspaces.

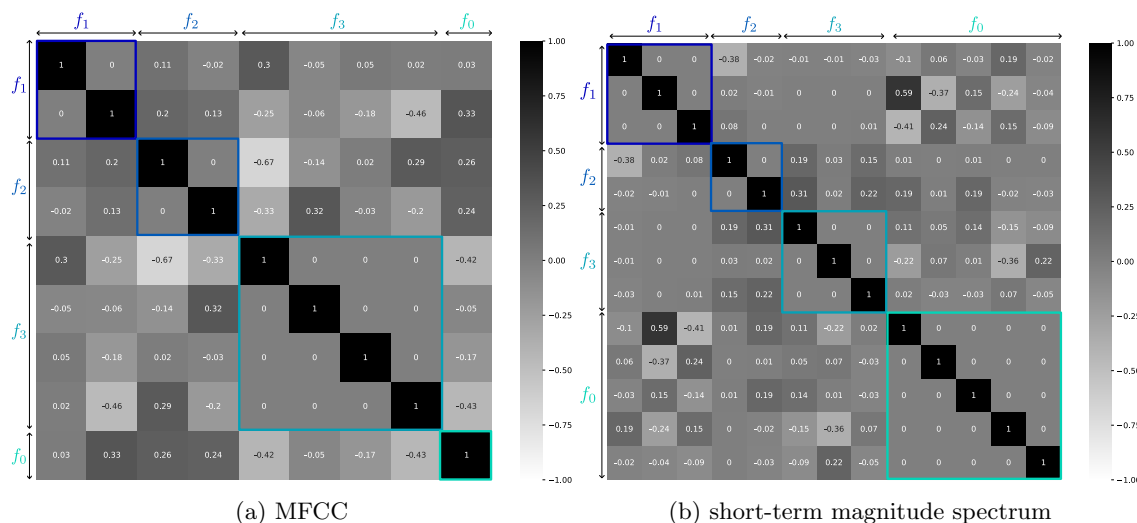


Figure E.7: Correlation matrix of the latent subspace basis vectors learned for MFCC and short-term magnitude spectrum.

- Akuzawa, K., Iwasawa, Y., & Matsuo, Y. (2018). Expressive speech synthesis via modeling expressions with variational autoencoder. In *Interspeech* (pp. 3067–3071).
- Anikin, A. (2019). Soundgen: An open-source tool for synthesizing nonverbal vocalizations. *Behavior Research Methods*, 51, 778–792.
- Bando, Y., Mimura, M., Itoyama, K., Yoshii, K., & Kawahara, T. (2018). Statistical speech enhancement based on probabilistic integration of variational autoencoder and non-negative matrix factorization. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (pp. 716–720).
- Banno, H., Hata, H., Morise, M., Takahashi, T., Irino, T., & Kawahara, H. (2007). Implementation of realtime STRAIGHT speech manipulation system: Report on its first implementation. *Acoustical Science and Technology*, 28, 140–146.
- Bengio, Y., Courville, A., & Vincent, P. (2013). Representation learning: A review and new perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35, 1798–1828.
- Bie, X., Leglaive, S., Alameda-Pineda, X., & Girin, L. (2022). Unsupervised speech enhancement using dynamical variational autoencoders. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 30, 2993–3007.
- Bishop, C. M. (2006). *Pattern Recognition and Machine Learning*. Springer.
- Blaauw, M., & Bonada, J. (2016). Modeling and transforming speech using variational autoencoders. In *Interspeech* (pp. 1770–1774).
- Boersma, P., & Weenink, D. (2021). Praat: doing phonetics by computer [Computer program]. Version 6.1.38, retrieved 2 January 2021 <http://www.praat.org/>.
- Bora, A., Jalal, A., Price, E., & Dimakis, A. G. (2017). Compressed sensing using generative models. In *International Conference on Machine Learning (ICML)* (pp. 537–546).
- Caillon, A., & Esling, P. (2021). RAVE: A variational autoencoder for fast and high-quality neural audio synthesis. ArXiv preprint arXiv:2111.05011.
- Camacho, A., & Harris, J. G. (2008). A sawtooth waveform inspired pitch estimator for speech and music. *The Journal of the Acoustical Society of America*, 124, 1638–1652.
