Music sound synthesis using machine learning : Towards a perceptually relevant control space

Fanny Roche

Université Grenoble Alpes

THÈSE

Pour obtenir le grade de

DOCTEUR DE L'UNIVERSITE GRENOBLE ALPES

Spécialité : **Signal, Image, Parole, Télécoms (SIPT)**

Arrêté ministériel : 25 mai 2016

Présentée par

Fanny ROCHE

Thèse dirigée par **Laurent GIRIN,** Professeur, Communauté Université Grenoble Alpes, GIPSA-Lab et
Co-encadrée par **Thomas HUEBER,** Chargé de Recherche, CNRS, GIPSA-Lab
et **Maëva GARNIER,** Chargée de Recherche, CNRS, GIPSA-Lab
et **Samuel LIMIER**, Ingénieur, ARTURIA

préparée au sein du **Laboratoire Grenoble Images Parole Signal Automatique (GIPSA-Lab) et d'ARTURIA**
dans **l'École Doctorale Electronique, Electrotechnique, Automatique, Traitement du Signal (EEATS)**

Music sound synthesis using machine learning: Towards a perceptually relevant control space

Thèse soutenue publiquement le **29 septembre 2020**,
devant le jury composé de :

Monsieur Christophe D'ALESSANDRO
Directeur de Recherche, CNRS, Institut Jean le Rond d'Alembert, Rapporteur, Président du jury
Monsieur Geoffroy PEETERS
Professeur, Télécom ParisTech, LTCI, Rapporteur
Madame Emilia GÓMEZ
Associate Professor, Universitat Pompeu Fabra, MTG, Examinatrice
Monsieur Antoine LIUTKUS
Chargé de Recherche, INRIA, LIRMM, Examinateur
Monsieur Laurent GIRIN
Professeur, Communauté Université Grenoble Alpes, GIPSA-Lab, Directeur de Thèse
Monsieur Thomas HUEBER
Chargé de Recherche, CNRS, GIPSA-Lab, Co-encadrant de Thèse
Madame Maëva GARNIER
Chargée de Recherche, CNRS, GIPSA-Lab, Co-encadrante de Thèse, Invitée
Monsieur Samuel LIMIER
Ingénieur, ARTURIA, Co-encadrant de Thèse, Invité

UNIVERSITÉ DE GRENOBLE ALPES

ÉCOLE DOCTORALE EEATS

Electronique, Electrotechnique, Automatique, Traitement du Signal

T H È S E

pour obtenir le titre de

docteur en sciences

de l'Université de Grenoble Alpes
Mention : Signal, Image, Parole, Télécoms

Présentée et soutenue par

Fanny ROCHE

Music sound synthesis using machine learning: Towards a perceptually relevant control space

Thèse dirigée par Laurent GIRIN et
co-encadrée par Thomas HUEBER, Maëva GARNIER et Samuel LIMIER

préparée au laboratoire Grenoble Images Parole Signal Automatique (GIPSA-Lab) et ARTURIA

soutenue le 29 septembre 2020

Jury :

Président du jury :	Christophe D'ALESSANDRO	CNRS, Institut Jean le Rond d'Alembert
Rapporteur :	Geoffroy PEETERS	Télécom ParisTech, LTCI
Examinatrice :	Emilia GÓMEZ	Universitat Pompeu Fabra, MTG
Examinateur :	Antoine LIUTKUS	INRIA, LIRMM
Directeur :	Laurent GIRIN	Communauté Université Grenoble Alpes, GIPSA-Lab
Encadrant :	Thomas HUEBER	CNRS, GIPSA-Lab
Invitée :	Maëva GARNIER	CNRS, GIPSA-Lab
Invité :	Samuel LIMIER	ARTURIA

Abstract

One of the main challenges of the synthesizer market and the research in sound synthesis nowadays lies in proposing new forms of synthesis allowing the creation of brand new sonorities while offering musicians more intuitive and perceptually meaningful controls to help them reach the perfect sound more easily. Indeed, today's synthesizers are very powerful tools that provide musicians with a considerable amount of possibilities for creating sonic textures, but the control of parameters still lacks user-friendliness and may require some expert knowledge about the underlying generative processes. In this thesis, we are interested in developing and evaluating new data-driven machine learning methods for music sound synthesis allowing the generation of brand new high-quality sounds while providing high-level perceptually meaningful control parameters.

The first challenge of this thesis was thus to characterize the musical synthetic timbre by evidencing a set of perceptual verbal descriptors that are both frequently and consensually used by musicians. Two perceptual studies were then conducted: a free verbalization test enabling us to select eight different commonly used terms for describing synthesizer sounds, and a semantic scale analysis enabling us to quantitatively evaluate the use of these terms to characterize a subset of synthetic sounds, as well as analyze how consensual they were.

In a second phase, we investigated the use of machine learning algorithms to extract a high-level representation space with interesting interpolation and extrapolation properties from a dataset of sounds, the goal being to relate this space with the perceptual dimensions evidenced earlier. Following previous studies interested in using deep learning for music sound synthesis, we focused on autoencoder models and realized an extensive comparative study of several kinds of autoencoders on two different datasets. These experiments, together with a qualitative analysis made with a non real-time prototype developed during the thesis, allowed us to validate the use of such models, and in particular the use of the variational autoencoder (VAE), as relevant tools for extracting a high-level latent space in which we can navigate smoothly and create new sounds. However, so far, no link between this latent space and the perceptual dimensions evidenced by the perceptual tests emerged naturally.

As a final step, we thus tried to enforce perceptual supervision of the VAE by adding a regularization during the training phase. Using the subset of synthetic sounds used in the second perceptual test and the corresponding perceptual grades along the eight perceptual dimensions provided by the semantic scale analysis, it was possible to constraint, to a certain extent, some dimensions of the VAE high-level latent space so as to match these perceptual dimensions. A final comparative test was then conducted in order to evaluate the efficiency of this additional regularization for conditioning the model and (partially) leading to a perceptual control of music sound synthesis.

Keywords: Synthesizer sounds, synthetic timbre perceptual characterization, timbre verbal descriptors, (variational) autoencoders, perceptually controlled audio synthesis.

Résumé

Un des enjeux majeurs du marché des synthétiseurs et de la recherche en synthèse sonore aujourd'hui est de proposer une nouvelle forme de synthèse permettant de générer des sons inédits tout en offrant aux utilisateurs de nouveaux contrôles plus intuitifs afin de les aider dans leur recherche de sons. En effet, les synthétiseurs sont actuellement des outils très puissants qui offrent aux musiciens une large palette de possibilités pour la création de textures sonores, mais également souvent très complexes avec des paramètres de contrôle dont la manipulation nécessite généralement des connaissances expertes. Cette thèse s'intéresse ainsi au développement et à l'évaluation de nouvelles méthodes d'apprentissage machine pour la synthèse sonore permettant la génération de nouveaux sons de qualité tout en fournissant des paramètres de contrôle pertinents perceptivement.

Le premier challenge que nous avons relevé a donc été de caractériser perceptivement le timbre musical synthétique en mettant en évidence un jeu de descripteurs verbaux utilisés fréquemment et de manière consensuelle par les musiciens. Deux études perceptives ont été menées : un test de verbalisation libre qui nous a permis de sélectionner huit termes communément utilisés pour décrire des sons de synthétiseurs, et une analyse à échelles sémantiques permettant d'évaluer quantitativement l'utilisation de ces termes pour caractériser un sous-ensemble de sons, ainsi que d'analyser leur "degré de consensualité".

Dans un second temps, nous avons exploré l'utilisation d'algorithmes d'apprentissage machine pour l'extraction d'un espace de représentation haut-niveau avec des propriétés intéressantes d'interpolation et d'extrapolation à partir d'une base de données de sons, le but étant de mettre en relation cet espace avec les dimensions perceptives mises en évidence plus tôt. S'inspirant de précédentes études sur la synthèse sonore par apprentissage profond, nous nous sommes concentrés sur des modèles du type autoencodeur et avons réalisé une étude comparative approfondie de plusieurs types d'autoencodeurs sur deux jeux de données différents. Ces expériences, couplées avec une étude qualitative via un prototype non temps-réel développé durant la thèse, nous ont permis de valider les autoencodeurs, et en particulier l'autoencodeur variationnel (VAE), comme des outils bien adaptés à l'extraction d'un espace latent de haut-niveau dans lequel il est possible de se déplacer de manière continue et fluide en créant de tous nouveaux sons. Cependant, à ce niveau, aucun lien entre cet espace latent et les dimensions perceptives mises en évidence précédemment n'a pu être établi spontanément.

Pour finir, nous avons donc apporté de la supervision au VAE en ajoutant une régularisation perceptive durant la phase d'apprentissage. En utilisant les échantillons sonores résultant du test perceptif avec échelles sémantiques labellisés suivant les huit dimensions perceptives, il a été possible de contraindre, dans une certaine mesure, certaines dimensions de l'espace latent extrait par le VAE afin qu'elles coïncident avec ces dimensions. Un test comparatif a été finalement réalisé afin d'évaluer l'efficacité de cette régularisation supplémentaire pour conditionner le modèle et permettre un contrôle perceptif (au moins partiel) de la synthèse sonore.

Mots clés : Sons de synthétiseurs, caractérisation perceptive du timbre synthétique, descripteurs verbaux de timbre, autoencodeurs (variationnels), contrôle perceptif de la synthèse sonore.

Aknowledgments

Il y a de ça maintenant un peu plus de six ans dans le froid et la nuit de cette contrée chère à mon coeur, une petite idée folle a germé dans mon esprit : pourquoi pas une thèse ? Et me voilà aujourd'hui arrivée au bout de ce périple qui va mettre un terme à ces (trop diront certains) longues années d'études, bien que ces trois dernières années aient eu un goût assez différent de la prépa ou l'école d'ingé. Je savais dès le départ que l'aventure de la thèse serait longue (sans compter l'"aide" de la pandémie mondiale qui a quelque peu reporté l'échéance...) et difficile, mais je ne me doutais pas de l'importance des personnes avec lesquelles on embarque et/ou qui nous soutiennent, que ce soit scientifiquement ou humainement. Je tiens à remercier dans ces quelques paragraphes qui vont suivre, les personnes qui ont de près ou de loin pris part à cette grande aventure et m'ont aidée à arriver au point où j'en suis aujourd'hui.

Je voudrais tout d'abord commencer par remercier les personnes sans qui l'aventure n'aurait jamais pu voir le jour et être menée à bien.

En tout premier lieu, je souhaite remercier le premier aventurier, ou plutôt devrais-je dire guide, à s'être engagé dans ce projet fou : Sam. Depuis mes débuts en stage à Arturia il y a un peu plus de cinq ans, tu as cru en moi et m'as fait une entière confiance. Tu ne m'as pas seulement suivie dans cette aventure, mais tu m'as aidée à la monter de toute pièce et t'es battu pour que ce projet voit le jour, et je ne saurais te remercier assez pour ça. Merci également pour ton soutien (même dans les moments où tu en avais bien plus besoin que moi), tes conseils toujours avisés, tes moultes relectures de tous mes (souvent trop longs) écrits, et j'en passe. Je suis heureuse d'avoir croisé un jour ta route (à la fois professionnelle et personnelle) et de pouvoir aujourd'hui te compter parmi mes amis.

Ensuite, est arrivé le guide en chef : Laurent. On ne pouvait rêver meilleur guide pour cette aventure. Merci de nous avoir fait confiance et de t'être lancé dans ce projet. Merci pour ton temps, ta disponibilité, ton implication, tes conseils, tes remarques/commentaires toujours pertinents, ta patience avec mon perfectionnisme, ta pédagogie (avec toi le traitement du signal et le framework probabiliste des VAEs n'ont plus de secrets !) et tout le reste. Grâce à toi, j'ai pu valider bon nombre de bulles de compétence de l'arbre du doctorat de ta formation ;) Et surtout, merci de ta confiance tout au long de la thèse et d'avoir su me rassurer à quelques moments clefs de ces trois ans.

Puis Laurent a motivé un nouveau guide : Thomas. Tu nous as rejoint assez rapidement en nous disant qu'on était complètement fous mais que tu voulais bien être de la partie. Motivé à fond, tu as su être moteur de ce projet dès le début et apporter une "tétrachiée" d'idées (si tu me permets de récupérer ton expression ;)) qui nous ont permis d'avancer sans cesse et de ne pas se démotiver. Merci donc pour toutes ces heures de discussions, d'idées échangées et de conseils que ce soit sur Python, Keras ou sur la structure des LSTM. Merci aussi de t'impliquer à fond dans CRISSP et de nous permettre d'avoir accès à des ressources de calculs comme le serveur GPU que j'ai fait chauffé bien plus d'une fois (pas très écolo tout ça mais bon...) et sans quoi certaines deadlines de papiers auraient été dures à tenir !

Puis finalement, la dernière guide est arrivée : Maëva. Merci de m'avoir fait découvrir le monde de la perception et des sciences cognitives, sans toi cette facette importante de la

thèse (que j'ai eu un immense plaisir à traiter et découvrir) n'aurait pas pu exister de cette manière. Je tiens à te remercier pour ta gentillesse, ta disponibilité, la (looooongue) liste de papiers/thèses traitant d'études perceptives de timbre que tu m'as donnée pour que je découvre ce monde, les nombreuses discussions, l'énorme état de l'art que tu as fait pour recenser tous les descripteurs verbaux de la littérature, tes relectures minutieuses et tous tes conseils avisés que ce soit sur l'organisation de tests perceptifs, la reformulation de phrases ou la réorganisation d'un chapitre de thèse ! :)

Enfin bref, j'ai eu une vraie équipe de choc pour m'accompagner dans cette aventure de part sa complémentarité, sa motivation, sa disponibilité ou encore sa facilité à échanger/discuter ouvertement et son écoute ! Après trois ans à Gipsa, j'ai pu voir à quel point ce n'est pas une chose évidente, je me considère donc comme vraiment chanceuse et je ne saurais vous exprimer à quel point je vous en suis reconnaissante. Merci pour tous les échanges, les réunions en présentiel ou par Skype (avec ou sans les enfants ;)), les repas à la piscine, le soutien que vous m'avez apporté durant ces trois ans et j'en passe ! Trouver un juste équilibre entre les attentes du labo et celles de l'entreprise n'est pas forcément facile dans le cadre d'une thèse Cifre, mais ici chaque partie a su tenir son rôle à merveille et si j'en suis à écrire mes remerciements de thèse aujourd'hui c'est en grande partie grâce à vous quatre.

Ensuite, j'aimerais remercier les membres de mon jury de thèse qui ont accepté d'évaluer ce travail : Christophe d'Alessandro, Geoffroy Peeters, Emilia Gómez and Antoine Liutkus. Je vous remercie de m'avoir fait l'honneur d'accepter d'évaluer mes travaux et de l'avoir fait avec pertinence et bienveillance à travers vos questions, vos remarques et vos conseils. J'aimerais d'ailleurs adresser un merci tout particulier à Christophe et Geoffroy qui n'ont pas hésité à braver les risques liés au Covid-19 pour venir à Grenoble assister à la soutenance en personnes et partager une coupe de champagne avec moi pendant le pot. Merci également à Christophe et Antoine pour leur implication dans mon CSI et leurs conseils avisés à des moments clefs de cette thèse.

J'aimerais également remercier les gens qui nous ont fait confiance à Arturia et ont permis à ce projet de voir le jour. Merci Frédéric de nous avoir fait confiance à Sam et moi pour mener à bien cette première thèse Arturienne et nous lancer dans ce long projet. Un grand merci aussi à Kévin M. pour son soutien et sa confiance dès le départ puis tout au long de la thèse et à Seb R. de nous avoir aidé à monter ce projet et d'avoir apporté la vision produit qui nous a permis de garder le cap pendant ces trois (presque quatre maintenant même) ans.

Merci aussi à tous mes collègues (et ex-collègues) Arturiens qui ont su me proposer un environnement agréable de travail le jour et des soirées bières toujours appréciables le mercredi soir. Un merci spécial à la team DSP, la team Soft, aux Arturiennes (en particulier ma *ppt*) et quelques autres qui se reconnaîtront et sont toujours dispo pour me changer les idées, faire un volley/squash/Mölkky/tarot ou une partie de *Limite Limite* !

I also would like to thank my colleagues from Gipsa. People from the CRISSP team and PhD students, in particular the ones I shared my office with (temporarily or not !): Gaël,

Bharat, Duc-Canh, Alex, Anne-Laure. Thanks also to Silvain Gerber for the great help on the statistical analysis of my perceptual tests and to the distinguished members of the MLST (*Machine Learning Seekers of Truth*) for the insightful discussions and advice. Finally and most importantly, I wanted to thank you Omar for all the discussions, your support and help during these past years, I really hope we will keep in touch and I wish you all the best for the future!

Un grand merci à Simon Leglaive qui m'a accueillie à l'INRIA plusieurs fois et m'a permis de mieux comprendre les VAE et leurs proba !

Je tiens également à remercier tous ceux qui ont passé un ou plusieurs de mes tests perceptifs. Sans vous ce manuscrit ne serait pas aussi complet !

Je n'oublie pas mes amis, éparpillés un peu partout en France et à l'étranger, ainsi que ma famille au sens large (cousins/cousines, tantes/oncles, "belle-famille") qui ont su par un message, un repas, une après-midi ou un weekend (voire un mariage !) me sortir de ma thèse. Un merci particulier aux *Koupaings* et aux potes de *Fanny et les Garçons*, toujours là pour une rando, un jeu de société ou un bon jam (et dont bon nombre font également partie de la catégorie précédente et ont usé leurs oreilles sur mes tests !), ainsi qu'aux *Zozos*, amis de looooongue date perdus en cours de route mais enfin retrouvés, soutiens de cette fin de thèse et on ne peut plus motivés pour la soutenance (sans tutus mais en costards !!). Je tiens par ailleurs à décerner une mention spéciale à Aneline (Koupine <3 <3), amie co-thésarde pendant un temps puis collègue post-doc par la suite, pour le partage des galères, de la clim, d'un bout de bureau ou d'un repas à la piscine (ça va vraiment me manquer !), mais aussi pour les coups de main avec mes slides, mon manuscrit, ma soutenance, tout ! Que ferais-je sans toi ??

Enfin, j'ai gardé les meilleurs pour la fin, je souhaiterais remercier ma famille. Un énorme merci à mes parents qui voient enfin aboutir (au grand soulagement de ma maman !!) près de 25 ans de scolarité dont plus de 10 en études supérieures. Merci de m'avoir toujours fait confiance, d'avoir accepté et soutenu (parfois avec inquiétude) mes choix, c'est vous qui m'avez permis d'arriver là où je suis aujourd'hui. Merci donc à mes parents et aussi à mon frère et ma sœur, mon beau-frère et ma belle-sœur ainsi qu'à mes deux neveux Hugo et Timothée (le *caganis* arrivé en cours de route) d'être un soutien sans faille, toujours présents que ce soit dans les moments difficiles ou pour fêter les belles choses.

Merci enfin à toi *Cœur*. Merci de partager ma vie et de croire en moi depuis toutes ces années.