- Carbajal, G., Richter, J., & Gerkmann, T. (2021). Guided variational autoencoder for speech enhancement with a supervised classifier. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (pp. 681–685).
- Chen, R. T. Q., Li, X., Grosse, R., & Duvenaud, D. (2018). Isolating sources of disentanglement in variational autoencoders. In *Advances in Neural Information Processing Systems (NeurIPS)* (pp. 2610–2620).
- Chen, X., Duan, Y., Houthoofd, R., Schulman, J., Sutskever, I., & Abbeel, P. (2016). Infogan: Interpretable representation learning by information maximizing generative adversarial nets. In *Advances in Neural Information Processing Systems (NeurIPS)* (pp. 2172–2180).
- Choi, H.-S., Lee, J., Kim, W., Lee, J. H., Heo, H., & Lee, K. (2021). Neural analysis and synthesis: Reconstructing speech from self-supervised representations. In *Advances in Neural Information Processing Systems (NeurIPS)*.
- Dai, B., & Wipf, D. (2018). Diagnosing and enhancing VAE models. In *International Conference on Learning Representations (ICLR)*.
- Dupuis, K., & Pichora-Fuller, M. K. (2010). Toronto emotional speech set (TESS). University of Toronto, Psychology Department.
- Fang, H., Carbajal, G., Wermter, S., & Gerkmann, T. (2021). Variational autoencoder for speech enhancement with a noise-aware encoder. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (pp. 676–680).
- Fant, G. (1970). *Acoustic theory of speech production*. 2. Walter de Gruyter.
- Flanagan, J. L., & Golden, R. M. (1966). Phase vocoder. *Bell System Technical Journal*, 45, 1493–1509.

- Garofalo, J. S., Graff, D., Paul, D., & Pallett, D. (1993). CSR-I (WSJ0) Sennheiser LDC93S6B. Web Download. <https://catalog.ldc.upenn.edu/LDC93S6B>. Philadelphia: Linguistic Data Consortium.
- Garofalo, J. S., Lamel, L. F., Fisher, W. M., Fiscus, J. G., Pallett, D. S., Dahlgren, N. L., & Zue, V. (1993). TIMIT acoustic phonetic continuous speech corpus. Web Download. <https://catalog.ldc.upenn.edu/LDC93s1>. Philadelphia: Linguistic Data Consortium.
- George, E. B., & Smith, M. J. (1997). Speech analysis/synthesis and modification using an analysis-by-synthesis/overlap-add sinusoidal model. *IEEE Transactions on Speech and Audio Processing*, 5, 389–406.
- Girin, L., Leglaive, S., Bie, X., Diard, J., Hueber, T., & Alameda-Pineda, X. (2021). Dynamical variational autoencoders: A comprehensive review. *Foundations and Trends in Machine Learning*, 15, 1–175.
- Girin, L., Roche, F., Hueber, T., & Leglaive, S. (2019). Notes on the use of variational autoencoders for speech and audio spectrogram modeling. In *International Conference on Digital Audio Effects (DAFx)* (pp. 1–8).
- Goetschalckx, L., Andonian, A., Oliva, A., & Isola, P. (2019). GANalyze: Toward visual definitions of cognitive image properties. In *IEEE/CVF International Conference on Computer Vision (ICCV)* (pp. 5744–5753).
- Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., & Bengio, Y. (2014). Generative adversarial nets. In *Advances in Neural Information Processing Systems (NeurIPS)* (pp. 2672–2680).
- Härkönen, E., Hertzmann, A., Lehtinen, J., & Paris, S. (2020). GANSpace: Discovering interpretable GAN controls. In *Advances in Neural Information Processing Systems (NeurIPS)* (pp. 9841–9850).
- Higgins, I., Amos, D., Pfau, D., Racaniere, S., Matthey, L., Rezende, D. J., & Lerchner, A. (2018). Towards a definition of disentangled representations. ArXiv preprint arXiv:1812.02230.