Contents

List of Figures xiii

List of Tables xvii

List of Acronyms xix

Introduction 1
Context and objectives . . . 1
Main challenges . . . 3
Manuscript organization . . . 4

1 State-of-the-art on music sound perception and synthesis 5
1.1 Musical timbre perception . . . 6
1.1.1 An ambiguous definition of musical timbre . . . 6
1.1.2 Timbre perception approaches . . . 6
1.1.2.1 Multidimensional scaling . . . 7
1.1.2.2 Qualitative description of timbre . . . 10
1.1.3 Perceptual dimensions analysis . . . 11
1.1.4 Remarks on timbre perception studies . . . 12
1.2 Music sound synthesis . . . 12
1.2.1 Abstract algorithms . . . 13
1.2.2 Processed recordings . . . 14
1.2.3 Spectral modeling . . . 16
1.2.4 Physical modeling . . . 17
1.3 Synthesis using deep machine learning . . . 17
1.3.1 Machine learning and deep neural models . . . 18
1.3.1.1 Machine learning . . . 18
1.3.1.2 Artificial neural networks . . . 19
1.3.2 Deep neural models for image synthesis . . . 20
1.3.3 Deep neural models for sound synthesis . . . 22
1.3.3.1 Autoencoder-based models . . . 23
1.3.3.2 GAN-based models . . . 24
1.4 Conclusion . . . 25

2 Perceptual characterization of timbre 27
2.1 Chosen method . . . 28
2.2 First perceptual test: Free verbalization . . . 28
2.2.1 Participants . . . 29
2.2.2 Stimuli . . . 29

2.2.2.1 Arturia dataset generation . . . 29
2.2.2.2 Samples selection . . . 30
2.2.2.3 Participants stimuli assignment . . . 32
2.2.3 Protocol of the study . . . 34
2.2.4 Results analysis . . . 34
2.2.4.1 Objectives of the analysis . . . 34
2.2.4.2 Results pre-processing . . . 35
2.2.4.3 Semantic proximity analysis . . . 36
2.2.4.4 Obtained semantic categories . . . 39
2.2.4.5 Verbal descriptors selection . . . 40
2.2.5 Conclusion . . . 42
2.3 Second perceptual test: Semantic scales analysis . . . 43
2.3.1 Participants . . . 44
2.3.2 Stimuli . . . 44
2.3.2.1 Training stimuli versus main phase stimuli . . . 44
2.3.2.2 Samples selection . . . 45
2.3.3 Protocol of the study . . . 46
2.3.4 Results analysis . . . 47
2.3.4.1 Objectives of the analysis . . . 47
2.3.4.2 Results pre-processing . . . 47
2.3.4.3 Intra-subject consensus . . . 49
2.3.4.4 Inter-subject consensus . . . 50
2.3.4.5 Final label vectors computation . . . 56
2.3.5 Conclusion . . . 56
2.4 Conclusions and perspectives . . . 57

3 Unsupervised extraction of a high-level control space for audio synthesis 59
3.1 Motivation . . . 60
3.2 Analysis-transformation-synthesis methodology . . . 60
3.2.1 Global methodology . . . 60
3.2.2 AE-based models . . . 62
3.2.2.1 Linear AE: PCA . . . 62
3.2.2.2 AE and deep AE . . . 62
3.2.2.3 Recurrent AE . . . 63
3.2.2.4 Variational AE . . . 63
3.2.3 Data representation . . . 67
3.2.3.1 Representation in the time-frequency domain . . . 67
3.2.3.2 Statistical modeling and implications for VAE training . . . 68
3.2.3.3 Phase spectrogram reconstruction . . . 70
3.3 Comparative study of different AE-based models on two datasets . . . 72
3.3.1 Datasets . . . 73
3.3.1.1 NSynth dataset . . . 73
3.3.1.2 Arturia dataset . . . 73
3.3.2 Data pre-processing . . . 74
3.3.3 AE-based models implementations . . . 74

3.3.4 Experimental results . . . 75
3.3.4.1 Analysis-synthesis . . . 75
3.3.4.2 Cross-correlation of latent dimensions . . . 82
3.3.4.3 AE-based sound morphing . . . 86
3.3.5 Conclusion . . . 89
3.4 Max/MSP prototype . . . 89
3.4.1 Main principle . . . 89
3.4.2 Qualitative observations and comments . . . 91
3.5 Conclusion . . . 91

4 Towards weak supervision of autoencoder models using timbre perception 93
4.1 Motivation . . . 94
4.2 Perceptual regularization methodology . . . 94
4.2.1 Timbre-based regularization methodology . . . 94
4.2.2 Weakly supervised learning . . . 96
4.2.3 Proposed methodology . . . 97
4.2.3.1 2-step learning procedure . . . 97
4.2.3.2 Perceptual regularization metric . . . 98
4.3 Regularizing VAE-based models using perceptually meaningful continuous labels 98
4.3.1 Datasets . . . 98
4.3.2 Data pre-processing . . . 99
4.3.3 Regularized AE-based models implementation . . . 99
4.3.4 Experimental results . . . 99
4.3.4.1 Analysis-synthesis . . . 100
4.3.4.2 Latent space organization . . . 103
4.4 Perceptual evaluation of regularized AE-based models . . . 105
4.4.1 Stimuli . . . 105
4.4.2 Protocol of the perceptual study . . . 106
4.4.3 Results analysis . . . 107
4.4.3.1 Applied methodology . . . 107
4.4.3.2 Results . . . 109
4.4.4 Discussion . . . 111
4.5 Conclusions and perspectives . . . 111

Conclusion and perspectives 113

Appendices 119

A Free verbalization test additional material 121
A.1 Participants . . . 121
A.1.1 Pre-test questionnaire . . . 121
A.1.2 Collected information . . . 122
A.2 Detailed protocol explanations . . . 123
A.3 Manual terms grouping . . . 124

A.4 Final perceptual dimensions . . . 126

B Semantic scale study additional material **133**
B.1 Participants . . . 133
B.1.1 Pre-test questionnaire . . . 133
B.1.2 Collected information . . . 134
B.2 Test protocol . . . 135
B.3 Results analysis . . . 136
B.3.1 Intra-subject consensus analysis . . . 136
B.3.2 Inter-subject consensus clustering analysis . . . 137

C Mathematical additional material **139**
C.1 Probability distributions . . . 139
C.1.1 Gaussian distributions . . . 139
C.1.2 Gamma distribution . . . 139
C.1.3 Poisson distribution . . . 140
C.2 Mathematical developments for the variational inference . . . 140
C.2.1 Variational Lower Bound (VLB) . . . 140
C.2.2 Differentiability of regularization term . . . 140

D AE-based models experiments additional material **143**
D.1 Analysis-synthesis experiments on Arturia dataset . . . 143
D.2 AE-based sound morphing . . . 143
D.2.1 NSynth dataset . . . 143
D.2.2 Arturia dataset . . . 143

E Weak perceptual supervision of the model additional material **149**
E.1 A/B testing detailed protocol explanations . . . 149

F Author publications **151**

Bibliography **169**

Résumé en français **183**

List of Figures

1 Illustration of the main objectives of the thesis. 3

1.1 Representation of the psychoacoustic (bottom-up) approach. Figure taken and translated from [Gaillard 2000]. 7
1.2 Representation of the semioacoustic (top-down) approach. Figure taken and translated from [Gaillard 2000]. 8
1.3 Example of results obtained with MDS method for orchestral instruments – abbreviations for stimulus points: O1, O2 = oboes; C1, C2 = clarinets; X1, X2, X3 = saxophones; EH = English horn; FH = French horn; S1, S2, S3 = strings, TP = trumpet; TM = trombone; FL = flute; BN = bassoon. Figure extracted from [Grey 1977]. 9
1.4 Diagram of the most basic FM synthesis instrument with 2 sinusoidal oscillators: one for the modulator and one for the carrier. Picture taken from [Miranda 2002]. 14
1.5 Representations of different granular synthesis approaches. Figures taken from [Miranda 2002]. 15
1.6 Data flow model of a concatenative synthesis system. Diagram taken from [Schwarz 2006]. 15
1.7 Representation of additive synthesis with 3 sinusoidal oscillators ($M = 3$). Figure extracted from [Miranda 2002]. 16
1.8 Generic waveguide filter instrument. Figure taken from [Miranda 2002]. . . . 17
1.9 An illustration of the structure of an artificial neuron. w_i is the weight associated to the i^{th} input, b is the bias and φ is called activation function, it is usually non-linear. 19
1.10 Structure of a deep neural network (DNN). 20
1.11 General structure of an autoencoder where *enc* represents the *encoder* network, *dec* the *decoder* network and $\mathbf{z}$ is the latent representation of $\mathbf{x}$. The model is trained to minimize the reconstruction error between $\mathbf{x}$ and $\hat{\mathbf{x}}$. Figure inspired from [Goodfellow et al. 2016]. 21
1.12 Diagram of the structure of a GAN. 22
1.13 Example of structure of a VAE/GAN model. Figure extracted from [Larsen et al. 2016]. 22
1.14 WaveNet autoencoder model. Figure extracted from [Engel et al. 2017]. . . . 24

2.1 Example of t-SNE projection of the samples in a 2-dimensional space obtained using our visualization tool. In red are the samples selected for the free verbalization perceptual study. We can see that, thanks to the k-means clustering, the selected samples cover well the acoustic space (within the limits of the selected synthesizer used to generate the dataset). Different shapes and different colors indicate the different clusters obtained with k-means. 32
2.2 Representation of the block design. Sn represents the n^{th} participant passing the test and Bm the m^{th} block of 10 samples. 33
2.3 Histogram of the answers received per sample depending on the number of participants that passed the test in the context of an organization of the samples in blocks. Sn represents the n^{th} participant that described the samples. . . . 33
2.4 Screenshot of the free verbalization test interface. We designed and implemented the test using the Web Audio Evaluation Tool[1][Jillings et al. 2015]. . 34
2.5 3D occurrences matrix example. 37
2.6 Example of a dendrogram. The distance threshold is represented by the orange dotted line, dividing the dataset into 4 clusters and one singleton. 38
2.7 Illustration of the samples distribution for the perceptual test with semantic scales. The dotted arrows represent random selection whereas the plain arrow illustrates the sorted selection of all the stimuli of the dataset. 45
2.8 Screenshot of the interface of the test using semantic scales. Here again, the study was implemented using the Web Audio Evaluation Tool [Jillings et al. 2015]. 48
2.9 Screenshot of free expression test page on semantic scales. 48
2.10 Correlation coefficients for the evaluation of the intra-subject consensus. . . . 50
2.11 Histogram of inter-subject correlation coefficients given the semantic scales. . 52
2.12 Resulting dendrograms for the inter-subject correlation coefficients for each semantic scale. 53
2.13 Participants repartition for the different clusters resulting from the HAC analysis for each scale. 54
2.14 Scatter plot of the standard deviation of the ratings of each sound in function of their corresponding mean value for each cluster depending on the perceptual scale. 55

3.1 Global diagram of the analysis-transformation-synthesis process. 61
3.2 Example of modification of one particular latent dimension trajectory by the users. 61
3.3 General architecture of shallow and deep autoencoders. 63
3.4 Structure of a many-to-many LSTM network for an input sequence $\mathbf{x}$ and a corresponding output sequence $\mathbf{z}$ with the same length, h_t being the internal state of the LSTM layer at time t. 64
3.5 General architecture of a VAE. Grey dotted arrows represent sampling process. 65
3.6 Illustration of the concepts behind the Griffin and Lim algorithm: consistency and iterative framework. These two figures are taken from [Sturmel and Daudet 2011]. 70

3.7 Phase unwrapping processing of spectral peaks. Both figures are taken from [Magron 2016]. 72
3.8 Reconstruction error (RMSE in dB) obtained with different AE-based models as a function of latent space dimension (trained on the NSynth dataset). . . . 77
3.9 PEMO-Q measures obtained with different AE-based models as a function of latent space dimension (trained on the NSynth dataset). 78
3.10 Reconstruction error (RMSE in dB) obtained with different AE-based models as a function of latent space dimension (trained on the Arturia dataset). . . . 80
3.11 PEMO-Q measures obtained with different AE-based models as a function of latent space dimension (trained on the Arturia dataset). 81
3.12 Reconstruction error (RMSE in dB) obtained with different AE-based models as a function of latent space dimension (trained on the Arturia dataset computed with smaller temporal windows - 1024-point STFT). 83
3.13 PEMO-Q measures obtained with different AE-based models as a function of latent space dimension (trained on the Arturia dataset computed with smaller temporal windows - 1024-point STFT). 84
3.14 Correlation matrices of the latent dimensions (average absolute correlation coefficients) for PCA, DAE, LSTM-AE and VAEs trained on the NSynth dataset. 85
3.15 Correlation matrices of the latent dimensions (average absolute correlation coefficients) for PCA, DAE, LSTM-AE and VAEs trained on the Arturia dataset. 85
3.16 Examples of decoded magnitude spectrograms after sound interpolation of 2 NSynth samples (top) in the latent space using respectively PCA (2nd row), DAE (3rd row), LSTM-AE (4th row) and VAE (bottom). A more detailed version of the figure can be found in Appendix D.2.1 or at http://www.gipsa-lab.fr/~fanny.roche/PhD_thesis.html. 87
3.17 Examples of decoded magnitude spectrograms after sound interpolation of 2 Arturia samples (top) in the latent space using respectively PCA (2nd row), DAE (3rd row), LSTM-AE (4th row) and VAE (bottom). A more detailed version of the figure can be found in Appendix D.2.2 or at http://www.gipsa-lab.fr/~fanny.roche/PhD_thesis.html. 88
3.18 Graphical user interface of the non real-time Max/MSP prototype. 90

4.1 Diagram of the perceptually-regularized VAE extracted from [Esling et al. 2018b]. 95
4.2 Reconstruction error (RMSE in dB) obtained with different versions of the perceptually-regularized VAE. 101
4.3 PEMO-Q measures obtained with different versions of the perceptually-regularized VAE. 102
4.4 2-dimensional projection of the 8 first dimensions of the latent space encoded by the VAE models using the t-SNE algorithm on the labeled dataset. On the left column are presented the projections corresponding to the classic VAE and the right column the perceptually-regularized VAE. From the top to the bottom are represented the models trained with respectively 1, 2 and 3 iterations of the 2-step learning procedure. On this example the color represents the mean ratings given by the participants on the "Métallique" scale. 104

4.5 Screenshot of the A/B perceptual test. Similarly as before, this test was implemented using the Web Audio Evaluation Tool [Jillings et al. 2015]. *Note: Here the B sound has been selected.* 108
4.6 Results of the A/B testing analysis for the different scales and the 2 datasets (*orig* corresponds to the training annotated dataset and *test* to the test samples). The red line indicates chance. For each scale and the two datasets, the mean answer is illustrated with a 95% confidence interval. 109

A.1 Repartition of the questionnaire answers collected during the free verbalization test in percentage of the 101 participants. The different possible answers are given as labels of the graphs. 122

B.1 Repartition of the questionnaire answers collected during the test with semantic scales in percentage of the 71 participants. The different possible answers are given as labels of the graphs. 134
B.2 Thresholding to select reliable subjects from the histogram of the intra-subject correlation coefficients for each semantic scale. 136
B.3 Inter-subject correlation coefficients matrices given the semantic scales. 137

D.1 Reconstruction error (RMSE in dB) obtained with PCA, AE, DAEs (with and without layer-wise training) and LSTM-AE as a function of latent space dimension (trained on the Arturia dataset computed with smaller temporal windows - 1024-point STFT). 143
D.2 Examples of decoded magnitude spectrograms after sound interpolation of 2 NSynth samples in the latent space using respectively PCA (left) and DAE (right) 144
D.3 Examples of decoded magnitude spectrograms after sound interpolation of 2 NSynth samples in the latent space using respectively LSTM-AE (left) and VAE (right). 145
D.4 Examples of decoded magnitude spectrograms after sound interpolation of 2 Arturia samples in the latent space using respectively PCA (left) and DAE (right) 146
D.5 Examples of decoded magnitude spectrograms after sound interpolation of 2 Arturia samples in the latent space using respectively LSTM-AE (left) and VAE (right). 147

List of Tables

2.1 Table of the 5 classes maximizing the frequency and transversality criteria. . 39
2.2 Most frequent terms given by subjects for the free verbalization test. The transversality measure corresponds to the number of different participants that used the term. The terms are sorted by the total number of occurrences. . . . 41
2.3 Table of the final labels for the scales. 43
2.4 Table of the number of selected participants after the intra-subject consensus analysis for each scale. 51

4.1 Table of the results of the analysis for the different scales. The second column represents the results of the *anova* test aiming at evidencing a potential impact of the "dataset" factor on the answers (the *Chisq(1)* and p-value are reported), the third column presents the results of the comparison with chance (z and p-value) and the last column presents the results of the AUC – ROC curve measures. 110

A.1 Groups of terms sharing the same lexical root for the free verbalization test. . 124
A.2 Table of perceptual classes for the free verbalization test. 126

List of Acronyms

AE	Autoencoder
ANN	Artificial neural network
API	Application programming interface
AUC	Area under the curve
DAE	Deep autoencoder
DBN	Deep belief network
DCT	Discrete cosine transform
DNN	Deep neural network
FM	Frequency modulation
G&L	Griffin and Lim
GAN	Generative adversarial network
GUI	Graphical user interface
HAC	Hierarchical agglomerative clustering
IS	Itakura-Saito
ISTFT	Inverse short-time Fourier transform
KL	Kullback-Leibler
LSTM	Long short-term memory
MDS	Multidimensional scaling
MFCC	Mel-frequency cepstral coefficients
MIDI	Musical instrument digital interface
ML	Maximum-likelihood
MSE	Mean squared error
MUMS	McGill University Master Samples
NMF	Nonnegative matrix factorization
NN	Neural network
NSGT	Non-stationary Gabor transform

OLA Overlap-add

OSC Open sound control

PCA Principal component analysis

PDF Probability density function

RMSE Root-mean squared error

RNN Recurrent neural network

SD Semantic differential

STFT Short-time Fourier transform

VAE Variational autoencoder

VLB Variational lower bound

ZCR Zero-crossing rate

Introduction

Context and objectives

Humans have always used sounds to communicate, be it information or emotions, and tried to master their production by learning to talk or sing, or by crafting musical instruments, constantly looking for new sonorities and new controls. In the end of the nineteenth century, thanks to the huge improvements in electricity, electronics and also later in digital technologies, new techniques allowing to generate sound signals emerged giving birth to audio synthesis. In particular, music sound synthesis substantially changed the face of the music history thanks to its ability to generate new timbres commonly called "electronic" sounds that acoustic instruments could not produce and that are nowadays (almost) omnipresent in the music industry.

One of the first musical instruments embedding one of these techniques was the Telharmonium, an early electrical organ, created in 1896 by Thaddeus Cahill that used tonewheels to generate musical sounds using electric signals. During the beginning of the twentieth century audio synthesis experienced great scientific advances giving birth to several new musical instruments like the Theremin (1920) or the Ondes Martenot (1928) but we had to wait until the late 1960s to see commercialized synthesizers on the market and musicians using them for composing. Since that time, synthesizers have seen many evolutions following the progress in computing performances and have started to be proposed as software gear, thus becoming more affordable and more widespread in the musicians community. Thanks to the huge improvements in computing power, synthesizers were able to provide musicians with more and more possibilities to create sounds. But due to the complexity of audio synthesis methods used by these synthesizers (that will be presented in section 1.2), this comes with a large set of low-level control parameters whose manipulation may require some expert knowledge about generative processes. One of the main challenge of the synthesizer market and the research in sound synthesis nowadays thus lies in proposing new forms of synthesis allowing the creation of new sonorities while offering musicians more intuitive control parameters to help them reach the sounds they truly desire more easily. This is in this context that this PhD thesis, which is part of a collaboration[2] between GIPSA-Lab[3] and Arturia[4], took place. Indeed, one of the main concerns of Arturia being to provide the musicians with intuitive, easy-to-use and innovative instruments, it seemed natural to investigate what could be the most intuitive control parameters allowing both amateurs to have easily access to such powerful instruments and professionals to develop a new experience in creation and sound exploration.

[2]This work was supported by the ANRT in the framework of a CIFRE PhD program.

[3]Signal processing laboratory based in Grenoble.

[4]Company based in Montbonnot Saint-Martin (proximity of Grenoble) which produces musical equipments and in particular hardware and software synthesizers (https://www.arturia.com/).

Inspired by the recent advancements in image generation using machine learning allowing for an intuitive control of the synthesis [Reed et al. 2016; Chen et al. 2018; Xian et al. 2018; Zhang et al. 2018], we got interested in finding a way to control music sound synthesis using high-level parameters with a perceptual relevance. Indeed, just like controlling image synthesis by sketching [Xian et al. 2018] or by describing the wanted image textually [Reed et al. 2016; Zhang et al. 2018], having a reduced number of perceptually meaningful controls for the music sound synthesizer would allow every musician to explore and create sounds without the need of a substantial expertise in sound design. Moreover, this would bring expert users new patterns to help them finding the sound they really seek, which could be a tedious task using classic synthesis methods.

A few studies have investigated the use of high-level intuitive parameters to generate sounds. In the very specific context of everyday sounds generation, researchers have focused on parameters such as the perceived material for synthesizing impact sounds [Aramaki et al. 2010] or the type of interaction (scratching, rolling, rubbing) for the synthesis of continuous-interaction sounds [Conan et al. 2014] for example. But in a more general context, these descriptors may not be relevant anymore or at least neither sufficient nor spontaneous to characterize sonic textures. In this case, to control music sounds synthesis, the most intuitive high-level parameters that seem to be adapted are adjectives commonly used by musicians to describe musical timbre such as *bright, woody* or *harsh* [Gounaropoulos and Johnson 2006; Howard et al. 2007; Kreković et al. 2016]. In these few preliminary studies, the datasets used to extract the verbal descriptors that will then serve as synthesis parameters are exclusively constituted of sounds produced by acoustic instruments. In the context of this thesis, the nature of the samples is slightly different as we focus on purely synthetic sounds. The commonly used vocabulary will then probably differ from the verbal descriptors highlighted by these studies but, as language is the easiest manner to express our feelings and perception, using terms usually employed by musicians still appears to us as a promising research line. Plus, by using these controls in combination with a machine learning model, this would allow us to map our perceptual space to a representation space hence enabling to generate interesting hybrid sounds. For example, it would be possible to synthesize a sound which is 40% *metallic* + 20% *aggressive* + 40% *evolving*. This would thus provide unlimited possibilities for musicians to explore sound material.

The main objective of this thesis can therefore be summarized as developing and evaluating new data-driven machine learning methods for music sound synthesis where control parameters are high-level descriptors related to the specific lexicon of musicians to describe music timbre in the case of synthesizer sounds, and which allow to generate new high-quality sounds. As a first step towards this very challenging objective, we got interested into synthesizing new timbres by modifying sounds according to these perceptual high-level controls as illustrated in Figure 1. To achieve this goal, we had to face different challenges and research problems involving very diverse fields of expertise going from cognitive science to machine learning through digital signal processing. They are depicted in Figure 1 and will be described in the following section.

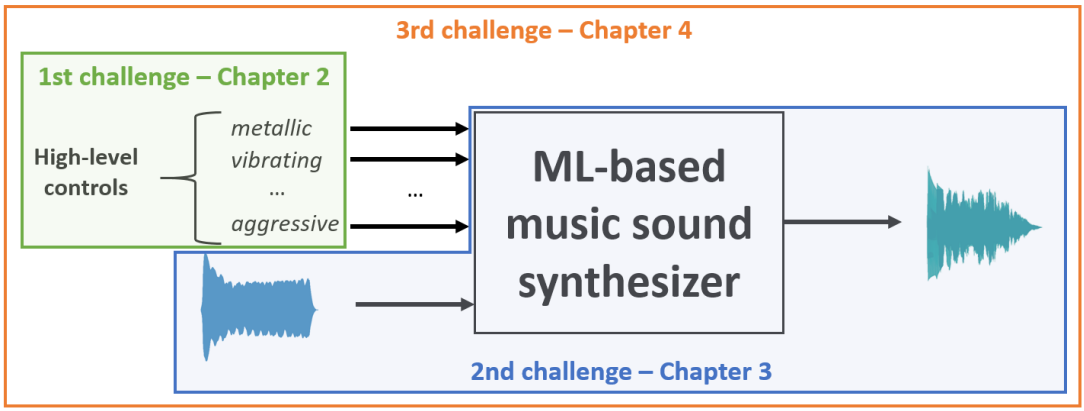


Figure 1: Illustration of the main objectives of the thesis.

Main challenges

To begin with, in order to be able to use verbal descriptors of synthetic timbres as control parameters of a synthesis algorithm, it is necessary to define clearly which are these descriptors and how they are used. Most of studies investigating musical timbre through the way people perceive and talk about it focus on datasets of acoustic instruments or synthetic sounds aiming at reproducing orchestral instruments. To our knowledge, no previous studies defined precisely what are the most frequent and consensual terms to describe synthetic sounds that do not imitate orchestral instruments although some already listed verbal descriptors to characterize such sounds [Lichte 1941; von Bismarck 1974; Zacharakis 2013]. The first main challenge we had to deal with was thus to characterize musical synthetic timbre perceptually by gathering terms which are both commonly and consensually used by musicians.

Once the terms defined, another important issue was to find a suited machine learning method able to realize a mapping between the sounds space and a high-level representation space with interesting interpolation and extrapolation properties. This would allow us to navigate through the space smoothly, moving from one sound to another without discontinuity, and to explore sonic timbres beyond the limits of the training database in order to create new sounds with high quality. The main idea was then to compare the dimensions extracted by the model with the verbal descriptors used to perceptually characterize synthetic timbres and evaluate how they relate, or not, and if we could directly use this new representation space as control space of our new synthesizer.

As there were few chances that this representation space automatically made sense perceptually, the final key point of this thesis was to relate this high-level dimensional space extracted by the deep learning algorithm to the verbal descriptors. By applying conditioning or weak supervision to the neural model, the objective was to provide human knowledge to the algorithm in order to eventually get relevant dimensions for a perceptual control of the synthesis.

Manuscript organization

The manuscript is organized as follows:

Chapter 1 – State-of-the-art on music sound perception and synthesis This chapter will go through the state-of-the-art in music sound perception and synthesis. We will first present what is musical timbre and how researchers investigated its perception and attempted to characterize it using different sets of perceptual dimensions. Then we will introduce current approaches used for music sound generation and in particular we will present the latest advancements in sound synthesis using deep machine learning methods.

Chapter 2 – Perceptual characterization of timbre This chapter will describe the method we chose and the perceptual studies we carried out to evidence terms that are frequently and consensually used by musicians to describe synthesizer sounds. The results we obtained will be presented and discussed.

Chapter 3 – Unsupervised extraction of a high-level control space for audio synthesis This chapter will detail the different unsupervised machine learning methods we explored and compared to extract a usable representation space and generate sounds with good quality. The realized experiments and prototypes together with their results will also be presented and discussed.

Chapter 4 – Towards weak supervision of autoencoder models using timbre perception Lastly, this chapter will introduce weak supervision and regularization of the neural models, and then present and discuss preliminary results towards a perceptually meaningful control of music sound synthesis.

Finally, this manuscript will be concluded by a summary of the different achievements and contributions of this thesis and a discussion on the possible perspectives for future work.

It is interesting to note that this work, which consists in mapping sounds to perceptually meaningful high-level control parameters for synthesis, is closely related to other research topics such as expressive speech synthesis which investigates a mapping between sound representations (e.g. audio descriptors) and emotional descriptors [d'Alessandro and Doval 2003; Akuzawa et al. 2018] or other fields of study that explore the use of alternative types of high-level control parameters for audio synthesis like gesture as proposed in [d'Alessandro et al. 2006a; d'Alessandro et al. 2006b; Feugère et al. 2017].

Chapter 1

State-of-the-art on music sound perception and synthesis

Contents

1.1 Musical timbre perception . . . 6
1.1.1 An ambiguous definition of musical timbre . . . 6
1.1.2 Timbre perception approaches . . . 6
1.1.2.1 Multidimensional scaling . . . 7
1.1.2.2 Qualitative description of timbre . . . 10
1.1.3 Perceptual dimensions analysis . . . 11
1.1.4 Remarks on timbre perception studies . . . 12
1.2 Music sound synthesis . . . 12
1.2.1 Abstract algorithms . . . 13
1.2.2 Processed recordings . . . 14
1.2.3 Spectral modeling . . . 16
1.2.4 Physical modeling . . . 17
1.3 Synthesis using deep machine learning . . . 17
1.3.1 Machine learning and deep neural models . . . 18
1.3.1.1 Machine learning . . . 18
1.3.1.2 Artificial neural networks . . . 19
1.3.2 Deep neural models for image synthesis . . . 20
1.3.3 Deep neural models for sound synthesis . . . 22
1.3.3.1 Autoencoder-based models . . . 23
1.3.3.2 GAN-based models . . . 24
1.4 Conclusion . . . 25

The objective of the thesis being developing a data-driven machine learning method for music sound synthesis controlled by perceptually meaningful verbal descriptors, many notions from diverse fields of expertise are involved such as musical timbre, perception, audio synthesis and deep machine learning models. In this chapter we will introduce these notions and present the related literature.

1.1 Musical timbre perception

1.1.1 An ambiguous definition of musical timbre

The perception of musical timbre has been of interest for research for more than 150 years [von Helmholtz 1875] but timbre is still a sound attribute that is not objectively defined. One of the main reasons for this is that, unlike pitch or loudness for instance, it is not directly linked to a single acoustic dimension but rather is a multidimensional perceptual attribute [Risset and Wessel 1982; Krumhansl 1989; Caclin et al. 2005].

One of the official definitions we can find is given by the American National Standards Institute (ANSI) [American National Standards Institute 1973] and states : "*Timbre is that attribute of auditory sensation in terms of which a subject can judge that two sounds similarly presented and having the same loudness and pitch are dissimilar*". But according to [Donnadieu 2007], this definition is not satisfactory enough as timbre refers to different concepts depending on the level of analysis. Indeed, as explained in [Traube 2004] or [Castellengo 2015], the notion of timbre can be considered at two different levels. First it can be considered at a macroscopic level describing/identifying the "source" of the sound (its name or some of its attributes such as its material or the excitation mode, or whether it is alive or not for example). We refer to this as the *causal timbre*. Timbre can also be considered related to a microscopic view and allows to describe the subtle variations in sound qualities, for example discriminating the different playing modes of a same musical instrument. This is referred to as *qualitative timbre* (or also *intra-instrumental timbre* in some publications [Lavoie 2013]).

Moreover, there exists a hierarchy between these modes of timbre perception: sounds cannot be qualified if they have not been preliminary identified or categorized [Castellengo 2015; Castellengo and Dubois 2005].

In the literature, several methods have been adapted to study timbre perception and its acoustic correlates. Some of them will be presented in the next section.

1.1.2 Timbre perception approaches

Two main approaches in timbre perception exist: the psychoacoustic approach (bottom-up) and the semioacoustic approach (top-down).

The psychoacoustic approach starts with the sound acoustics and physics and connect them to the listener's perception, see Figure 1.1. It consists in first trying to analyze which are the acoustic dimensions that are important to characterize a sound in an objective way and then link them to the human perception. For example we can, by (analysis-)synthesis, vary or transform some acoustic features of sound samples, and investigate how this variation in the physical world affects the listener's perception (his or her evaluation of some perceptual dimensions, or even the labeling and relevance of these perceptual dimensions). This approach is well-adapted to both causal and qualitative timbre perception.

Conversely, the semioacoustic approach goes from the listener's perception to the sound characterization, see Figure 1.2. The main objectives are first to identify relevant criteria/dimensions and verbal descriptors on which is based the human perception and then to find their acoustic correlates. For example, this time it would consist in exploring the lexicon used by listeners to describe the timbre of a specific category of sounds, its semantic network,

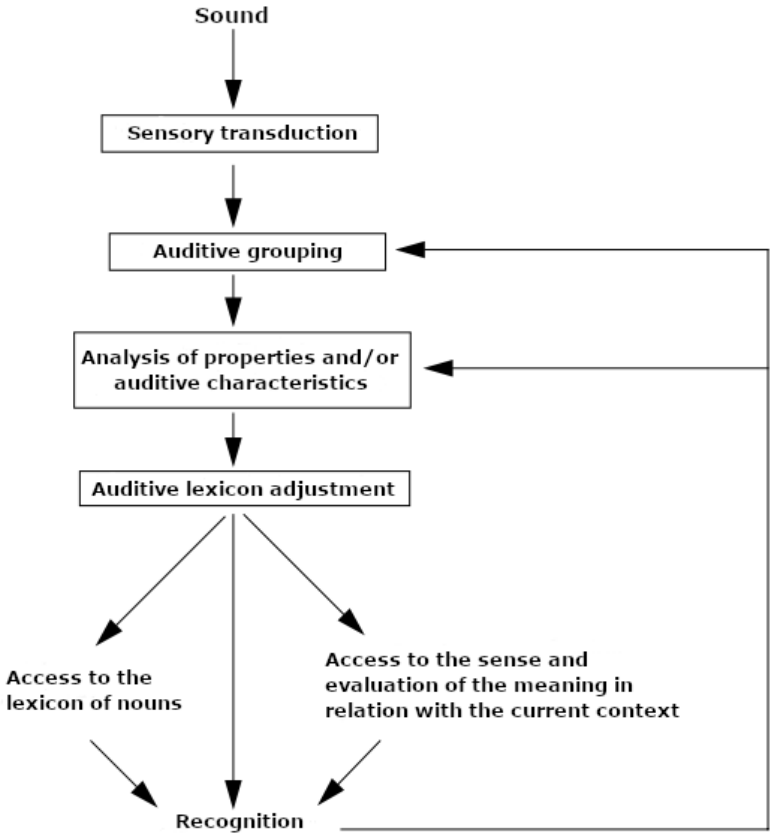


Figure 1.1: Representation of the psychoacoustic (bottom-up) approach. Figure taken and translated from [Gaillard 2000].

and then trying to find acoustic dimensions (isolated dimensions or combinations of them) that correlate with these perceptual dimensions. This approach is more adapted to qualitative timbre perception.

Although the two methods seem in opposition, they can actually be used jointly. Thus, several studies first adopted a semioacoustic approach to identify relevant perceptual dimensions and verbal descriptors for a sound category, relate them to acoustic cues and then use a psychoacoustic approach in a second step to validate the relevance of these perceptual dimensions and associated verbal descriptors, and establish in more detail the relationship between acoustic variations and variations in sensation along these perceptual dimensions.

In the following, some examples of methods using these two approaches for timbre perception focusing on the causal timbre or the qualitative timbre will be presented.

1.1.2.1 Multidimensional scaling

One of the most frequently used methods to study the timbre of musical instruments following a psychoacoustic approach, is the multidimensional scaling (MDS) [McAdams and Giordano 2009]. Listeners are asked to rate all pairs of stimuli within a dataset. Then a multidimensional scaling algorithm is used to map the subjective distance relationships into a geometric space with a limited number of perceptual dimensions, see example in Figure 1.3.

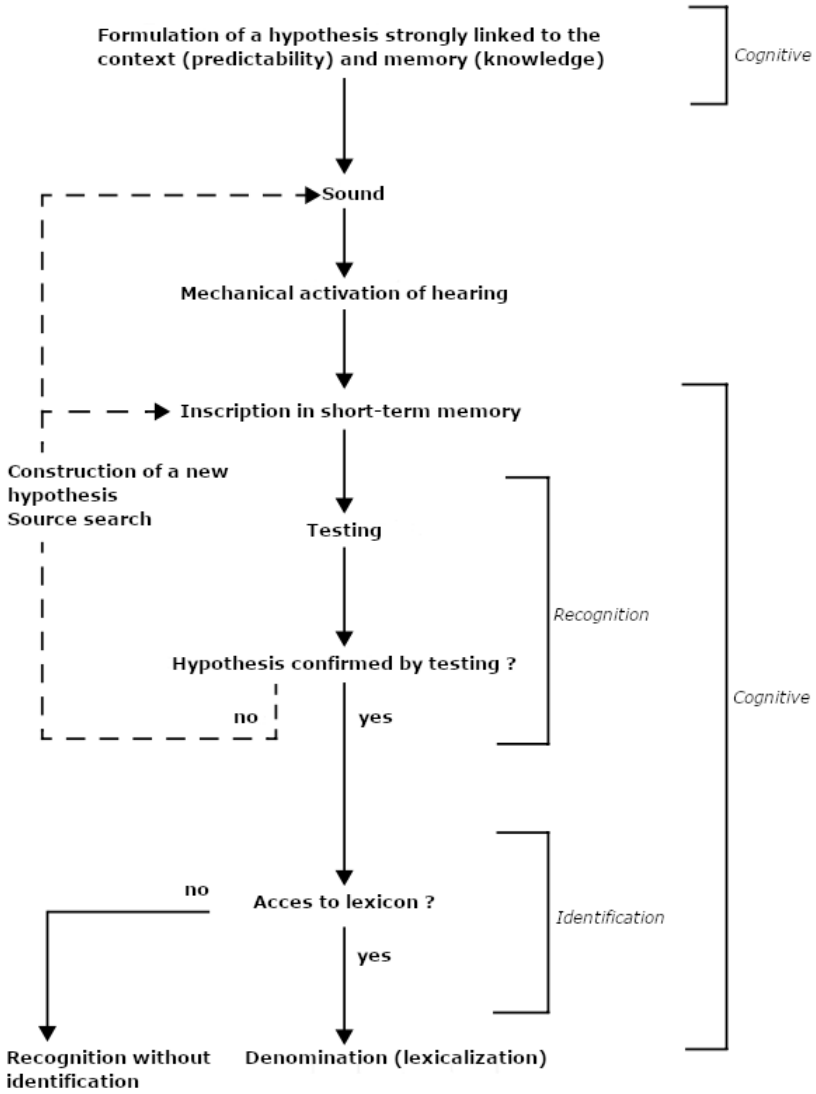


Figure 1.2: Representation of the semioacoustic (top-down) approach. Figure taken and translated from [Gaillard 2000].

Such method allowed to establish a set of salient perceptual dimensions on which listeners rely to categorize, identify and distinguish sounds of different musical instruments. Finally, acoustic analyses are conducted on the dataset to relate these perceptual dimensions to an acoustic descriptor (isolated parameter or combination of several parameters).

The first study applying such a method on musical instruments sounds was [Grey 1977]. The author used a database of 16 synthesized sound samples imitating orchestral instruments (e.g. clarinet, oboe, French horn) all equalized in pitch, loudness and duration. 20 listeners rated all pairs of stimuli using a discrete scale going from 1 (very dissimilar) to 30 (very similar). They identified 3 main perceptual dimensions that distinguish musical timbre, related to (i) spectral energy distribution, (ii) the synchronicity of high-frequency transients and (iii) the presence of low-amplitude, high-frequency energy at sound onset. Then followed many

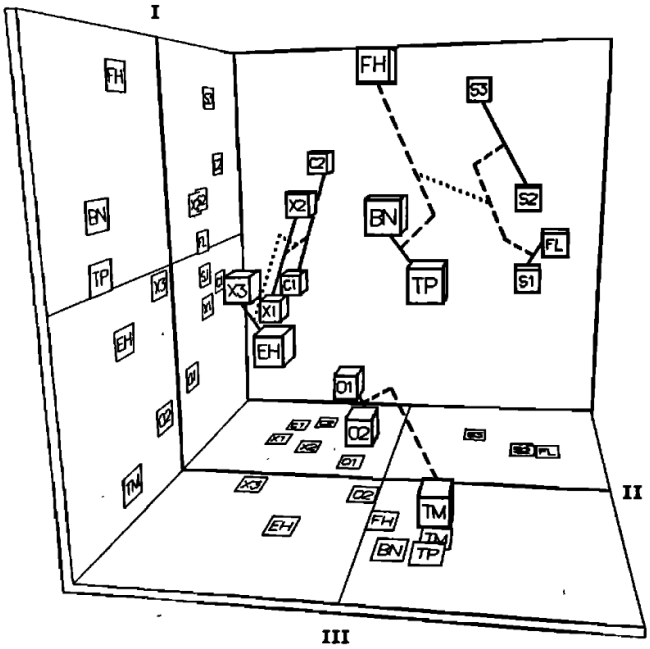


Figure 1.3: Example of results obtained with MDS method for orchestral instruments – abbreviations for stimulus points: O1, O2 = oboes; C1, C2 = clarinets; X1, X2, X3 = saxophones; EH = English horn; FH = French horn; S1, S2, S3 = strings, TP = trumpet; TM = trombone; FL = flute; BN = bassoon. Figure extracted from [Grey 1977].

other studies on musical timbre using the same method on different datasets and different similarity scales:

[Iverson and Krumhansl 1993] conducted a study on 16 samples of digitally recorded orchestral instruments extracted from the McGill University Master Samples (MUMS) Library [Opolko and Wapnick 1987]. They created 3 different stimuli sets by using respectively the whole samples, keeping only the tones onsets, and removing only the onsets. These 3 different datasets were then presented separately to participants for similarity ratings on a continuous scale going from *a little* to *a lot*. They distinguished musical timbre in a 2-dimensional perceptual space, whose first dimension corresponded to dynamic attributes while the second dimension was related to static spectral attributes.

[McAdams et al. 1995] studied a database of 18 digitally synthesized instruments developed by [Wessel 1987] including both traditional orchestral instruments and hybrids (e.g. guitarnet: mix of a guitar and a clarinet) and used discrete scales between 1 (very similar) and 9 (very dissimilar). They identified 3 perceptual dimensions related, in the acoustic domain, to: (i) rise time, (ii) spectral centroid and (iii) degree of spectral variations over time.

In [Lakatos 2000], the author conducted his study on samples of harmonic and percussive musical instruments from the MUMS dataset and used continuous scales.

Despite comparable approaches and sound samples, these different studies did not identify exactly the same perceptual dimensions and underlying acoustic parameters which organize the perception of musical timbre.

1.1.2.2 Qualitative description of timbre

There also exist several methods involving verbal qualitative descriptions to study and characterize the musical timbre as language is the first way to express how we perceive sounds and things in general [Castellengo 2015].

The first verbal method for qualifying timbre is free verbalization. This method consists in asking participants to freely describe verbally the stimulus they are listening to [Traube 2004; Garnier et al. 2007]. The description can be made orally during an interview (with an experimenter) or in written form. The stimuli may also be described using imitations or onomatopoeia as in [Lemaitre and Rocchesso 2014]. Sounds can be presented by pairs, as for MDS approach, and so the participants are asked to describe verbally the differences or similarities between them [Faure 2000], or they can be presented in an isolated way, and then the task is to describe the sound itself [Cance and Dubois 2015]. In [Faure 2000] 12 samples of digitally synthesized instruments from [Wessel 1987] were presented to musicians (professionals and amateurs) and non-musicians. In this study, the participants were asked to rate the dissimilarity between the pairs and describe the differences and similarities with as much detail as possible. [Traube 2004] focuses on guitar sounds and adjectives that best describe the timbre nuances produced by the instrument. The participants were professional guitarists and the task was to give 10 adjectives to describe isolated stimuli. In [Cance and Dubois 2015] the authors also conducted a free verbalization test on a guitar sound played backwards, with 3 different groups of students. This free verbalization method was also applied to the perception of operatic voices quality [Garnier et al. 2007] using a corpus of male singers presented to singing teachers from prestigious music schools.

A second method that can be used for qualification of timbre is free categorization. This method is based on two assumptions [Rosch 1999]. The first is that we do not perceive all the objects of the world on the same level but through the filter of our cognitive representations of the world (top-down approach). The second is that these cognitive representations of the world are not copies of all the encountered objects and experiences (exemplar theory) but abstract and organized representations of the different categories and subcategories of all these objects, of the prototypes of these categories, and of the dimensions that discriminate them. For example, mammals and fishes are two cognitive categories of "animals" and distinguish themselves by the presence or absence of legs. Dogs are the prototype of the mammals category whereas whales often don't belong to that cognitive category although it is theoretically a mammal from a biological point of view. For this method, participants are asked to freely group stimuli with respect to similarity and dissimilarity. The participants can freely decide the number of clusters, their size and the gathering criteria. They are often asked afterward to explain verbally their choices and the criteria they used for creating the clusters and the dimensions that distinguish the groups or the similarities within each one. Studies using this method have been conducted on musical sounds: in [Gaillard 2000] musicians are asked to group samples recorded from a steeldrum and in [Bensa et al. 2004] musicians (pianists or not) and non-musicians were asked to cluster piano note excerpts according to similarity. This method has also been applied on speech [Ehrette 2004] in the context of server voices (corpus of sentences recorded from professional speakers were presented to potential users) or on domestic sounds [Guyot 1996].

Another method, which can possibly be used as a complement to the two previous methods

as in [Faure 2000] or [Ehrette 2004], is the semantic differential (SD) method and belongs to the psychoacoustic approaches. Authors first select semantic descriptors from a former free verbalization or free categorization test (the most used adjectives for instance) or from other studies as in [Zacharakis et al. 2014]. The task is then for the participants to rate each stimulus on scales associated with the selected verbal descriptors. Just like for MDS, the used scales can be discrete or continuous, and they can be unipolar (e.g. going from *bright* to *not bright*) or bipolar (e.g. from *open* to *closed*).