- Higgins, I., Matthey, L., Pal, A., Burgess, C., Glorot, X., Botvinick, M., Mohamed, S., & Lerchner, A. (2017). Beta-VAE: Learning basic visual concepts with a constrained variational framework. In *International Conference on Learning Representations (ICLR)*.
- Hillenbrand, J., Getty, L. A., Clark, M. J., & Wheeler, K. (1995). Acoustic characteristics of American English vowels. *The Journal of the Acoustical Society of America*, 97, 3099–3111.
- Honnet, P.-E., Lazaridis, A., Garner, P. N., & Yamagishi, J. (2017). *The SIWIS French speech synthesis database - Design and recording of a high quality French database for speech synthesis*. Technical Report Idiap.
- Hosoya, H. (2018). Group-based learning of disentangled representations with generalizability for novel contents. ArXiv preprint arXiv:1809.02383.
- Hotelling, H. (1957). The relations of the newer multivariate statistical methods to factor analysis. *British Journal of Statistical Psychology*, 10, 69–79.
- Hsu, C.-C., Hwang, H.-T., Wu, Y.-C., Tsao, Y., & Wang, H.-M. (2016). Voice conversion from non-parallel corpora using variational auto-encoder. In *Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA)* (pp. 1–6).
- Hsu, W.-N., Zhang, Y., & Glass, J. (2017a). Learning latent representations for speech generation and transformation. In *Interspeech* (pp. 1273–1277).
- Hsu, W.-N., Zhang, Y., & Glass, J. (2017b). Unsupervised learning of disentangled and interpretable representations from sequential data. In *Advances in Neural Information Processing Systems (NeurIPS)* (pp. 1878–1889).
- Ito, K., & Johnson, L. (2017). The LJ speech dataset. <https://keithito.com/LJ-Speech-Dataset/>.
- Jadoul, Y., Thompson, B., & de Boer, B. (2018). Introducing Parselmouth: A Python interface to Praat. *Journal of Phonetics*, 71, 1–15.
- Jahani, A., Chai, L., & Isola, P. (2019). On the “steerability” of generative adversarial networks. In *International Conference on Learning Representations (ICLR)*.
- Jayaram, V., & Thickstun, J. (2020). Source separation with deep generative priors. In *International Conference on Machine Learning (ICML)* (pp. 4724–4735).
- Jekel, C. F., & Venter, G. (2019). PWLF: a Python library for fitting 1D continuous piecewise linear functions. URL: https://github.com/cjekel/piecewise_linear_fit_py/raw/master/paper/pwlf_Jekel_Venter_v2.pdf.
- Juvela, L., Bollepalli, B., Tsiaras, V., & Alku, P. (2019). GlotNeta raw waveform model for the glottal excitation in statistical parametric speech synthesis. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 27, 1019–1030.
- Kameoka, H., Li, L., Inoue, S., & Makino, S. (2019). Supervised determined source separation with multichannel variational autoencoder. *Neural Computation*, 31, 1891–1914.
- Kawahara, H. (2006). STRAIGHT, exploitation of the other aspect of VOCODER: Perceptually isomorphic decomposition of speech sounds. *Acoustical Science and Technology*, 27, 349–353.
- Kendall, M. (1957). *A course in multivariate analysis*. London: Charles Griffin.
- Khodai-Joopari, M., & Clermont, F. (2002). A comparative study of empirical formulae for estimating vowel-formant bandwidths. In *Australian International Conference on Speech, Science, and Technology* (pp. 130–135).
- Kim, H., & Mnih, A. (2018). Disentangling by factorising. In *International Conference on Machine Learning (ICML)* (pp. 2649–2658).
- Kim, J. W., Salamon, J., Li, P., & Bello, J. P. (2018). Crepe: A convolutional representation for pitch estimation. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (pp. 161–165).
- Kingma, D. P., & Ba, J. (2015). Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR)*.
- Kingma, D. P., & Welling, M. (2014). Auto-encoding variational Bayes. In *International Conference on Learning Representations (ICLR)*.
- Laroche, J. (2002). Time and pitch scale modification of audio signals. In *Applications of digital signal processing to audio and acoustics* (pp. 279–309). Springer.

- Laroche, J., & Dolson, M. (1999). Improved phase vocoder time-scale modification of audio. *IEEE Transactions on Speech and Audio processing*, 7, 323–332.
- Laroche, J., Stylianou, Y., & Moulines, E. (1993). HNS: Speech modification based on a harmonic+noise model. In *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)* (pp. 550–553).
- Lee, J., Choi, H.-S., Jeon, C.-B., Koo, J., & Lee, K. (2019). Adversarially trained end-to-end Korean singing voice synthesis system. In *Interspeech* (pp. 2588–2592).
- Leglaive, S., Alameda-Pineda, X., Girin, L., & Horaud, R. (2020). A recurrent variational autoencoder for speech enhancement. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (pp. 371–375).
- Leglaive, S., Şimşekli, U., Liutkus, A., Girin, L., & Horaud, R. (2019a). Speech enhancement with variational autoencoders and alpha-stable distributions. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (pp. 541–545).
- Leglaive, S., Girin, L., & Horaud, R. (2018). A variance modeling framework based on variational autoencoders for speech enhancement. In *IEEE International Workshop on Machine Learning for Signal Processing (MLSP)* (pp. 1–6).
- Leglaive, S., Girin, L., & Horaud, R. (2019b). Semi-supervised multichannel speech enhancement with variational autoencoders and non-negative matrix factorization. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (pp. 101–105).
- Locatello, F., Bauer, S., Lucic, M., Raetsch, G., Gelly, S., Schölkopf, B., & Bachem, O. (2019). Challenging common assumptions in the unsupervised learning of disentangled representations. In *International Conference on Machine Learning (ICML)* (pp. 4114–4124).
- Locatello, F., Bauer, S., Lucic, M., Rätsch, G., Gelly, S., Schölkopf, B., & Bachem, O. (2020a). A sober look at the unsupervised learning of disentangled representations and their evaluation. *Journal of Machine Learning Research*, 21, 1–62.
- Locatello, F., Poole, B., Rätsch, G., Schölkopf, B., Bachem, O., & Tschannen, M. (2020b). Weakly-supervised disentanglement without compromises. In *International Conference on Machine Learning (ICML)* (pp. 6348–6359).
- Locatello, F., Tschannen, M., Bauer, S., Rätsch, G., Schölkopf, B., & Bachem, O. (2020c). Disentangling factors of variation using few labels. In *International Conference on Learning Representations (ICLR)*.
- MacDonald, E. N., Purcell, D. W., & Munhall, K. G. (2011). Probing the independence of formant control using altered auditory feedback. *The Journal of the Acoustical Society of America*, 129, 955–965.
- Makhoul, J. (1975). Linear prediction: A tutorial review. *Proceedings of the IEEE*, 63, 561–580.
- Makhzani, A., Shlens, J., Jaitly, N., Goodfellow, I., & Frey, B. (2016). Adversarial autoencoders. In *International Conference on Learning Representations (ICLR) Workshop*.
- Markel, J. D., & Gray, A. J. (1976). *Linear Prediction of Speech*. New York: Springer-Verlag.
- Mauch, M., & Dixon, S. (2014). pYIN: A fundamental frequency estimator using probabilistic threshold distributions. In *International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (pp. 659–663).
- McAulay, R., & Quatieri, T. (1986). Speech analysis/synthesis based on a sinusoidal representation. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 34, 744–754.
- McFee, B., Raffel, C., Liang, D., Ellis, D. P., McVicar, M., Battenberg, E., & Nieto, O. (2015). librosa: Audio and music signal analysis in python. In *Proceedings of the 14th python in science conference* (pp. 18–25). volume 8.
- Mittag, G., & Möller, S. (2020). Deep learning based assessment of synthetic speech naturalness. In *Interspeech* (pp. 1748–1752).
- Morise, M., Yokomori, F., & Ozawa, K. (2016). World: a vocoder-based high-quality speech synthesis system for real-time applications. *IEICE Transactions on Information and Systems*, 99, 1877–1884.
- Morrison, M. (2020). torchcrepe. URL: <https://github.com/maxmorrison/torchcrepe>.
- Morrison, M., Jin, Z., Bryan, N. J., Caceres, J.-P., & Pardo, B. (2021). Neural pitch-shifting and time-stretching with controllable LPCNet. ArXiv preprint arXiv:2110.02360.