1.1.3 Perceptual dimensions analysis

Even if these approaches are very different, the first objective remains the same: to find the most salient perceptual dimensions. For the multidimensional scaling approach (Section 1.1.2.1), this space corresponds directly to the output of the MDS algorithm. For other approaches, it can be done using different techniques such as the analysis of the verbalization using linguistic approaches (e.g. isolating verbal units and counting occurrences) [Faure 2000; Ehrette 2004; Garnier et al. 2007], factorial analysis on the semantic scales [Faure 2000; Zacharakis et al. 2014] or additive similarity tree analysis in the case of free categorization for example [Gaillard 2000; Ehrette 2004].

The perceptual space highlighted by the analysis can then be used in many different ways. For example, it can be used for computing perceptual distances which can serve as additional regularization constraints for a variational autoencoder [Esling et al. 2018b] (see more details in Section 1.3.3.1). Another application can be to relate dimensions of the perceptual space to parameters defined by experts of the field like [Ehrette 2004] did for voice emotions or [Traube 2004] for guitar gestures. But one of the most used application of these methods is to find interpretations of the main dimensions in terms of acoustic correlates to better understand the notion of timbre (see nearly all the studies in Section 1.1.2.1 or [Zacharakis et al. 2014]).

Two main perceptual dimensions and underlying acoustic correlates emerge from these studies and Schaeffer's reflexions [Schaeffer 1966]. The first criterion/dimension contains everything that is linked to the temporal envelope of the sound (called *masse* criterion – "mass" – by Schaeffer [Schaeffer 1966]), such as the mode of excitation (sustained, impulse or repeated like scratching, grinding or rolling). And the second corresponds to the spectral content, thus more related to the size, the material or even the shape of the instrument and referred to as *facture* ("treatment") in [Schaeffer 1966]. More precise audio correlates of these two main perceptual dimensions have been suggested in the literature. Thus, almost all MDS studies on causal timbre acknowledge *attack time* and *spectral centroid* to be the most important parameters [Grey 1977; McAdams et al. 1995]. However, studies differ with regard to a third perceptual dimension, which would be related to *spectral flux*, *spectral deviation*, *amplitude envelope*, *spectral density* or even *noisiness* or *pitch strength* (see [Peeters et al. 2011] for more details about audio descriptors).

Acoustic correlates of qualitative timbre also vary among studies, and appear to strongly depend on the sound category that are dealt with. For example, in the case of operatic voices [Garnier et al. 2007] the frequencies of the first two formants correlate with perceptual dimensions such as *yawned*, *dark/bright* or *nasal* and the spectral balance is related to the *bright/dull* dimension. In the very different context of orchestral instruments [Zacharakis et al. 2014] the three main acoustic cues seem to be the energy distribution of harmonic partials

which correlate with the *texture* (*harshness/roughness*), the inharmonicity and spectral centroid variations which exhibit a strong correlation with the *luminance* (*brilliance/sharpness*), and finally the fundamental frequency which seems to affect the *mass* (*thickness/lightness*).

1.1.4 Remarks on timbre perception studies

A number of limitations exist for these studies. Indeed, dealing with subjective objects as timbre, the obtained results are very sensitive to the participants cultural and linguistic background. It is thus very important to choose wisely the descriptors for semantic differential methods and to be careful with the native language of participants. There also exists a dependency of the results with regards to the context since a same term can have very different meanings depending on the context in which it is used. For example the french word "*clair*" can be used to describe a sound that is well-defined, precise as opposition to faint, imprecise; or it can be used to describe a bright sound, resonant, opposed to a dull sound; or even describe a more metallic or percussive aspect [Cheminée et al. 2005]. Another limitation regarding perceptual studies is the inter-individual variability: perceptual dimensions, verbal descriptors and use of scales may vary with personality, personal background and/or expertise of the participant, or even with the category of sounds. It is therefore very important to study the inter-listener agreement [Kreiman and Gerratt 1998]. All these limitations have to be taken into account when designing and/or analyzing a perceptual study.

The duration of the audio samples is also an important aspect that has to be wisely chosen depending on the purpose of the study. Indeed, since the task of perceptually describing, naming or even comparing stimuli is intrinsically linked to short-term and working memory, the longer the duration the more the listener will focus on the macroscopic aspect of timbre. If the study deals with qualitative timbre, the audio stimuli presented will thus have to be rather short to allow participants to concentrate on microscopic aspects of timbre.

A final remark that can be done is that the vast majority of the timbre perception studies have been realized on clearly definite sound sources such as a dataset of orchestral instruments (or synthesized versions of them) [Grey 1977; Iverson and Krumhansl 1993; McAdams et al. 1995; Lakatos 2000; Faure 2000; Zacharakis et al. 2014; Cance and Dubois 2015], a dataset of a particular musical instrument: guitar [Traube 2004], steeldrum [Gaillard 2000], piano [Bensa et al. 2004]; noise [Guyot 1996; Cance and Dubois 2015] or even voice [Ehrette 2004; Garnier et al. 2007]. To our knowledge, very few studies have been realized on qualifying the timbre of purely synthesized sounds [Lichte 1941; von Bismarck 1974; Miller and Carterette 1975; Plomp 1976; Samson et al. 1997; Zacharakis 2013].

1.2 Music sound synthesis

Simultaneously with attempting to understand the true nature of the musical timbre and how we perceive it, researchers have also investigated means to reproduce electrically and/or digitally already existing timbres along with creating new ones. Indeed, as stated in the introduction of this manuscript, since the late nineteenth century audio synthesis has been an ever growing field of research always experimenting new techniques in order to provide users, and in particular musicians, new ways to explore sound material and create

new timbres. Nowadays, the most used techniques of audio synthesis embedded into hardware and/or software synthesizers can be organized into four categories according to [Smith 1991]:

- abstract algorithms,
- processed recordings,
- spectral modeling,
- physical modeling.

In the next sections we will present the different categories of sound synthesis methods through their main principles and illustrate them using some examples. The list of presented synthesis techniques is absolutely not exhaustive and its purpose is only to give an overview of the most applied methodologies. For more examples and/or details about the different techniques the reader is referred to [Roads 1996], [Miranda 2002] or [Russ 2008].

1.2.1 Abstract algorithms

Abstract sound synthesis techniques use mathematical functions and algorithms to generate sounds without a direct physical interpretation [Kleimola 2013]. They allow to efficiently create very interesting sounds that could not be achieved physically. In particular, they require low memory and offer a dynamic control of the produced spectrum using very few parameters, although not so easy to understand and manipulate [Miranda 2002].

One of the most famous abstract synthesis techniques is the frequency modulation (FM) synthesis. It has been developed in the 1970s [Chowning 1973] and was the basis of some of the early digital synthesizers like the well-known Yamaha DX7. The basic principle of FM synthesis is to use a modulator signal to modify the frequency of the carrier audio signal that is played, see Figure 1.4. Thus, if we take the easiest example where the two oscillators generate sine waves, we obtain the resulting formula:

$$y(n) = a_c \ \sin\Big(2\pi(f_c + d \ \sin(2\pi f_m \ n)) \ n\Big),$$

where n is the audio sample and y is the output of the synthesizer, f_c and f_m being the respective frequencies of oscillation of the carrier and the modulator, a_c the amplitude of the carrier and d the modulation index. This method allows to generate signals with a rich harmonic content from a very simple waveform and few parameters.

Another example of abstract technique is waveshaping synthesis, also called non-linear distortion. It consists in passing the original signal through a chosen non-linear transfer function to distort the waveform and modify its harmonic content to create a richer signal for instance [Tolonen et al. 1998; Miranda 2002]. In particular, it can be very simple to get a signal containing only odd or even harmonics with this technique by passing unit-amplitude sinusoid through respectively an odd ($f(-x) = f(x)$) or an even ($f(-x) = -f(x)$) transfer function [Puckette 2007]. Commonly used functions are polynomials (e.g. Chebyshev polynomials) or piecewise functions (e.g. hard clipping).

Most of the abstract techniques for sound synthesis have been developed in the 1970s and 1980s and according to [Serra 2007] were considered obsolete at some point, as more expressive and efficient methods emerged. However they remain famous techniques that take an important place in the sound synthesis history.

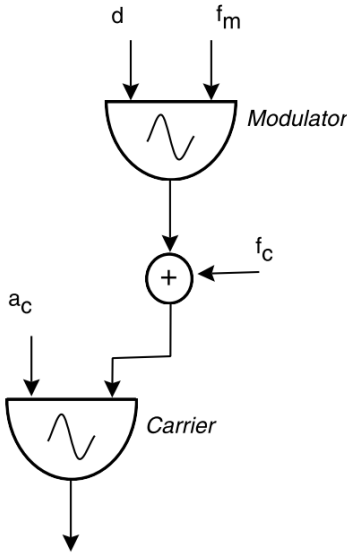


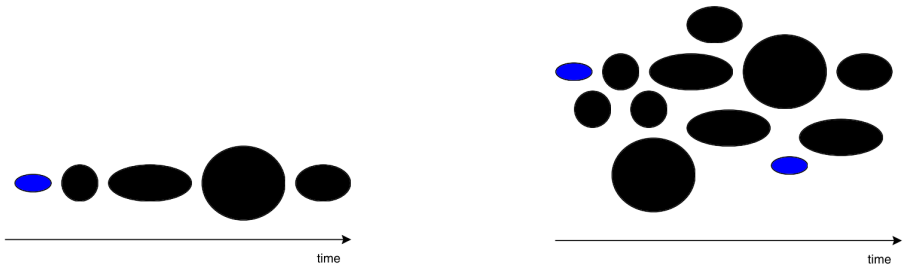
Figure 1.4: Diagram of the most basic FM synthesis instrument with 2 sinusoidal oscillators: one for the modulator and one for the carrier. Picture taken from [Miranda 2002].

1.2.2 Processed recordings

Processed recordings techniques are time-based approaches that use stored recordings of existing sounds to reproduce or create new timbres. At the beginning, in the 1950s, recordings were stored on tapes that could be played with various speed either forward or backward, and that also could be cut, pasted or looped [Miranda 2002]. Then, with the development of computers, samples started to be stored digitally and diverse time-modeling methods arose.

Granular synthesis is one of the first examples of processed recordings synthesis. It consists in putting end to end very short sounds called *grains* lasting from one to a few hundreds milliseconds in general [Tolonen et al. 1998]. These grains can either be slices of a longer recorded sound or artificially generated. To aggregate grains, three different approaches exist [Miranda 2002]: the sequential approach where the grains are generated one by one without overlap (see Figure 1.5a); the scattering approach where several grains can be generated at the same time synchronously or not, as if they were forming a *sound cloud* (see Figure 1.5b); and the granular sampling approach where a grain is created by first selecting a small portion of a sound sample and then applying an envelope to it. For the latter, the resulting sound is created either by replicating the grain many times, or by aggregating several grains extracted from different portions of the signal. For this method, the choice of the envelope is very important in order to avoid discontinuities between the consecutive grains when aggregated.

Concatenative synthesis is another example of sample-based method that is close to granular synthesis but with an additional notion of analysis of the units (equivalents of the grains of granular technique) by extracting audio descriptors from them. This method relies on a huge database containing sound samples from real recordings segmented into small units and their corresponding audio descriptors. Then, in order to generate the target sound, the best matching units (the ones which have the closest descriptors to the target sound) will be found using a search algorithm and concatenated using short cross-fades, possibly after some



(a) Sequential approach. (b) Scattering approach.

Figure 1.5: Representations of different granular synthesis approaches. Figures taken from [Miranda 2002].

modifications such as pitch shifting for example [Schwarz 2004]. The representative diagram of the concatenative synthesis can be found in Figure 1.6.

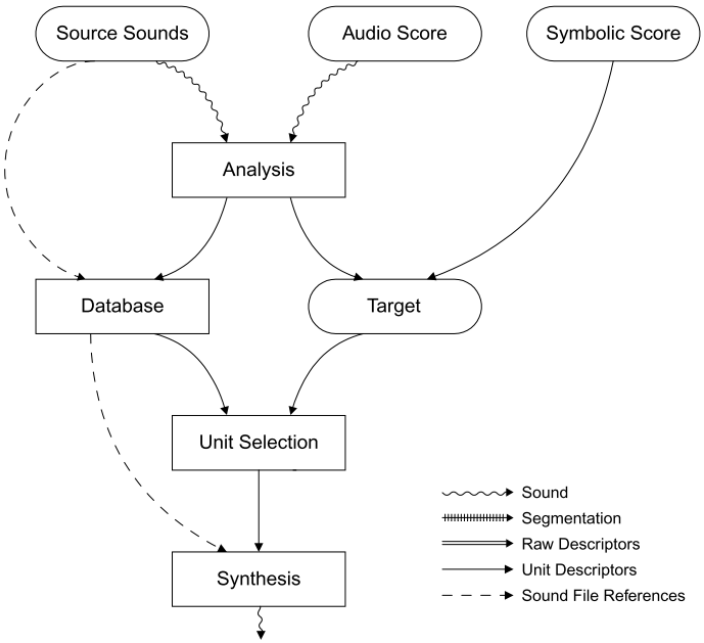


Figure 1.6: Data flow model of a concatenative synthesis system. Diagram taken from [Schwarz 2006].

A last example of temporal-based synthesis that can be given is wavetable synthesis. Actually this term refers to different synthesis methods, but all involve signal waveforms that are stored in a computer memory called wavetable or lookup table. The signals can either be purely synthetic (single-cycle waves for example, i.e. precisely one cycle/period of signal) or recorded excerpts. Several generation techniques exist such as repeatedly reading a waveform to create a periodic sound, cross-fading between two wavetables to create new timbres or combining several wavetables together (by applying an envelope to each one and then summing them) for instance [Miranda 2002].

One of the main drawbacks of these techniques is that they require an important amount of memory for storing the database of samples that could happen to be rather large. But as most of the information is already stored, they are usually efficient in terms of computations.

1.2.3 Spectral modeling

Spectral modeling approaches are based on characterizing the properties of the sound by focusing on its spectrum content and thus return models that are closer to human sound perception [Smith 2004] as the inner ear operates in the frequency domain. The parameters given by these techniques are therefore far from the acoustic mechanisms that allow to produce them but are more representative of the psychoacoustics [Miranda 2002; Tolonen et al. 1998].

The most famous and oldest method in the spectral modeling synthesis family is the additive synthesis (synthesis technique used in the first synthesizer: the Telharmonium). The method assumes, greatly inspired by the Fourier analysis theory, that any periodic waveform can be thought of as a sum of sinusoids with different amplitude envelopes and different frequencies [Miranda 2002], see example in Figure 1.7. The generation consists then in summing the output of several sinusoidal oscillators with different frequencies $f_k(n)$ and different amplitude envelopes $a_k(n)$. The main drawback of this very modular and powerful method is the number of parameters involved. Indeed, for synthesizing signals with a very rich harmonic content or noisy signals, the number of needed oscillators can be really high, inducing a difficult manual and computationally demanding control [Tolonen et al. 1998].

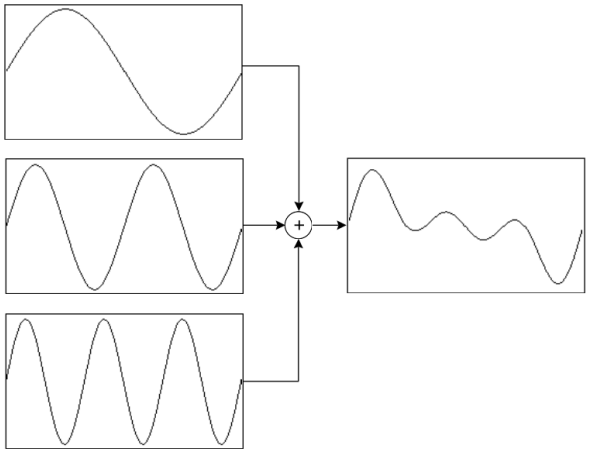


Figure 1.7: Representation of additive synthesis with 3 sinusoidal oscillators ($M = 3$). Figure extracted from [Miranda 2002].

Another famous example of spectral model technique is the subtractive synthesis used since the 1960s in many hardware and software commercialized synthesizers. This technique, also called source-filter synthesis, is at the opposite of additive synthesis. Instead of adding simple and harmonically poor waveforms to get a richer signal, in subtractive synthesis the idea is to start with a very rich excitation signal, usually pulses or noise, and pass it through a filter in order to remove the unwanted harmonics [Tolonen et al. 1998]. The controls are thus much less numerous than for additive synthesis, but are specific to the chosen architecture of the filter(s) and the produced sounds have almost always an "artificial" nature.

1.2.4 Physical modeling

The general principle of this method is to mathematically model the laws of physics responsible of the sound production by means of a set of equations and algorithms. In the case of musical instruments for example, sounds are synthesized by characterizing the behavior of all the elements of the instruments that are responsible for the sound such as the body, the reed or the strings [Serra 2007]. Although, contrary to spectral modeling techniques, physical modeling methods try to model the physical mechanisms of synthesis that are not necessarily directly linked to the perception but only to physics [Smith 2004], they give the user a sense of a real instrument [Tolonen et al. 1998].

According to [Tolonen et al. 1998] different categories of physical modeling exist. The first and oldest one is the numerical solving of differential waves equation introduced in [Hiller and Ruiz 1971]. This method is applicable to any vibrating object but has been used principally to model string instruments. Modal synthesis is another approach [Adrien 1991] where an instrument is represented by a combination of substructures called modal acoustic structures and their interactions [Tolonen et al. 1998]. These components can be membranes, air columns, metal plates, strings or bridges for instance. By assembling elements and using interactions that are physically impossible to realize, it allows to create new interesting timbres. A last example of physical modeling technique that can be given is the waveguide synthesis. It is one of the most used methods for physics-based commercialized instruments, probably because it is the most computationally efficient for real-time sound generation. For this method, the synthesis is also made by solving the wave propagation equation but in this particular case, the wave is simulated in the waveguide model which consists in a bidirectional delay line and filters [Smith 1992], see Figure 1.8.

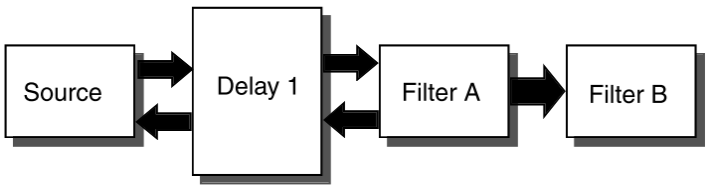


Figure 1.8: Generic waveguide filter instrument. Figure taken from [Miranda 2002].

Although physical modeling synthesis provide musicians with physically meaningful controls and the possibility to synthesize sounds with high quality, the resulting models are very specific and greatly depend on the type of sounds they have been designed to produce, such as plucked strings or reed instruments sounds for examples.

1.3 Synthesis using deep machine learning

The classic methods that were introduced previously (Section 1.2) all present advantages but also drawbacks, the most significant one being the fact that these methods provide musicians with synthesis parameters that are either too numerous (additive synthesis), too memory consuming (processed-recordings methods), extremely complicated to control (abstract algorithms) or very specific (physical modeling, subtractive synthesis). Mastering these techniques require expert knowledge about generative processes and often involves years of

training. Moreover, these methods are really different and allow to generate quite distinct types of sounds. It is thus very difficult to benefit from the whole variety of timbres using these synthesis techniques.

More recently, researchers have started to investigate new synthesis methods using data-driven machine learning techniques, most of them involving the use of artificial neural networks (ANNs). These methods are mostly designed to alleviate the previously mentioned issues (i.e. barriers between the different synthesis methods and complexity of control parameters) and allow the models to generate all the types of sounds it has been fed with during training and even more thanks to their extrapolation properties while extracting a unique control space.

1.3.1 Machine learning and deep neural models

In this section we introduce the fundamental concepts of machine learning and in particular deep learning which is the favored current framework for synthesizing sounds. This section does not aim at giving an exhaustive understanding of the models and how they work, but only at presenting the basics and illustrating some concepts that will be useful for the rest of the thesis. For more details about machine learning and deep neural models, the reader is referred to [Bishop 2006] or [Goodfellow et al. 2016].

1.3.1.1 Machine learning

Machine learning is a scientific field of study aiming at designing algorithms that learn how to perform a particular task from examples [Bishop 2006]. The main challenge for these algorithms is to learn how to solve tasks that are intuitive for people to realize but difficult to describe formally [Goodfellow et al. 2016], such as recognizing digits from handwritten samples for instance.

Main tasks The main tasks that can be realized by machine learning models can be grouped into two different categories: classification/clustering and regression. The first category consists in associating an input example to one (or several in the case of multi-class) discrete value corresponding to the class the example belongs to, or the cluster if the learning is unsupervised. Getting back to the digits recognition example, this would be the corresponding digit value for instance. For the regression task, the input is not mapped to discrete values but to a set of one or more continuous variables. These variables could correspond for example to features or to another continuous representation space of the input and the model would then perform a mapping from the data to this new space, which is of main interest to us.

Learning methods In order to learn, the machine learning algorithm tries to optimize a well-chosen objective function by modifying the parameters of the model as it is fed with examples. Carrying on with the digits example, the objective function to optimize could be the digit classification accuracy. Once the training has been done, the model is supposed to be able to effectively perform the task given a new never-seen example, this is called *generalization*.

There exist three main paradigms to train the models [Bishop 2006]. The first one is supervised learning. In this case, during training, the model is fed with both the input example and its target label (e.g. the handwritten sample and its associated digit, say "4"). The second technique is unsupervised learning. Contrary to supervised learning, there is no target label associated to the input data and the model has to learn how to perform the task from the examples only. The last main learning technique is the reinforcement learning. This method is slightly different from the others as it involves finding the best action to take given a specific situation in order to maximize a reward. There are also some other training techniques, such as semi-supervised learning (that will be discussed in Chapter 4) where the example dataset is not fully labeled, but they are out of the scope of this short machine learning introduction.

1.3.1.2 Artificial neural networks

Artificial neural networks (ANNs) constitute a specific type of machine learning models that is composed of a set of interconnected computational units called *neurons* (as originally inspired by the behavior of biological neurons [Rosenblatt 1961]), see Figure 1.9. Each unit is connected to several inputs (x_i) and returns a single output y that consists in the results of the (most often) non-linear activation function φ applied to a weighted sum of all the inputs plus a bias b. The weights w_i together with the bias b constitute the parameters of the unit.

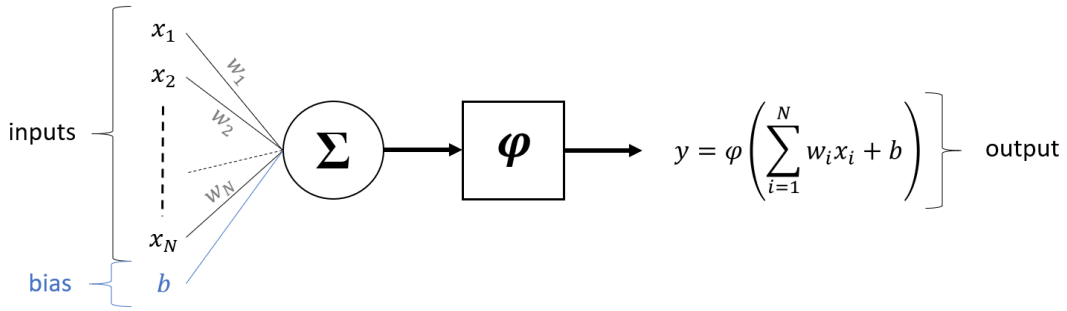


Figure 1.9: An illustration of the structure of an artificial neuron. w_i is the weight associated to the i^{th} input, b is the bias and φ is called activation function, it is usually non-linear.

In ANNs, the units are organized into layers where all the units in a same layer share the same inputs but are not connected together, forming the so-called feed-forward neural network. In the case of deep learning, the models present several layers of neurons whose outputs serve as inputs for the neurons of the next layer, forming a deep neural network (DNN), see Figure 1.10. We can distinguish 3 different types of layers: the input layer, that is the first layer of the model; the hidden layers whose number varies depending on the structure of the model; and the final output layer. For regression tasks, the activation function of the output layer is usually chosen to be linear.

The training of the ANN is done by finding the optimal parameters of each neuron of the model (weights and bias) in order to optimize the overall objective function. This optimization problem can be addressed using several variants of the gradient descent algorithm (such as the well-known stochastic gradient descent – SGD) combined with back-propagation algorithms

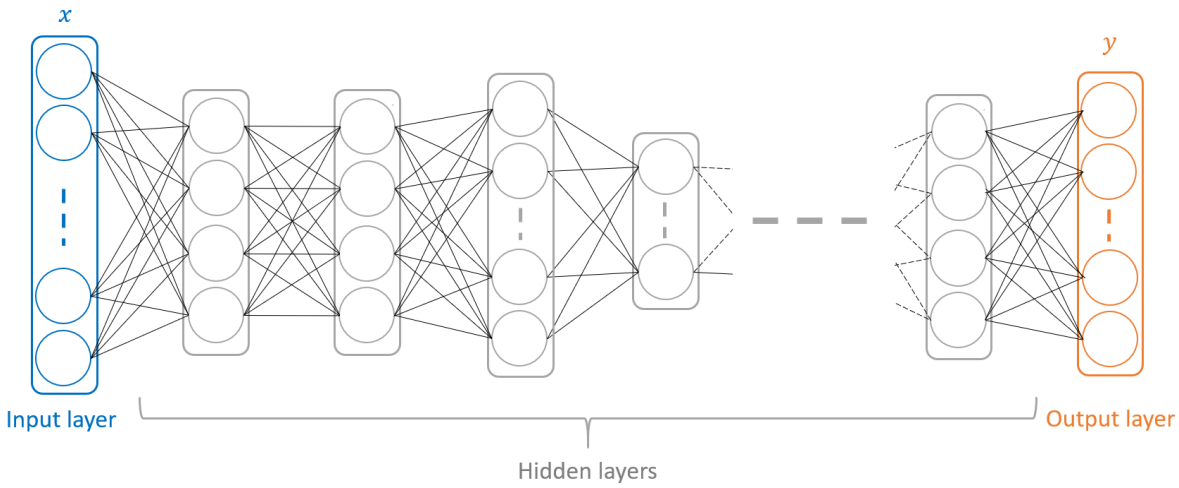


Figure 1.10: Structure of a deep neural network (DNN).

[Rumelhart et al. 1986].

In the next sections we will present in more detail specific deep neural models that are commonly used for data generation. Most of the time, these neural models have first been investigated focusing on high-quality image synthesis before being applied to audio.

1.3.2 Deep neural models for image synthesis

Deep neural networks (DNNs) and in particular those trained in an unsupervised (or self-supervised) way such as autoencoders (AEs) [Hinton and Salakhutdinov 2006] or generative adversarial networks (GANs) [Goodfellow et al. 2014], have shown interesting properties to extract latent dimensions from large and complex datasets.

AEs are a specific type of DNNs that can learn from data a non-linear projection of the signal space into a low-dimensional latent space (*encoding* step), followed by an inverse non-linear transformation of the latent coefficients into the original space (*decoding* step) [Vincent et al. 2010], see Figure 1.11. They have essentially been used as an unsupervised technique for data dimension reduction. For example, in [Hinton and Salakhutdinov 2006], AEs have been applied to three different image datasets on an analysis-synthesis task: a dataset of random samples of curves that they artificially generated, images from the MNIST dataset [LeCun et al. 1998], and grayscale images derived from the Olivetti faces dataset[1]. In their paper they compared the reconstructed images obtained with AE to those obtained with another dimension reduction technique, the principal component analysis (PCA) using the same compression factor (the number of principal components used to reconstruct the data for the PCA, and the number of neurons in the bottleneck layer for AEs, see Section 3.2.2.2 for more details). They showed that AE greatly outperforms PCA on these three datasets, synthesizing images from their latent representations that were much closer to the original data both qualitatively and in terms of mean squared error (MSE). In [Vincent et al. 2010] they also used AEs to synthesize images from the MNIST dataset but in addition, they introduced a variation of AEs that is the denoising AE, a model trained to reconstruct data

[1]The Olivetti dataset is publicly available from https://cs.nyu.edu/~roweis/data.html.

from a corrupted version (by noise principally). In their paper they compared denoising AE with standard deep AE and deep belief networks (DBN) by analyzing the reconstructed images obtained after decoding using a non-parametric sampling procedure (see details in [Vincent et al. 2010]), and showed the power of this variation of an AE for corrupted data.

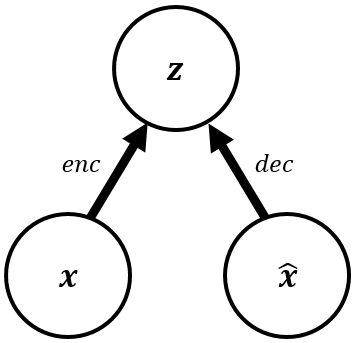


Figure 1.11: General structure of an autoencoder where *enc* represents the *encoder* network, *dec* the *decoder* network and $\mathbf{z}$ is the latent representation of $\mathbf{x}$. The model is trained to minimize the reconstruction error between $\mathbf{x}$ and $\hat{\mathbf{x}}$. Figure inspired from [Goodfellow et al. 2016].

More recently, another variation of AEs that has been used for data generation is the variational autoencoder (VAE) [Kingma and Welling 2014]. It can be seen as a probabilistic/generative extension of standard AEs as, instead of deterministically mapping the input vector $\mathbf{x}$ to a unique latent vector $\mathbf{z}$ as done in AEs, the VAE encoder network maps $\mathbf{x}$ into the parameters of a conditional distribution $q_\phi(\mathbf{z}|\mathbf{x})$ of $\mathbf{z}$. Similarly, the decoder network maps a vector of latent coefficients $\mathbf{z}$ into the parameters of a conditional distribution $p_\theta(\mathbf{x}|\mathbf{z})$ of $\mathbf{x}$. VAEs are thus considered as generative models as they try to capture the probability distribution of the data. Importantly, in a VAE, a prior can be placed on the distribution of the latent variables $\mathbf{z}$ so that they are well-suited for the control of the generation of new data (see Section 3.2.2.4 for more details). These models have been extensively used for image synthesis of many types: digits from the already introduced MNIST dataset [Kingma and Welling 2014; Salimans et al. 2015], faces images [Kingma and Welling 2014; Rezende et al. 2014; Kulkarni et al. 2015; Higgins et al. 2017], tiny images of real life objects (CIFAR dataset [Krizhevsky 2009]) [Gregor et al. 2015] or even 3D models of chairs [Kulkarni et al. 2015; Higgins et al. 2017], or prediction of the future of static images [Walker et al. 2016]. VAEs appear to be well-adapted for generating high-quality images, although slightly blurry, and seem to extract a representation space with very interesting properties by constraining the statistical properties of the latent dimensions, insuring some amount of decorrelation between them. These latter properties make them good candidates for extracting good control parameters for synthesis.

GANs are another state-of-the-art type of deep generative models that have been widely used for image synthesis [Goodfellow et al. 2014]. They are composed of a pair of competing neural networks: a *generator* and a *discriminator*, see Figure 1.12. The purpose of the discriminator is to distinguish the fake data generated by the generator from the real data. The generator on the other hand, is trained to fool the discriminator by synthesizing images

that are indistinguishable from the real images of the dataset. These two networks are trained alternately, either optimizing the objective function of the discriminator or the one of the generator. GANs have been successfully employed for synthesizing images from very diverse nature: digits from MNIST dataset [Goodfellow et al. 2014; Shmelkov et al. 2018], faces images [Goodfellow et al. 2014; Karras et al. 2018], everyday images from the CIFAR dataset or the LSUN dataset [Yu et al. 2015]; [Goodfellow et al. 2014; Karras et al. 2018; Shmelkov et al. 2018] or other more complex datasets [Reed et al. 2016; Ledig et al. 2017; Isola et al. 2017; Zhang et al. 2017; Zhu et al. 2017]. GANs appear to be rather difficult to train correctly, but seem to be the state-of-the-art for high-quality image generation.

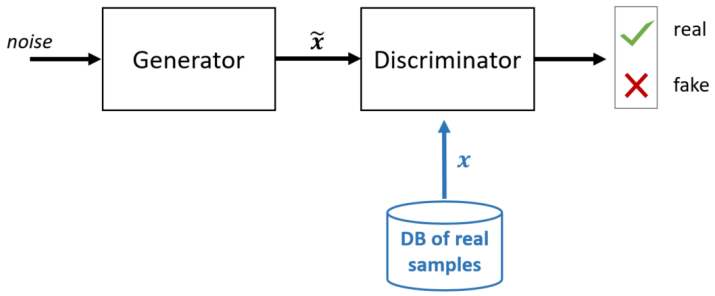


Figure 1.12: Diagram of the structure of a GAN.

Finally, some researchers have also investigated some deep models combining VAEs and GANs for image synthesis purposes [Makhzani et al. 2015; Larsen et al. 2016; Mescheder et al. 2017], see Figure 1.13. The idea behind this combination is to generate images with a very high quality while extracting a high-level representation space, giving thus some extension of the VAE model. These new types of VAEs have been applied for the generation of images from MNIST, faces or street numbers and gave very good results, the synthesized images being generally much sharper than with classic VAE models.

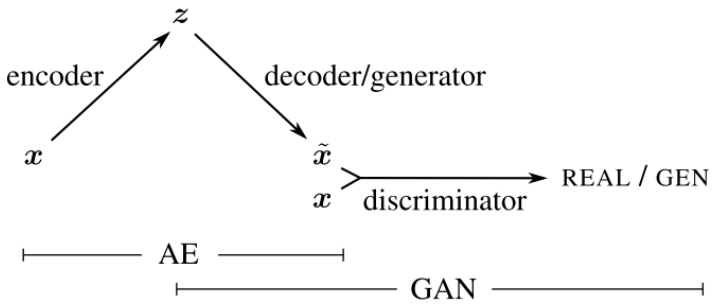


Figure 1.13: Example of structure of a VAE/GAN model. Figure extracted from [Larsen et al. 2016].

1.3.3 Deep neural models for sound synthesis

Motivated by the results obtained for image synthesis, researchers started to apply these deep neural models to audio. These methods have already been extensively applied for music information retrieval (MIR) applications such as melody extraction [Park and Yoo 2017;

Basaran et al. 2018], musical track/source separation [Leglaive et al. 2015; Miron et al. 2017; Chandna et al. 2019], instrument recognition [Han et al. 2017; Pons et al. 2017] or tempo estimation [Schreiber and Müller 2018; Foroughmand and Peeters 2019]. However, so far, the use of such models for music sound synthesis applications is still a very open subject and only a few studies have dealt with it. We will present them in the following sections. For more clarity we differentiated the studies applying autoencoder-based models that inspired our work, and the very recent papers published on GAN-based models for audio synthesis.

1.3.3.1 Autoencoder-based models

Most of the studies dealing with audio processing using deep neural models involved autoencoder-based models with a same general principle that is similar to image synthesis/transformation: first the original signal is projected into a low-dimensional latent space (encoding), then possible modifications are applied to the latent coefficients (also called embeddings), and finally an inverse transformation of the (possibly modified) latent coefficients is performed to get back to the original space (decoding).

To our knowledge, the first study applying classic AE models for processing music sound is [Sarroff and Casey 2014]. In this study, the authors extracted normalized magnitude spectra from a dataset of 8,000 songs from 10 different musical genres and used them as input of the AE. The audio signal is then reconstructed using the decoded magnitude spectra and the phase of the original signal. They evaluated the model by computing the mean squared error (MSE) between the original and reconstructed magnitude spectra and a limited reconstruction accuracy was observed.

In [Colonel et al. 2017], the authors tried to improve the results obtained in [Sarroff and Casey 2014] by applying AE on a more controlled database of sounds (all generated with a MicroKORG synthesizer) and by using a different optimizer. In both [Sarroff and Casey 2014] and [Colonel et al. 2017], several topologies of the network were evaluated going from shallow AEs to 9-layer deep models varying the number of neurons per layer and the activation functions.

An alternative to using classic feed-forward layers for AE applied on magnitude spectra for music sound synthesis has been used by Google's Magenta team in [Engel et al. 2017]. The authors took inspiration from the WaveNet speech synthesizer [van den Oord et al. 2016b] (itself inspired from the image processing pixelRNN [van den Oord et al. 2016a]) to implement a time-domain autoencoder, thus avoiding to reconstruct or generate the phase spectrum in addition to the amplitude spectrum to synthesize the output signal (or to store the original phase spectrum as in the two previous studies). The model, conditioned on pitch, is composed of a temporal encoder made of a 30-layer residual network of dilated convolutions followed by 1x1 convolutions, and a WaveNet decoder, see Figure 1.14. The authors also built a large-scale multi-instrument and multi-pitch database (the NSynth dataset) that they used to feed the model (as raw audio). This model leads to a latent space of size 125x16 which is then upsampled before being sent to the decoder. Qualitative evaluation showed that this high-level latent embedding was well suited for "morphing" between instruments of the dataset. In this study, the authors compared this NSynth model to a convolutional autoencoder applied on normalized log-magnitude power spectra (called baseline) and reported that their model was able to better reconstruct the signal. One of the main drawbacks of this model is that it

requires a lot of computational power[2] which makes it difficult to train.

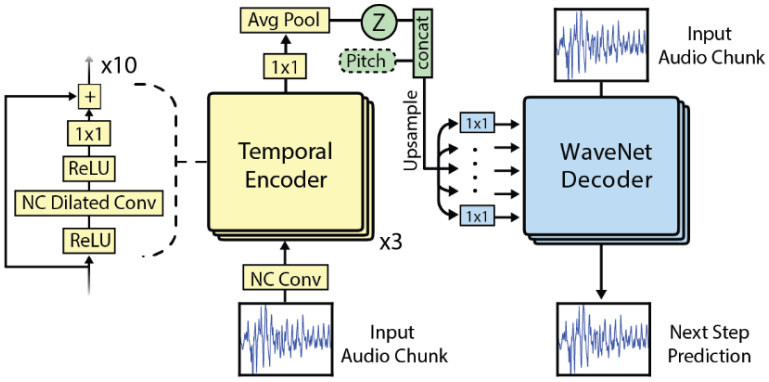


Figure 1.14: WaveNet autoencoder model. Figure extracted from [Engel et al. 2017].

As this was the case for image synthesis, VAEs have also recently been exploited for audio. They have first been applied for modeling, transformation and synthesis of speech signals [Blaauw and Bonada 2016; Hsu et al. 2017; Akuzawa et al. 2018]. In a slightly different line, VAEs have recently been used to model (clean) speech signal for speech enhancement in noise [Bando et al. 2018; Leglaive et al. 2018; Leglaive et al. 2019a; Leglaive et al. 2019b]. These models have also been applied for music sound synthesis [Esling et al. 2018b]. In this last study, the authors tried several spectral representations of signal as input (using only one frame randomly extracted from the sustained part of sounds from an acoustic instrument dataset) and added a perceptual regularization to the model in addition to the classic latent space regularization loss of VAEs. As perceptual "labels", they used perceptual dissimilarity ratings obtained across five different and independent timbre studies based on MDS methods [Grey 1977; Krumhansl 1989; Iverson and Krumhansl 1993; McAdams et al. 1995; Lakatos 2000], see Section 1.1.2.1. The additional perceptual regularization seems to organize much better the latent space and not to impact significantly the quality of the reconstruction which is a very promising and encouraging result towards a perceptually meaningful controlled audio synthesis. This study which is very related to the present thesis work will be extensively presented and discussed in Section 4.2.1.

1.3.3.2 GAN-based models

Very recently, some researchers started to experiment the use of GANs for music sound synthesis [Donahue et al. 2019; Engel et al. 2019] and these are, to our knowledge, the first applications of GANs to unsupervised audio generation.

In [Donahue et al. 2019], the authors introduced two GAN-based models inspired from the DCGAN [Radford et al. 2016]: one called WaveGAN for generating raw audio and the other called SpecGAN for generating magnitude spectrograms (and then applying the Griffin & Lim algorithm to reconstruct the phase spectrogram [Griffin and Lim 1984]). They applied this model to different sound datasets: uttered digits, drum sound effects, bird vocalizations, piano and spoken sentences from TIMIT [Garofolo et al. 1993]. They evaluated them using

[2] According to the authors, the WaveNet model takes around 10 days to converge at 200,000 iterations by training on 32 NVIDIA Tesla K40 synchronous GPUs (graphics processing units).

inception score [Salimans et al. 2016], nearest neighbor comparisons and realized a perceptual study to gather qualitative human judgments, leading to very encouraging results.

In the other study investigating GAN for audio generation [Engel et al. 2019], they applied a GAN on log-magnitude spectrogram of signals from the NSynth dataset [Engel et al. 2017] conditioned on pitch. They compared this model to the WaveNet model and the WaveGAN introduced just above and reported very competitive results.

The use of GAN-based models for synthesizing audio is just at its early stages but it has shown rather promising results and its advancements will thus have to be followed closely in the future.

1.4 Conclusion

In this chapter, we first introduced the different methods that have been applied through history to attempt to better understand and define the musical timbre and its perception through both its macroscopic (causal) and microscopic (qualitative) aspects. In particular, we presented various methods that have been used to extract the most relevant perceptual dimensions of musical timbre and their acoustic correlates.

The commonly used (and commercialized) audio synthesis methods were then reported as music sound synthesis is at the heart of our thesis project and has highly interested the research for the past 150 years. We described and illustrated the main synthesis categories and presented their advantages and drawbacks. We were then able to point out the necessity for a new method that is more robust and adaptable, allowing for possible direct and easy interaction with our perception of timbre for controlling sound generation. To this purpose, we introduced NNs techniques that are commonly used for data generation and focused on studies applying these models for music sound synthesis.

Before describing the deep neural models we explored for synthesizing audio with good quality in Chapter 3 and Chapter 4, in the next chapter we will present and explain the methods we chose in order to extract perceptually relevant verbal descriptors for characterizing synthetic timbres.

Chapter 2

Perceptual characterization of timbre

Contents

2.1 Chosen method **28**
2.2 First perceptual test: Free verbalization **28**
2.2.1 Participants 29
2.2.2 Stimuli 29
2.2.2.1 Arturia dataset generation 29
2.2.2.2 Samples selection 30
2.2.2.3 Participants stimuli assignment 32
2.2.3 Protocol of the study 34
2.2.4 Results analysis 34
2.2.4.1 Objectives of the analysis 34
2.2.4.2 Results pre-processing 35
2.2.4.3 Semantic proximity analysis 36
2.2.4.4 Obtained semantic categories 39
2.2.4.5 Verbal descriptors selection 40
2.2.5 Conclusion 42
2.3 Second perceptual test: Semantic scales analysis **43**
2.3.1 Participants 44
2.3.2 Stimuli 44
2.3.2.1 Training stimuli versus main phase stimuli 44
2.3.2.2 Samples selection 45
2.3.3 Protocol of the study 46
2.3.4 Results analysis 47
2.3.4.1 Objectives of the analysis 47
2.3.4.2 Results pre-processing 47
2.3.4.3 Intra-subject consensus 49
2.3.4.4 Inter-subject consensus 50
2.3.4.5 Final label vectors computation 56
2.3.5 Conclusion 56
2.4 Conclusions and perspectives **57**

Previous studies on timbre perception showed how timbre verbal description depends on the native language and the category of sounds [Castellengo and Dubois 2005; Zacharakis et al. 2014], and very few studies specifically focused on the perception of synthetic sound that do not imitate orchestral musical instruments. The first main challenge of this thesis was thus to evidence perceptually meaningful and consensual French verbal descriptors adapted to describe synthesizer sounds and to evaluate how they could be used in order to serve as control parameters for a new intuitive synthesis method.