- Moulines, E., & Charpentier, F. (1990). Pitch-synchronous waveform processing techniques for text-to-speech synthesis using diphones. *Speech Communication*, 9, 453–467.
- Pariente, M., Deleforge, A., & Vincent, E. (2019). A statistically principled and computationally efficient approach to speech enhancement using variational autoencoders. In *Interspeech* (pp. 3158–3162).
- Pearson, K. (1901). On lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 2, 559–572.
- Pirker, G., Wohlmayr, M., Petrik, S., & Pernkopf, F. (2011). A pitch tracking corpus with evaluation on multipitch tracking scenario. In *Interspeech* (pp. 1509–1512).
- Plumerault, A., Borgne, H. L., & Hudelot, C. (2020). Controlling generative models with continuous factors of variations. In *International Conference on Learning Representations (ICLR)*.
- Prenger, R., Valle, R., & Catanzaro, B. (2019). Waveglow: A flow-based generative network for speech synthesis. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (pp. 3617–3621).
- Qian, K., Zhang, Y., Chang, S., Hasegawa-Johnson, M., & Cox, D. (2020). Unsupervised speech decomposition via triple information bottleneck. In *International Conference on Machine Learning (ICML)* (pp. 7836–7846).
- Rabiner, L., Cheng, M., Rosenberg, A., & McGonegal, C. (1976). A comparative performance study of several pitch detection algorithms. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 24, 399–418.
- Rezende, D. J., Mohamed, S., & Wierstra, D. (2014). Stochastic backpropagation and approximate inference in deep generative models. *International Conference on Machine Learning (ICML)*, (pp. 1278–1286).
- Richter, J., Carbajal, G., & Gerkmann, T. (2020). Speech enhancement with stochastic temporal convolutional networks. In *Interspeech* (pp. 4516–4520).
- Sekiguchi, K., Bando, Y., Nugraha, A. A., Yoshii, K., & Kawahara, T. (2019). Semi-supervised multichannel speech enhancement with a deep speech prior. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 27, 2197–2212.

- Sekiguchi, K., Bando, Y., Yoshii, K., & Kawahara, T. (2018). Bayesian multichannel speech enhancement with a deep speech prior. In *Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA)* (pp. 1233–1239).
- Serra, X., & Smith, J. (1990). Spectral modeling synthesis: A sound analysis/synthesis system based on a deterministic plus stochastic decomposition. *Computer Music Journal*, *14*, 12–24.
- Shu, R., Chen, Y., Kumar, A., Ermon, S., & Poole, B. (2020). Weakly supervised disentanglement with guarantees. In *International Conference on Learning Representations (ICLR)*.
- Sorrenson, P., Rother, C., & Köthe, U. (2020). Disentanglement by nonlinear ICA with general incompressible-flow networks (GIN). In *International Conference on Learning Representations (ICLR)*.
- Tappert, C., Martony, J., & Fant, G. (1963). Spectrum envelopes for synthetic vowels. *Speech Transmission Laboratory Quarterly Progress Status Report*, *4*, 2–6.
- Thiemann, J., Ito, N., & Vincent, E. (2013). Demand: a collection of multi-channel recordings of acoustic noise in diverse environments. In *International Congress on Acoustics (ICA)* (pp. 1–6).
- Vahdat, A., & Kautz, J. (2020). NVAE: A deep hierarchical variational autoencoder. In *Advances in Neural Information Processing Systems (NeurIPS)* (pp. 19667–19679).
- Valin, J.-M., & Skoglund, J. (2019). LPCNet: Improving neural speech synthesis through linear prediction. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (pp. 5891–5895).
- Wang, X., Takaki, S., & Yamagishi, J. (2019). Neural source-filter waveform models for statistical parametric speech synthesis. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, *28*, 402–415.
- Webber, J. J., Perrotin, O., & King, S. (2020). Hider-finder-combiner: An adversarial architecture for general speech signal modification. In *Interspeech* (pp. 3206–3210).
- Yamamoto, R., Felipe, J., & Blaauw, M. (2019). pysptk. URL: <https://github.com/r9y9/pysptk>.