The first part of Chapter 1 introduced the main approaches that have been developed by researchers to study timbre perception and its acoustic correlates. This chapter will detail the global methodology, inspired by these studies, that we decided to apply in order to achieve our objectives. We will describe the different perceptual tests we conducted, our analysis of the collected answers and finally the obtained results towards the identification of a usable perceptual space.

2.1 Chosen method

In the context of our project, the perceptual characterization of synthetic timbre can be divided into two phases. The first step is to identify a set of perceptually meaningful and consensual terms that are used to describe synthesizer sounds and that will constitute the perceptual space. The second stage is then to select the most relevant terms, i.e. the main perceptual dimensions of our sound space, and to project different sound samples into this reduced space (as for MDS studies) in order to understand how they could possibly be used for control. This last step can also be seen as labeling a subset of samples from a larger dataset in anticipation of a potential (weak) supervision of neural models training (see Chapter 4).

Considering the nature of our data, i.e. synthesizer sounds, it seemed more relevant to us to focus on the microscopic aspects of timbre in order to truly describe the precise characteristics of the sound rather than dealing with source identification (which could be a very complicated task for purely synthetic sounds). Hence, we concentrated on qualitative timbre approaches introduced in Section 1.1.2.2. Taking inspiration from [Faure 2000] and [Ehrette 2004], we decided to use a combination of two timbre perception approaches in order to define our perceptual space: first conducting a free verbalization perceptual study to gather consensual and frequently used terms, and then using a semantic differential (SD) approach involving scales labeled with the terms that emerged from the first test.

2.2 First perceptual test: Free verbalization

Free verbalization perceptual studies aim at collecting descriptions of a set of stimuli by asking participants to freely characterize them. In our case, the main objective of this perceptual test was to gather verbal descriptors that are frequently and consensually used to describe synthesizer sounds. For this study, we asked participants to describe the sounds

using the French language[1].

In this section we will present the participants, the stimuli and detail the protocol of our free verbalization study. We will then explain how we analyzed the collected answers and finally present and discuss the obtained results.

2.2.1 Participants

Before taking the test, the participants were asked to answer a quick questionnaire in order to collect personal information concerning:

- their mother tongue,
- their experience with audio (e.g. musician or audio researcher),
- their age category (e.g. 15-24 or 25-34),
- their listening conditions (e.g. quality headphone or studio monitors) while strongly advising them not to use laptop speakers.

The purpose of this stage was to get information about the participants that could be helpful for the analysis of their answers.

The perceptual study was distributed online using specialized diffusion lists and our personal networks. In total we collected 101 responses. Most of the participants were French native speakers and about half of them were musicians. The vast majority of the participants used headphones to take the test and almost 65% of them were under 35 years-old.

For more details about the questionnaire given to the participants or the collected information, please refer to Appendix A.1.

2.2.2 Stimuli

When designing a perceptual test, the choice of the stimuli is crucial and the samples have to be wisely chosen depending on the goals to achieve. In our case, we wanted the participants to focus on qualitative aspects of timbre (and not on instrument recognition/categorization for example). Plus, the most basic and consensual definition of timbre being "what allows to distinguish two sounds presenting the same loudness and same pitch" [von Bismarck 1974], it was thus mandatory to try to limit as much as possible the variations of the sounds in terms of pitch, loudness and duration in order to help listeners focus on the target characteristics and not to be distracted by other aspects of the sounds.

The first step towards the selection of the stimuli was thus to generate a sample dataset that fulfill these criteria.

2.2.2.1 Arturia dataset generation

To obtain appropriate samples for the perceptual test, the first requirement was thus to generate stimuli all set to one and same pitch. Indeed, a change of pitch can greatly modify the timbre perception of the sound [Castellengo 2015; Marozeau et al. 2003], it is thus very important to set it when comparing timbres of different instruments/sounds. We arbitrarily

[1] Actually two tests were created and distributed, one in French and one in English, but the number of collected answers were much higher for the French test than for the English one. We thus decided for this manuscript to present only the results obtained for the French test.

chose to generate sounds with a fundamental frequency of 164.81 Hz (corresponding to an E) as it is contained in the frequency band going from 150 to 200 Hz, this band being the one with the highest density of orchestral instruments (thus comparable to other studies) and at the intersection of male and female voice ranges. Plus, this pitch is low-frequency enough for a satisfactory sampling of the spectral envelope, which makes it an appropriate value for our study.

A second requirement was to normalize the samples in loudness. Indeed, if we have stimuli with different frequency contents (e.g. one sound presenting a high energy around 3 kHz compared to the others) the perceived intensity of some of them could be much higher than for others, which could have an important impact on the perceived timbres (e.g. making them sound more aggressive). It is thus very important to try to limit the variations in loudness.

Finally, as we wanted the participant to concentrate on the qualitative timbre, as explained in Section 1.1.4, the duration of the samples was a very important parameter. Hence we chose to limit the study to rather short samples with a duration of between 2 and 2.5 seconds to favor focus of the participants on microscopic aspect.

Following these necessary conditions, we generated audio samples from every preset[2] (around 5,000) of the software applications of Arturia[3], constraining the generation to a particular pitch, intensity, and a given duration. After normalization in loudness from informal listening and removal of non-adapted samples (e.g. samples generated from sequence presets[4], polyphonic presets[5], or unpitched presets[6]) we eventually obtained a dataset of 1,233 purely synthetic sound samples. For the rest of this thesis, this dataset will be referred to as "the Arturia dataset".

2.2.2.2 Samples selection

Once the appropriate samples well-defined and generated, the next step towards the implementation of the perceptual test was to select the audio samples that will be presented to the participants.

Although it is a simple way to collect a huge quantity of data, we decided not to go through an Amazon Mechanical Turk process[7] as it involves to be very careful while treating and analyzing the data. Indeed, the seriousness of the participant is not evaluated which implies the need to carefully examine the reliability of the collected data and this was a time consuming process we wanted to avoid. It was thus unrealistic to recruit enough participants to describe more than a thousand stimuli and we had to limit the number of annotated

[2] A preset is a pre-programmed configuration of the synthesizer resulting in one particular sound. In general, digital synthesizers are commercialized with several factory presets created by sound designers.

[3] Partner company of this PhD thesis. A list of the concerned synthesizers is available at https://www.arturia.com/products.

[4] A preset generating a sequence of notes instead of only one note when one key is pressed.

[5] A preset where several notes are played at the same time instead of one when one key is pressed, like a chord for example.

[6] For example a percussive sound without pitch.

[7] Amazon Mechanical Turk (AMT) is a crowdsourcing marketplace enabling users to have people doing more or less complex tasks for them (e.g. passing a perceptual test) in exchange for little wages. It allows researchers to collect a lot of answers to their online tests for a reasonable amount of money.

samples and select them among the whole generated dataset. It was therefore necessary to evaluate the adapted number of samples to be presented to the participants and to choose them wisely to be as representative as possible of the acoustic space described by the whole dataset.

Sizing the test To determine the number of stimuli to select for the test, we started from the acceptable number of samples that we could present to a single participant: in our case we set this number to 20 so that the test has a duration of about 20 minutes. This duration seemed adapted in order for the participants to avoid fatigue, distraction and reluctance to take the test. Then we set the number of participants we thought we were able to reach, in our case 50, and finally the minimum number of descriptions we needed to collect for each sound in order to have sufficient statistical power which is of about 20.

Following these criteria, the resulting number of samples to select for being described through the free verbalization perceptual study was set to 50.

Audio descriptors extraction When selecting the 50 stimuli that will be presented to the participants of the test, we wanted be as representative of the acoustic space defined by the Arturia dataset as possible. To obtain an objective measure/representation of this space, we thus decided to extract a set of audio descriptors that are usually applied for characterizing sounds [Peeters et al. 2011]:

- mel-frequency cepstral coefficients (MFCC),
- spectral centroid,
- spectral bandwidth,
- spectral contrast,
- spectral flatness,
- spectral roll-off,
- zero-crossing rate (ZCR).

These descriptors were extracted using the *librosa* Python package [McFee et al. 2015]. For more details about the chosen descriptors, the reader is referred to the online documentation of the package[8], [Peeters et al. 2011] or [Peeters 2004].

Random selection by clustering Once the audio descriptors extracted and then normalized (by removing the mean and dividing by the standard deviation for each individual descriptor), we used them as inputs of a clustering algorithm in order to divide the acoustic space into different regions where the samples are supposed to share similar acoustic properties. We chose to use the k-means [Arthur and Vassilvitskii 2007] algorithm with 50 clusters to get 50 different regions of the space with particular acoustic attributes from which we could select the stimuli that would be presented in the study. The final sounds were then randomly sampled one by one for each cluster.

The stimuli being non-uniformly distributed among the acoustic space, two different randomly chosen samples from close clusters might be very similar and this is something we wanted to avoid. The samples were thus manually selected to validate the fact that they were

[8] https://librosa.github.io/librosa/

dissimilar enough from one another to represent as much as possible the variety of the whole dataset of sounds. To help us do so, we implemented a tool in Python allowing us to both listen to and visualize every sample of the whole dataset (including the selected ones) in a two-dimensional projection of the acoustic space using t-SNE algorithm [van der Maaten and Hinton 2008] on the audio descriptors, see example in Figure 2.1. From this figure, we can confirm that the samples selected for the test do actually cover well the sonic space created by the entire dataset.

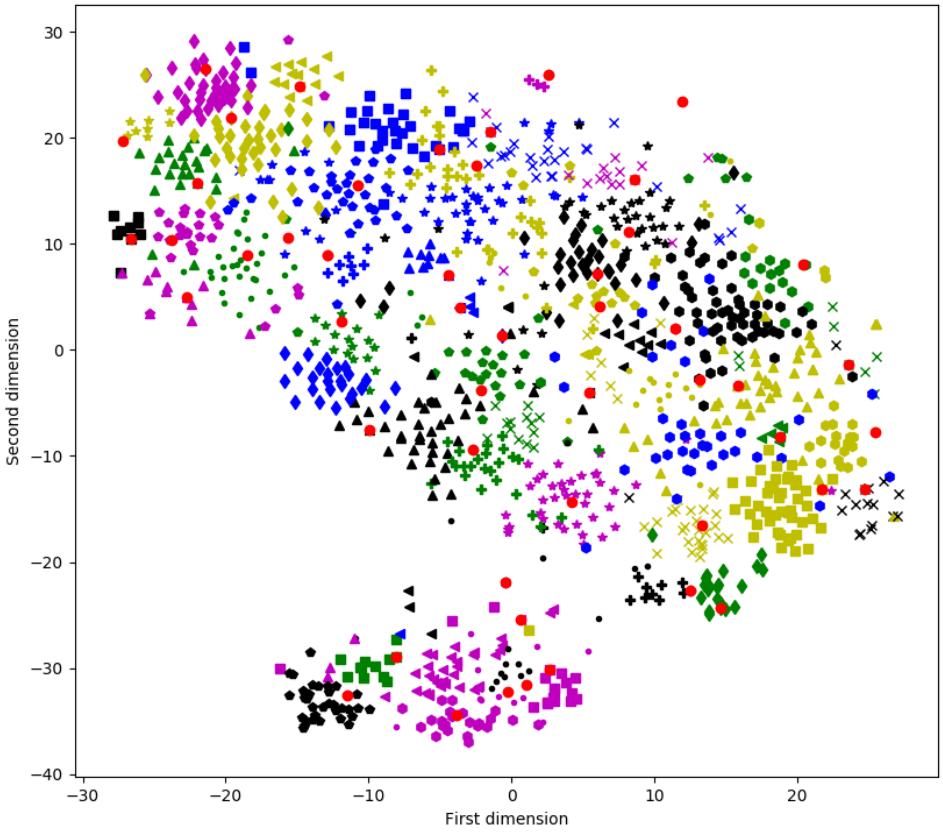


Figure 2.1: Example of t-SNE projection of the samples in a 2-dimensional space obtained using our visualization tool. In red are the samples selected for the free verbalization perceptual study. We can see that, thanks to the k-means clustering, the selected samples cover well the acoustic space (within the limits of the selected synthesizer used to generate the dataset). Different shapes and different colors indicate the different clusters obtained with k-means.

2.2.2.3 Participants stimuli assignment

The last stage in processing the samples for the perceptual study was to decide how they would be organized and assigned to the participants. The main concerns we had were to guarantee a certain amount of overlap between samples evaluated by different participants in order to get sufficient statistical power, and to make sure that the histogram of the answers was as flat as possible (i.e. to balance the number of evaluations between the sounds) for at least the 50 first raters.

To do so a block design was applied. The stimuli were randomly grouped within blocks of 10 samples, thus resulting in 5 different blocks. Then, each time a participant started the test, he or she was assigned two blocks of samples, see schematics in Figure 2.2. Then

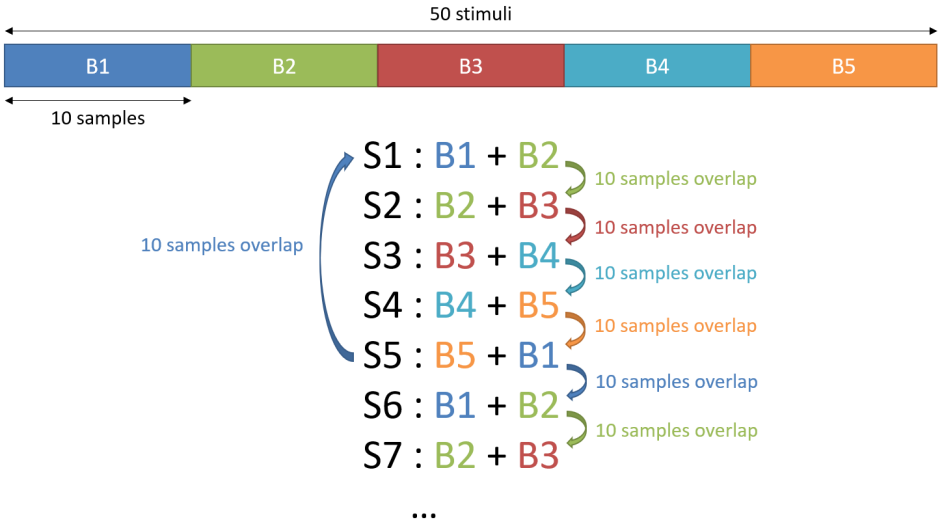


Figure 2.2: Representation of the block design. Sn represents the n^{th} participant passing the test and Bm the m^{th} block of 10 samples.

when another person started the test, he or she received two other blocks by making sure there was an overlap of 1 block with the previous participant. By this mean, an overlap of 10 samples was always guaranteed between consecutive participants. Plus, this method insured the answer histogram to be perfectly flat every 5 listeners, allowing a perfect balance between the described stimuli and maintaining a statistical equity, see Figure 2.3.

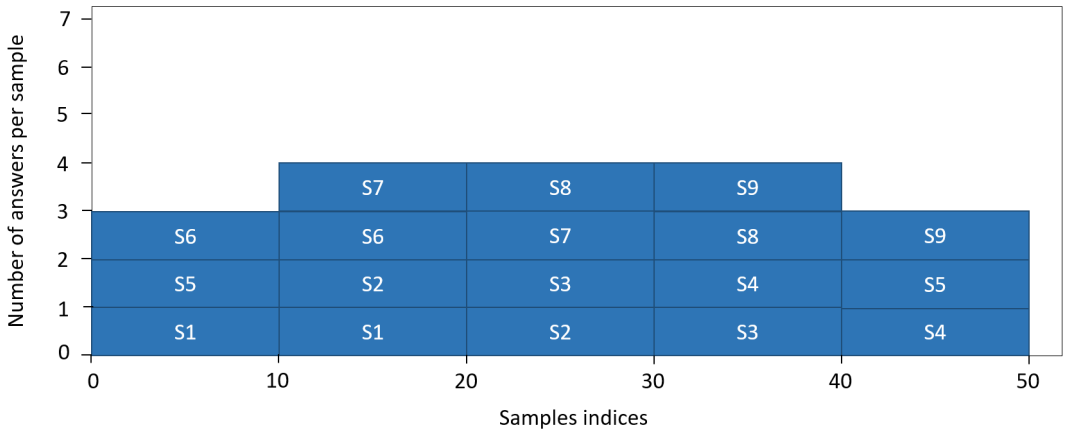


Figure 2.3: Histogram of the answers received per sample depending on the number of participants that passed the test in the context of an organization of the samples in blocks. Sn represents the n^{th} participant that described the samples.

Note that each participant was associated with 2 well-determined blocks, but during the test, the samples were presented randomly.

2.2.3 Protocol of the study

The test was designed to be taken online. After a quick explanation of the protocol, the 20 samples were presented successively to the participants. They could listen to each sample as many times as necessary and were asked to describe verbally the sound sample using isolated words or short sentences. They had 5 empty spaces to write their description (see the screenshot of the interface in Figure 2.4) and had to fill one of them at least before they could move on to the next sample. They were incited to fill as many slots as possible and to use words that are as descriptive as possible and to avoid any value judgment term such as *beautiful* or *ugly*.

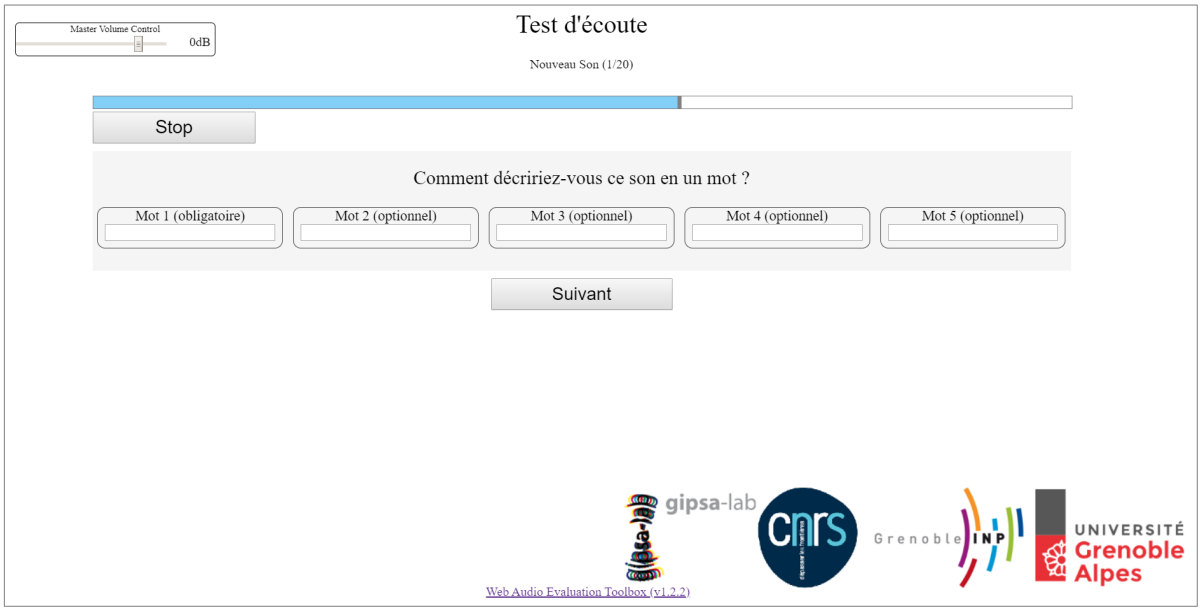


Figure 2.4: Screenshot of the free verbalization test interface. We designed and implemented the test using the Web Audio Evaluation Tool[9][Jillings et al. 2015].

For more details about the protocol explanation given to the participants, the reader is referred to Appendix A.2.

2.2.4 Results analysis

2.2.4.1 Objectives of the analysis

The purpose of this perceptual test was to collect verbal descriptors that are used to describe synthetic sounds in French. As explained in Section 1.1.4, participants may use different terms to qualify the same sound depending on their expertise or the listening context/goal. The main objective of this analysis is then to select, from the collected terms, the ones which are the most frequent and consensually used among the participants and possibly semantically orthogonal to avoid redundancy.

[9]This tool provides javascript (js) and XML programming interfaces (API) that enable creating dynamic perceptual online studies and storing results in XML files very easily. The given API has been enriched by new elements we developed (js) to create our perceptual test.

The first idea we had was to apply some already trained word embeddings models [Mikolov et al. 2013] to create the categories and group the terms as these models have been designed to create projections of the words into a semantically meaningful space. This was unfortunately unsuccessful due to the word embedding model lacking an extensive auditory perception lexicon enriched with contextual information. Indeed, in a purely semantic context the categories made totally sense, but in a timbre perception context, the categories were irrelevant. For example, one of the group was constituted of both *chaleureux/chaud* ("warm") and *froid/glacial* ("cold/freezing"). From a semantic point of view, this makes perfectly sense, but they actually present different characteristics in a perceptual context. Another example that can be given is a category grouping *inquiétant* ("worrying"), *angoissant* ("frightening") with *grave*. Here again, from a purely semantic point of view this is logical as in French *grave* can mean "serious", "grave", but in the context of the sound it rather means "bass", "deep" and so the group does not make much sense anymore. This experiment confirms the fact that in an auditory context the terms can have a different meaning from the "common sense" (a similar effect can be observed in an olfactory context [Dubois 2008]). A few techniques have been proposed to adapt word embedding models (word2vec) to the context of sound verbal description [Lopopolo and Miltenburg 2015; Vijayakumar et al. 2017]. However, since no ready-to-use model was available, in particular in French and for synthetic sounds, we did not use them.

We therefore focused on finding the most frequently and transversally used (i.e. by the greatest number of participants) perceptual categories obtained by grouping the words that were used in the same context by the participants and that could thus be considered as "semantically related" or "semantically close" for our study. To do so, we first had to clean the data in order to remove possible misspellings and other artifacts of the test, then analyze the semantic proximity of the collected terms so that we could group them into categories and finally evaluate their frequency and transversality to select the final perceptual dimensions that will be used for the rest of the thesis. These different steps of the analysis will be detailed in the following sections.

2.2.4.2 Results pre-processing

As explained in the previous section, before performing any analysis of the result, it was necessary to first clean the collected data and try to reduce the number of used terms by grouping them appropriately. In order to do so, some manual processing was applied.

First, the responses were standardized. We removed capital letters, corrected misspellings and also deleted accents.

Then, we made the strong (an somewhat simplistic) choice to reduce sentences[10] to noun groups (e.g. "a sound that vibrates" would be condensed to "vibrating" or "sine wave with a lot of background noise" to "noisy sine wave") although we are very conscious that variations in morphosyntax, and not only lexicon, may express semantic distinctions [Dubois 1997].

Finally, we made the strong choice to group the terms sharing the same lexical root (e.g. "vibrating" would be grouped with "vibrate", "vibration" or "vibrations") although we are conscious, again, that these different forms, in particular different word terminations, may correspond to different meanings (e.g. the French words "nasal" and "nasillard").

[10] Actually, very few participants used sentences to describe the sounds, they mostly use isolated terms such as adjectives.

From this manual pre-processing/cleaning of the data, we managed to reduce the number of different entries from 871 to 784 (see Appendix A.3 for more details about the created groups)[11].

We can note that among the most frequently used terms, more than half of them were classical terms (such as *doux*, *brillant*, *métallique*, *résonnant*, *vibrant*, *aigu*, *continu* or *chaud* for instance) used to describe the timbre of musical instruments or nuances of timbre of a same instrument that have already been listed in other previous studies [Faure 2000; Cheminée et al. 2005; Garnier et al. 2007; Lavoie 2013]. Interestingly, less than half of them were also more novel and unexpected terms that appear to be specific to the description of synthesizer sounds, such as *spatial*, *sirène*, *robotique*, *distordu*, *rebondissant*, *bourdonnement*, *explosif*, *saccadé*, or *rétro-futuriste* for example.

2.2.4.3 Semantic proximity analysis

Once the first groups were defined, we focused on analyzing the semantic proximity between the different terms (or groups of terms) in order to merge them into semantically consistent perceptual categories.

In the context of this perceptual study, two different cases of semantic proximity emerged:

- when two terms are systematically (at least more than twice) used together by an individual to describe different sounds: *intra-subject* case
- when two terms are used by different participants in a same auditory context: *inter-subject* case.

So as to study these two cases of semantic proximity, we started by computing the 3-dimensional matrix of terms occurrences.

3D occurrences matrix The main interest for creating this 3D matrix is to condense the information of the participants through a 3-dimensional binary matrix of shape (number of terms $\times$ number of participants that took the test $\times$ number of sounds), see Figure 2.5. Indeed, by having a 1 when a term (isolated or from a group of terms) has been used by a particular participant for a given sound and 0 otherwise, all the information of the perceptual test has been represented quantitatively under a very convenient form that is then usable for automatic analysis.

The computations finally resulted into an occurrence matrix of size ($784 \times 101 \times 50$).

Then, in order to evaluate semantic proximity cases using this matrix, we got interested in computing the Jaccard distances.

Jaccard distances matrix The Jaccard distance [Jaccard 1912] is a metric that is used in statistics to measure the dissimilarity between two sets of samples A and B by taking the complementary of the ratio of the cardinals of respectively the intersection and the union of the sets:

$$d_J(A,B) = 1 - \frac{|A \cap B|}{|A \cup B|}.$$

[11] The entire list of terms and groups of terms (i.e. the 784 entries) is available in the companion webpage: http://www.gipsa-lab.fr/~fanny.roche/PhD_thesis.html.

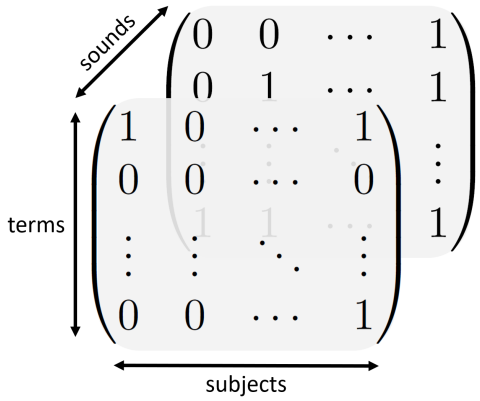


Figure 2.5: 3D occurrences matrix example.

In particular, for binary ensembles the Jaccard distance can be written:

$$d_J(A,B) = 1 - \frac{e_{11}}{e_{01} + e_{10} + e_{11}},$$

where e_{ab} is the number of elements where A has a value of a and B has a value of b. For example, if $A = \{1,0,1,0,0,0,0\}$ and $B = \{1,0,0,1,0,1,1\}$ then we have $e_{11} = 1$, $e_{01} = 3$ and $e_{10} = 1$, and then:

$$d_J(A,B) = 1 - \frac{1}{3+1+1} = 0.8.$$

This metric is thus particularly well-suited in our case if we choose A and B to be respectively the occurrences (2D) matrix of two terms i and j, as the presence of zeros does not mean anything (i.e. two terms presenting a 0 for the same sound and a same participant does not give any information about their relation). This is mainly why we preferred this distance over correlation measures.

Semantic proximity cases evaluation As stated earlier, we got interested in evaluating two different cases of semantic proximity.

The first case: the intra-subject semantic proximity, can be evaluated by computing the Jaccard distance for each participant S_k and for every pair of terms (i,j) on the whole set of stimuli:

$$J^{\text{intra}}_{i,j,S_k} = d_J\Big(\mathbf{M}(i,S_k,:),\ \mathbf{M}(j,S_k,:)\Big),$$

where d_J is Jaccard distance and $\mathbf{M}$ is the 3D occurrences matrix, $\mathbf{M}(i,S_k,:)$ representing the vector of occurrences of the i^{th} term for the participant S_k and all the stimuli of the dataset. This finally results in a 3D matrix $\mathbf{J}^{\text{intra}}$ of shape (number of terms $\times$ number of terms $\times$ number of participants).

The inter-subject semantic proximity case can be handled in a similar way by confronting the participants by pair (the target participant S_k with all the other participants S_l where $l \neq k$) for every pair of terms (i,j), computing their Jaccard distance on the whole dataset

of stimuli and then averaging with respect to these other participants S_l:

$$J^{\text{inter}}_{i,j,S_k} = \frac{1}{K-1} \sum_{\substack{l=1 \\ l \neq k}}^{K} d_J\Big(\mathbf{M}(i, S_k, :),\ \mathbf{M}(j, S_l, :)\Big),$$

where K is the number of participants. As for the intra-subject semantic proximity case, this results in a 3D matrix $\mathbf{J}^{\text{inter}}$ of size (number of terms $\times$ number of terms $\times$ number of participants).

Finally, as we are interested in evaluating the global semantic proximity of terms, we computed the arithmetic mean of the 3D matrices obtained for the two different cases presented above:

$$\mathbf{J} = 0.5\ (\mathbf{J}^{\text{intra}} + \mathbf{J}^{\text{inter}}).$$

In order to group terms into perceptual categories, the next step was thus to give this new distance matrix representing the semantic proximity of terms as an input to a clustering algorithm that will create classes by grouping terms with the minimal distances. To do so, we got interested into hierarchical agglomerative clustering (HAC).

Hierarchical clustering analysis Hierarchical agglomerative clustering is a bottom-up clustering approach aiming at grouping samples into clusters starting with individual samples and then gathering them one by one using a dissimilarity/distance metric until there is only one single cluster [Day and Edelsbrunner 1984]. All the grouping events (i.e. each time two clusters are grouped) are then summarized into a graph called dendrogram and to get the final clusters, it is necessary to find the optimal distance threshold given the task to be realized, see example in Figure 2.6. This is usually done by finding the value for which the grouping has created the main discontinuity in the homogeneity curve.

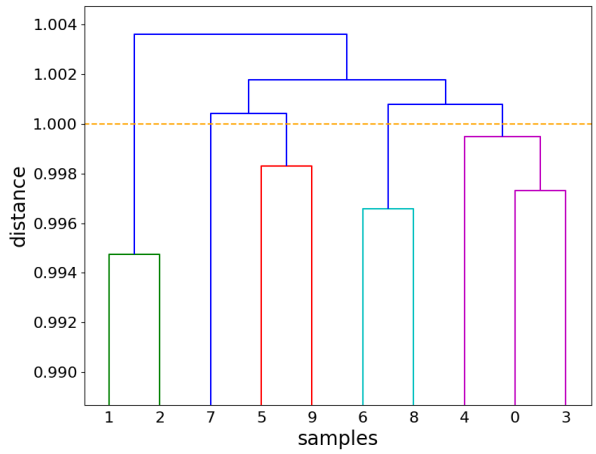


Figure 2.6: Example of a dendrogram. The distance threshold is represented by the orange dotted line, dividing the dataset into 4 clusters and one singleton.

There exist several ways to agglomerate the samples. During the process, each time two samples are grouped into a cluster, new distances to the other clusters and samples have to be computed. Different computation strategies are possible: minimizing the variance of

the clusters being merged, averaging the distance of the clusters being merged or always use either the maximum or the minimum distance.

In the context of this perceptual test, as we wanted the clustered samples to be the terms given by participants, we needed a 2-dimensional distance matrix of the size (number of terms × number of terms) to be used in the HAC algorithm as the initial distance matrix. Hence we averaged the 3D Jaccard distance matrix $\mathbf{J}$ with respect to the participants:

$$\overline{\mathbf{J}} = \frac{1}{K} \sum_{k=1}^{K} \mathbf{J}_{S_k}$$

where $\mathbf{J}_{S_k}$ is Jaccard distance matrix corresponding to participant S_k and K is the total number of participants. Then the agglomerative method had to be chosen. We investigated several of the proposed techniques, as well as different threshold values, and empirically selected the one that was optimizing the size of the clusters, avoiding to have very big clusters with no perceptual meaning nor semantic relevance and many singletons, i.e. a method minimizing the variance of the clusters being merged.

2.2.4.4 Obtained semantic categories

Once the final threshold value for the dendrogram was chosen, the HAC analysis resulted in a high number of clusters (98) grouping 8 terms in average. We computed the total number of occurrences of each cluster (by summing the total number of occurrences of every term in the clusters) and also the number of different participants that used at least one of the terms of the category, which we call "transversality measure". We finally selected the 5 classes with the largest number of occurrences, see results in Table 2.1. In Appendix A.4 can be found the whole classes created by the HAC.

Free Verbalization Test		
Grouped Terms	Occurrences	Transversality
['doux', 'resonnant', 'sourd', 'rond', 'etouffe']	143	49
['metallique', 'froid', 'aigu']	114	44
['agressif', 'electrique', 'desagreable', 'strident']	97	40
['chaud', 'grave', 'profond', 'sombre']	83	37
['vibrant', 'cuivre', 'bourdonnant', 'vrombissant', 'abeilles', 'essaim', 'insectes']	82	47

Table 2.1: Table of the 5 classes maximizing the frequency and transversality criteria.

The first observation we can make about these groupings is that they are meaningful in both the common sense and an auditory context. These groupings (in particular the one of "doux" with "sourd" and "étouffé"; "métallique" and "froid"; "agressif" and "strident" or "chaud" and "grave") are also consistent with other previous studies that explored the

semantic network of timbre description for other instruments (Piano: [Cheminée et al. 2005; Bellemare and Traube 2005]; Guitar: [Townsend 1996; Traube 2004; Lavoie 2013]; Violin: [Fritz et al. 2012]; Operatic voice: [Garnier 2003]).

Another notable observation is that among these 5 most frequent and transverse categories, all of them are related to the spectral content of the signal corresponding, to the second criterion of the spectro-temporal shape of the sound (the first criterion being the temporal dynamics) which represents the main underlying acoustical correlates of both causal and qualitative timbres [Castellengo 2015] as already introduced in Section 1.1.3. This result seems to highlight the importance of spectral content for characterizing sounds in comparison with the temporal dynamics.

However, two perceptual groups drew our particular attention as they are, in a way, also related to temporal dynamics of the sound. The first category (['doux', 'résonnant', 'sourd', 'rond', 'étouffé'] which would be close to ['soft', 'resonating', 'dull', 'round', 'muffled'] in English) is indeed somehow related to both spectral and temporal criteria, in particular with the presence of *résonnant* whose meaning is two-fold: depending on the background of the participants, it could be related to reverberations or echos, or in a completely different context, to a resonant filter. The second particular group is the third category: ['agressif', 'électrique', 'désagréable', 'strident'] equivalent to ['aggressive', 'electric/electrical', 'unpleasant', 'strident'] in English. This category is particularly interesting as it seems to be related, in the one hand, to both the temporal dynamics and the spectral content of the signal (e.g. a sound with a lot of energy in the 3-4 kHz frequency range together with a very harsh attack would probably be perceived as "aggressive" or "unpleasant") and on the other hand to aesthetic or value judgment and thus probably more likely to present more inter-subject variability than other types of perceptual categories.

What is important to note from these groupings, is that there is, at this point, no perceptual category linked to purely temporal characteristics of timbre and that consequently, some of the main timbre dimensions are missing, which is an issue we needed to tackle while selecting the final perceptual descriptors to be used for the rest of the study.

2.2.4.5 Verbal descriptors selection

As stated in Section 2.1, the objective of this perceptual test was to collect terms that are frequently and consensually used to describe synthesizer sounds. The second step is to select the most relevant/representative term for each of these category. The resulting set of terms will be used as the high-level control parameters (i.e. verbal descriptors) in the proposed music sound synthesizer. But before, we need to evaluate their relevance as control parameters. To that objective, they will first be used as labels of a second complementary perceptual test that will be presented in the next section (Section 2.3).

In the previous section, we selected the 5 most frequent and transverse perceptual categories. For each category, the prototype (i.e. the most representative term) was selected as both the most frequent and transverse term (i.e. used by the greatest number of participants), assuming it was thus the most consensual and/or "intuitive" verbal descriptors of the category, see Table 2.2. In order to select them, we thus had to find the appropriate tradeoff between frequency and transversality.

This resulted in a first selection of 5 terms: *doux*, *métallique*, *agressif*, *chaud* and *vibrant*.

30 Most Frequently and Transversally Used Terms		
Terms	Occurrences	Transversality measure
metallique	61	32
synthetique	50	18
doux	43	27
agressif	38	18
resonnant	38	18
froid	37	14
vibrant	32	19
chaud	30	15
grave	29	18
sourd	28	14
echo	26	15
electrique	26	15
long	26	8
gresillant	25	10
cuivre	24	18
souffle	24	15
evolutif	23	10
rond	23	13
desagreable	22	17
bruite	20	13
futuriste	20	11
aigu	18	15
etouffe	18	14
electronique	18	6
bourdonnant	16	12
klaxon	16	13
percussif	16	7
simple	16	8
strident	15	9
plat	14	6

Table 2.2: Most frequent terms given by subjects for the free verbalization test. The transversality measure corresponds to the number of different participants that used the term. The terms are sorted by the total number of occurrences.

In order to have a preliminary evaluation of the relevance of these chosen descriptors before distributing the perceptual test, we realized a small pilot test. From a correlation analysis using a t-test on the results of this pilot study, it was shown that *agressif* and *doux* (respectively "aggressive" and "soft" in English) were anti-correlated. To satisfy our criterion to avoid redundancy in the selected perceptual dimensions and avoid adding bias to the test by having terms that are too close or conversely opposite in the common sense, we thus decided to keep

the second most frequent and transverse term of this category which is *résonnant*.

As stated in the previous section, these descriptors do not cover all the dimensions of timbre and we had to tackle this issue. It seemed to us that the most important dimensions of timbre that were missing according to [Castellengo 2015] were the global temporal evolution of the sound (whether it is stationary, decaying or repeated like scratching for instance), the attack and also the noisiness of the spectral content.

To select appropriate prototypes for these new perceptual dimensions that we decided to add to the previously selected ones, we chose the most frequently and transversally used terms related to these dimensions: *evolutif*, *percussif* and *soufflé*, see Table 2.2.

It thus finally resulted in the following descriptors:

- *résonnant* ("resonating"),
- *métallique* ("metallic"),
- *agressif* ("aggressive"),
- *chaud* ("warm"),
- *vibrant* ("vibrating"),
- *evolutif* ("evolving"),
- *percussif* ("percussive"),
- *soufflé* ("breathy").

In order to avoid any additional bias in the results of the test, the wording of the prototypical terms finally selected as labels of the scales had to be decided very carefully. For example, the French word *vibrant* (equivalent of the English term "vibrating") can have some emotional sense, referring to something moving or to passion. We thus changed slightly the formulation to *qui vibre* ("which is vibrating") in order to remove this emotional aspect as we wanted it to only represent the concept of a vibrating/oscillating sound. Regarding the *résonnant* term, we decided to focus on its temporal aspect and thus chose to present the descriptor as *qui résonne* (in the same manner as for *vibrant*) so as to limit the bias produced by the two-fold meaning of this term. Finally, for the sake of consistency, we also made the choice of changing *évolutif* for *qui évolue*.

For more clarity, the final chosen scale labels were then organized in 3 different groups related to the criteria of [Castellengo 2015]: the terms related to spectral characteristics, the terms related to the temporal aspect of the sounds and finally the "other" terms, i.e. "aggressiveness"-related descriptor which is both linked to spectro-temporal characteristics and aesthetic judgment. This eventually resulted into 8 perceptually meaningful verbal descriptors, see Table 2.3, that are used as labels for the SD study, see Section 2.3.

2.2.5 Conclusion

To summarize, in this section, we presented the free verbalization perceptual test we conducted in order to collect terms that are frequently and consensually used to describe synthesizer sounds. In order to extract some of the terms that are the most consensual, transverse (i.e. used by several different participants) and that are semantically orthogonal, we realized a series of analyses on the results. After some pre-processing/cleaning of the data, we evaluated the semantic proximity between the different terms (taking both the intra-subject case and the inter-subject case into account) and then applied a HAC in order

	Verbal Descriptors
Spectral	Métallique Chaud Soufflé Qui vibre
Temporal	Percussif Qui résonne Qui évolue
Other	Agressif

Table 2.3: Table of the final labels for the scales.

to create consistent semantic categories. Interestingly, from this test was evidenced that for describing sounds, the participants used both terms that are commonly used for describing the timbre of classic musical instruments (e.g. *métallique*, *chaud*, *brillant* – respectively "metallic", "warm", "bright") but also some new terms that have, to our knowledge, never been reported for describing timbre (e.g. *spatial*, *robotique*, *distordu*, *explosif*, *saccadé*, *rétro-futuriste*)[12]. From these categories, we then performed a frequency and transversality analysis in order to select the perceptual dimensions as well as their prototypes. Surprisingly, from the 5 most frequent and transverse perceptual categories that were created, we observed that they were all related to the spectral content of the sounds which would indicate that this criterion is the most important to characterize synthetic timbres. The perceptual test finally resulted in the highlighting of 8 perceptual verbal descriptors: *métallique*, *chaud*, *soufflé*, *qui vibre*, *percussif*, *qui résonne*, *qui évolue* and *agressif* (English closest matches: "metallic", "warm", "breathy", "vibrating", "percussive", "resonating", "evolving" and "aggressive") that will be used as labels of the scales for the SD perceptual test that will be presented in the next section.

2.3 Second perceptual test: Semantic scales analysis

As already seen in Section 1.1.2.2, semantic differential methods aim at evaluating samples using perceptual scales labeled with verbal descriptors. Inspired by [Faure 2000] and [Ehrette 2004], we conducted such a semantic differential study using as scale labels the 8 terms highlighted by the free verbalization test presented in the previous section.

The purpose of this perceptual test was to evaluate the consensus of the 8 verbal descriptors extracted from the previous free verbalization test and get a quantitative evaluation of some sound samples using these scales. This would allow us to get a subset of stimuli from our Arturia dataset (see Section 2.2.2.1) evaluated on several perceptual dimensions in prevision for a possible (weak) supervision of neural models (see Chapter 4).

In this section, we will first describe the perceptual test with semantic scales we designed,

[12]Closest English translations: "space", "robotic", "distorted", "explosive", "jerky" or "retro-futurist".

then we will detail the analysis of the participants answers we realized and present the final results.

2.3.1 Participants

As for the free verbalization test, before taking the test the participants were asked to answer a short questionnaire aiming at collecting personal information about them and their listening conditions for the test. The questionnaire was identical to the one of the first test (see Section 2.2.1) with one additional question to know whether they participated in the free verbalization study or not.

This perceptual test was also distributed online using music technology related diffusion lists and personal networks. A total of 83 answers were collected but only 71 participants fully completed the test and evaluated all the stimuli. Among them, about 66% declared being musicians and 24% stated that they had no experience related to music (neither musician nor sound-related professional field). Most of them were under 34 years-old (62%) and as for the free verbalization study, the vast majority used headphones (80% against 17% using domestic speakers or studio monitors).

For more details about the questionnaire given to the participants or the collected information, please refer to Appendix B.1.

2.3.2 Stimuli

For this new perceptual study, the exact same database of stimuli as for the free verbalization test was used, i.e. the Arturia dataset. Indeed, as semantic scaling focuses also in better understanding the qualitative aspect of timbre, the samples from the dataset fulfill all the required criteria, see Section 2.2.2. However, new samples had to be chosen as we needed to adapt the number of stimuli that would be presented to participants and their acoustic characteristics.

2.3.2.1 Training stimuli versus main phase stimuli

In order to evaluate the consensus of the scales and be able to quantitatively evaluate samples using the evidenced perceptual dimensions, two different aspects were to be analyzed. First, it was necessary to analyze the consistency of the participants while using the scales to evaluate whether they could be clearly understood and consistently used. Then the second aspect was to evaluate how the use of these scales was consensual among participants.

We decided to perform the test in two phases: a training phase to allow participants to get familiar with the sounds and have time to get an idea about what the scales were representing for them using a first set of stimuli; and the main phase using another set of stimuli which will be used at the end for the quantitative evaluation of samples using the perceptual dimensions. In order not to add bias between participants, the training stimuli were the same for all and always presented in the same order. Plus, to evaluate the consistency of each participants, we needed them to evaluate at least some samples twice. To do so, we chose to reinsert some training stimuli during the main phase, this time randomly selected from the training set.

As for the previous study, the test was designed to last about 20 minutes. Participants were thus successively presented with 40 sounds to evaluate. To get a good balance between

the training and the main phases, it was decided to have a training phase of 10 samples and a main phase of 30 stimuli. Plus, among the 30 stimuli of the main phase, 5 samples randomly chosen from the training set were reinserted. This process is illustrated in Figure 2.7.

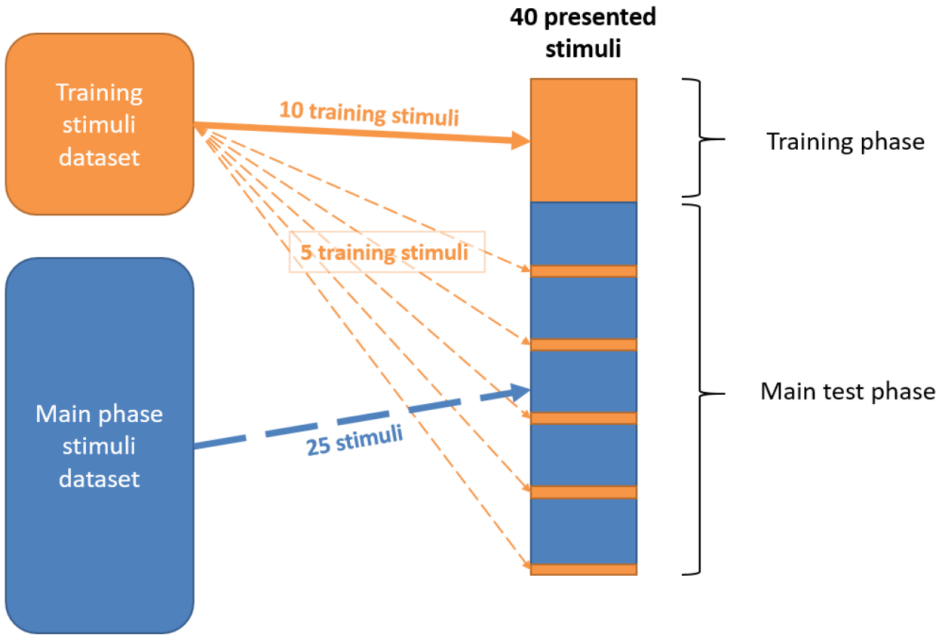


Figure 2.7: Illustration of the samples distribution for the perceptual test with semantic scales. The dotted arrows represent random selection whereas the plain arrow illustrates the sorted selection of all the stimuli of the dataset.

To obtain sufficient statistical power we needed at least 20 different analyzed evaluations per samples and we expected to be able to reach around 50 participants. As previously defined, the number of training stimuli was set to 10. Given the fact that participants were evaluating 25 different main phase stimuli each, the number or samples for the main dataset to be evaluated was thus set to 70.

2.3.2.2 Samples selection

To select these 80 new stimuli (10 for the training and 70 for the main phase), as for the previous test, we wanted them to be as representative of the acoustic space as possible and thus not to be selected in a purely random manner so as to be very different from each other. In the next paragraphs will be presented the different stages needed for the selection of the audio stimuli.

Descriptors extraction For this new perceptual test, we also extracted a set of audio descriptors to get an objective representation of the acoustic space. However, we noticed that for the previous test we had (almost) only spectral features which is not accurate enough in particular as some of the perceptual scales are directly linked to temporal characteristics of the sounds. We thus needed to add some temporal features (actually 5), resulting in 12 different descriptors:

- attack time,
- attack slope,
- decay time,
- jitter,
- shimmer,
- mel-frequency cepstral coefficients (MFCC),
- spectral centroid,
- spectral bandwidth,
- spectral contrast,
- spectral flatness,
- spectral roll-off,
- zero-crossing rate (ZCR).

The temporal descriptors are scalars in opposition to the spectral descriptors that are vectors. In order not to get an imbalance between spectral and temporal features, we decided not to extract spectral descriptors on all the frames but only at 3 different moments in time: 25%, 50% and 75% of the signal duration.

As for the free verbalization test, the spectral descriptors were extracted using the *librosa* Python package [McFee et al. 2015]. The *jitter* and *shimmer* descriptors were extracted using the *Parselmouth* Python interface for the *Praat* software [Jadoul et al. 2018] and the *attack time*, *attack slope* and *decay time* were extracted using the Matlab *MIRtoolbox* [Lartillot et al. 2008].

For more information about the descriptors, the reader is referred to [Peeters et al. 2011], [Peeters 2004] or the cited papers.

Random selection by clustering Once the descriptors extracted, we applied the exact same methodology as we did for the free verbalization test for selecting the samples to be presented to the participants, i.e. we randomly selected samples from clusters in the acoustic space formed by performing a k-means algorithm (using $k = 70$ for selecting the 70 samples from the main dataset), see Section 2.2.2.2. Afterward, some manually operated adjustments were made, with the help of our t-SNE visualization and listening tool, in order to insure that the final stimuli were as far from one another as possible to be good representatives of the overall acoustic space.

Concerning the 10 training samples[13], they were all manually selected after informal perceptual listening so that they covered the range of the scales and were as different from one another as possible.

2.3.3 Protocol of the study

As for the free verbalization study, the test was designed to be taken online. After a quick explanation about the task to perform, the 40 stimuli were presented successively to the participants. They could listen to each sound sample as many times as necessary and then evaluate it by using the 8 proposed semantic scales: *métallique*, *chaud*, *soufflé*, *qui vibre*, *percussif*, *qui résonne*, *qui évolue* and *agressif*.

[13]The training samples are available for listening in the companion webpage.

Since we did not conduct a detailed semantic analysis on the terms evidenced by the first perceptual test, and since we do not know exactly their antonyms, we decided to use unipolar scales going from *pas du tout* ("not at all" corresponding to "-1" for us) to *extrêmement* ("extremely" corresponding to "+1" for us) with an initial cursor position set to the middle (corresponding to "0" for us) in order to reflect the notion of "neutrality" [14]. This choice can have consequences but was, to our opinion, the one with the lowest influence on the answers. These scales have been chosen to be continuous rather than discrete in order to compute basic statistics directly from the results (e.g. mean or standard deviation).

No definitions nor explanations about the labels of the scales were given to the participants, they had to evaluate the stimuli given their own understanding of the verbal descriptors. However, at the end of the test (i.e. after evaluating the 40 sounds), they were asked to express themselves freely in a written form about the different scales, how they used them and what they meant for them. This final task was added to the test in order to collect more information about the context and the way participants used and understood the scales. This additional information could thus be useful to analyze whether there exists a clear consensus on their use.

In Figure 2.8 and Figure 2.9 are depicted screenshots of respectively the main perceptual test pages and the free expression page.

For more details about the protocol explanation given to the participants, the reader is referred to Appendix B.2.

2.3.4 Results analysis

2.3.4.1 Objectives of the analysis

As already stated before, the main objective of this test was to evaluate how consensual were the selected verbal descriptors and get a quantitative evaluation of some sound samples in prevision for the training of a weakly supervised model (see Chapter 4). But in order for the perceptual evaluations on the semantic scales to be meaningful and the quantitative evaluations to be relevant and accurate, it was very important to first make sure that participants used the scales consistently (high intra-subject consensus) and that they were understood and used similarly (high inter-subject consensus or agreement) [Kreiman et al. 1993].

In the next sections we will detail the methods we used in order to test the reliability of the participants evaluations. It is important to note that here the analysis is sequential and that the results of each step serve as a starting point for the analysis of the following step.

2.3.4.2 Results pre-processing

As usual, before analyzing the results it is important to start by cleaning the data. As already stated in the first chapter, participants may use the scales differently [Kreiman and Gerratt 1998]. Some will use the extrema whereas others will always stay rather close to the middle. It can thus be interesting to normalize the evaluations with respect to the use of scales of each participant. Indeed, we observed that not all the participants used the extrema.

[14] For example to evaluate a sound that is not extremely aggressive but that cannot be considered as "not aggressive at all".

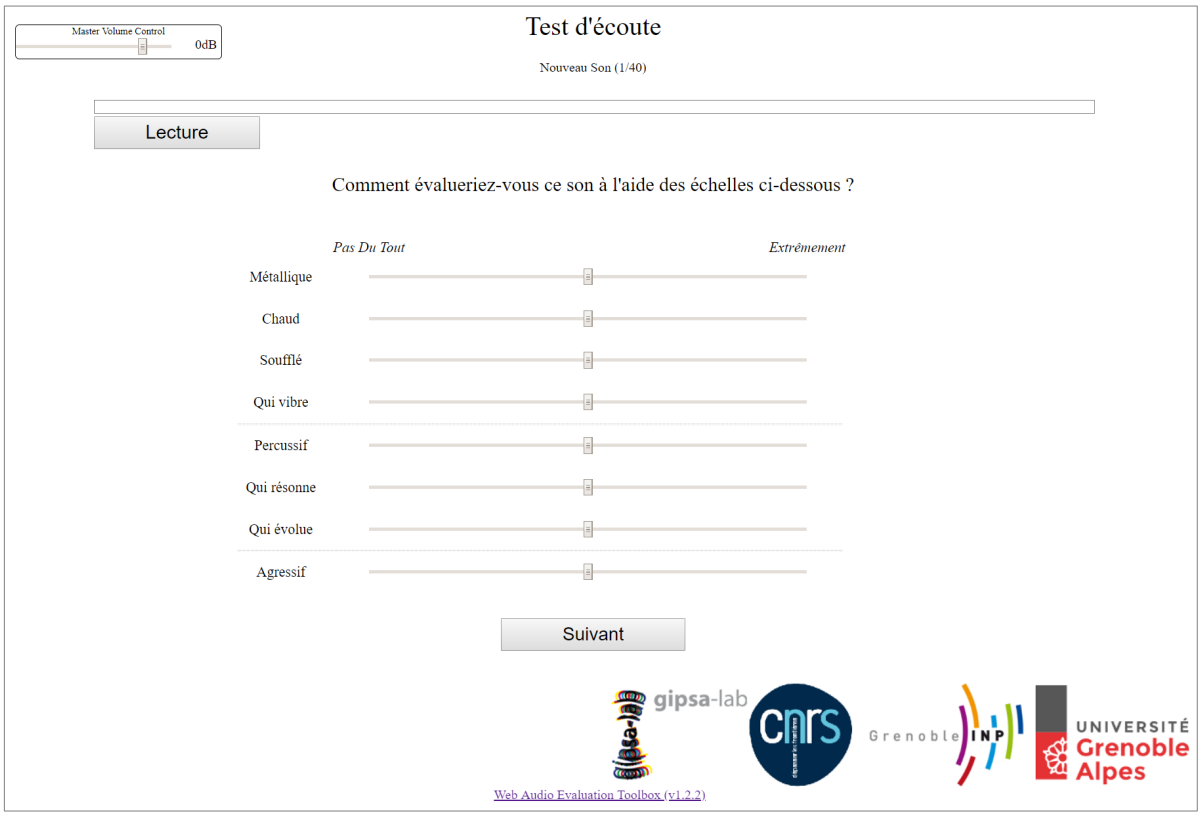


Figure 2.8: Screenshot of the interface of the test using semantic scales. Here again, the study was implemented using the Web Audio Evaluation Tool [Jillings et al. 2015].

Figure 2.9: Screenshot of free expression test page on semantic scales.

In our test, 14 out of 71 participants used less than 90% of the whole range of the scales. However, except one participant that used only about half the scale, all participants used more than 75% of the scales.

To normalize the use, we decided to keep the neutral position in the middle (i.e. the "0" value) and to set the maximum absolute value used by the participant to 1, thus optimizing the range of values that were used. Another choice here could have been to deal with the 8 scales as actually 16 half-scales and separate the positive and negative evaluations on each scale, thus optimizing each half-scale separately. However, except for 3 participants that did not explore much (even not at all for one) the negative values of the scales, there was no difference of use between positive and negative parts, it was thus not relevant to split the scales into two and so we kept them whole.

2.3.4.3 Intra-subject consensus

Once the data cleaned, the first step towards the analysis of the results of the study was to evaluate the intra-subject consensus, i.e. how the evaluations of a participant were consistent at two different times of the test for a same sound. The aim of such an analysis was to remove potential participants that did not use the scales consistently during the test and who could thus degrade the results of the study.

Correlation computations The intra-subject consensus was evaluated for each participant by comparing their evaluation of the 5 stimuli duplicated between the training phase and the main phase of the test. Hence, for each participant S_k, as the scales are continuous, we can quantify this consensus by computing, for each scale separately, the Pearson's r correlation between the evaluations [Kreiman et al. 1993]:

$$C^{\text{intra}}_{S_k,d} = \text{corr}\left(\mathbf{v}^{\text{main}}_{S_k,d}(\mathbf{t}_{S_k}), \mathbf{v}^{\text{train}}_{S_k,d}(\mathbf{t}_{S_k})\right),$$

where $\mathbf{v}^{\text{phase}}_{S_k,d}$ states for the evaluation vector of the participant S_k on the scale d during the stage *phase* (*train* for training or *main* for the main phase of the test) and $\mathbf{t}_{S_k}$ represents the set of training samples that appeared twice during the test for the participant S_k, corr being the function computing the correlation coefficient between the two rating vectors.

Results The resulting two-dimensional $\mathbf{C}^{\text{intra}}$ matrix is represented in Figure 2.10 where the different scales are displayed along the x-axis and the participants are along the y-axis. It can be observed that only one participant can be considered as generally irregular in the use of the scales. Most of them were consistent with themselves and gave evaluations that correlate well between the training and the main phases. However, some participants seemed to have had some trouble using specific scales. Indeed, we can see in Figure 2.10 that some coefficients are negative or null (red to white in the figure), meaning that the results are poorly correlated or even anti-correlated between the training and the main phase, which indicates that the participants had no clear conception if this term and answered randomly, or that their understanding of the scale evolved from the training phase to the main phase of the test. The examination of their comments at the end of the test did not really enabled us to understand this poor intra-subject consensus. Except a few participants who expressed trouble in understanding the concept of a particular scale or that they used

it for two different characteristics of the sound[15], most of them gave relevant comments. Interestingly, one participant expressed a lack of understanding of a particular scale and, as a matter of fact, did not use it during the whole test.

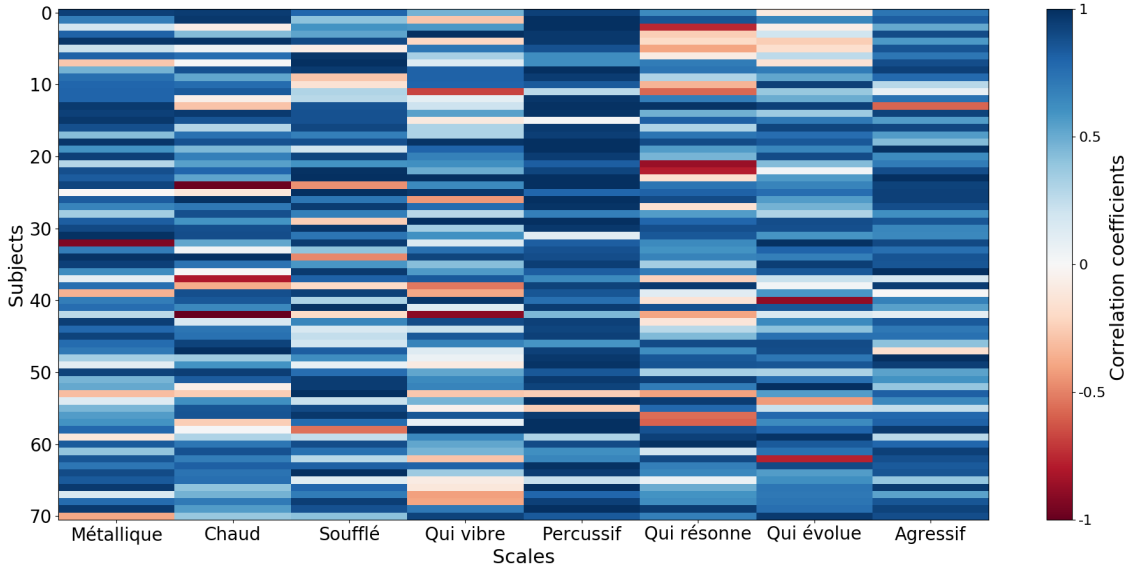


Figure 2.10: Correlation coefficients for the evaluation of the intra-subject consensus.

Scale-wise participants selection In order to avoid inconsistent results from the participants who had trouble understanding some scales, we considered only the data of the participants who showed a correlation coefficient greater than 0.5. This value of 0.5 was empirically chosen as a tradeoff between the consistency of the answers and the number of participants kept for each scale for the rest of the analysis, for more details see Figure B.2 in Appendix B.3.1. The number of participants selected among the 71 for each scale are reported in Table 2.4. From these results it can clearly be observed that some scales are easier to understand or at least to get a personal representation from than others (e.g. *percussif* or *agressif* in comparison with *qui vibre*).

2.3.4.4 Inter-subject consensus

Once the participants who poorly understood the scales were removed, the next step was to evaluate the inter-subject consensus, i.e. the consistency of the use of scales between the different participants who took the test.

Correlation computations The inter-subject consensus can be quantified, similarly to the intra-subject consensus, by computing the Pearson correlation coefficient between the evaluation of each pair of participants (S_k, S_l) with $l \neq k$, on the subset of sounds commonly

[15]For example, for the *soufflé* ("breathy") scale, a participant explained that he or she used the scale to evaluate both how flute-like the sounds were and how slow the attack of the sounds were, which could explain the inconsistency.

Verbal Descriptors	Number of Selected Participants	Resp. Percentage
Métallique	49	69.0%
Chaud	46	64.8%
Soufflé	47	66.2%
Qui vibre	35	49.3%
Percussif	62	87.3%
Qui résonne	41	57.7%
Qui évolue	48	67.6%
Agressif	58	81.7%

Table 2.4: Table of the number of selected participants after the intra-subject consensus analysis for each scale.

rated by those two participants during the second main phase of the test $\mathbf{o}_{S_k,S_l} = \mathbf{r}_{S_k} \cap \mathbf{r}_{S_l}$ for the scale d (representing from 4 to 29 stimuli in common):

$$C^{\text{inter}}_{S_k,S_l,d} = \text{corr}\left(\mathbf{v}^{\text{main}}_{S_k,d}(\mathbf{o}_{S_k,S_l}), \mathbf{v}^{\text{main}}_{S_l,d}(\mathbf{o}_{S_k,S_l})\right).$$

This resulted in a set of 8 two-dimensional matrices $\mathbf{C}^{\text{inter}}_d$ of size (number of participants for scale d $\times$ number of participants for scale d), one for each semantic scale (these matrices are depicted in Appendix B.3.2).

In order to evaluate the inter-subject consensus for each scale, we computed the histogram of the inter-subject correlation coefficient $C^{\text{inter}}_{S_k,S_l,d}$ for each scale d, see Figure 2.11. From this figure, it can clearly be seen that, on average, participants were in agreement with each other, showing a mean inter-subject correlation value significantly greater than 0 for every scale, the minimum value being 0.23 for the *qui vibre* scale. It can also be noticed that, accordingly with the observations on the intra-subject correlation matrix, some of the scales appear less consensual and present lower mean values (e.g. 0.23 for *qui vibre* and *qui résonne* or 0.31 for *soufflé*) compared to others (e.g. 0.56 for *percussif* or 0.51 for *agressif*).

Hierarchical clustering analysis To investigate further the reasons for these differences in participants agreement, we applied a HAC algorithm scale-wise on these inter-subject correlation coefficients. The purpose of this step was to group the subjects into different clusters depending on their use of the scale. This way, if there exist several conceptions of the scales among participants, they should be represented as different clusters.

As explained in Section 2.2.4.3 the HAC algorithm takes a distance matrix as input for performing clustering. In order to convert the inter-subject correlation matrices into distance matrices, we computed $D^{\text{inter}}_{S_k,S_l,d} = 1 - C^{\text{inter}}_{S_k,S_l,d}$ where $C^{\text{inter}}_{S_k,S_l,d}$ is the previously computed inter-subject correlation coefficient between participants S_k and S_l for scale d ($l \neq k$) and $D^{\text{inter}}_{S_k,S_l,d}$ are the corresponding inter-subject distance coefficients leading to eight 2-dimensional matrices $\mathbf{D}^{\text{inter}}_d$. This is a classically used method for converting correlations into distances [Kaufman and Rousseeuw 1990]. Indeed, the initial correlation coefficients that lie in $[-1, 1]$

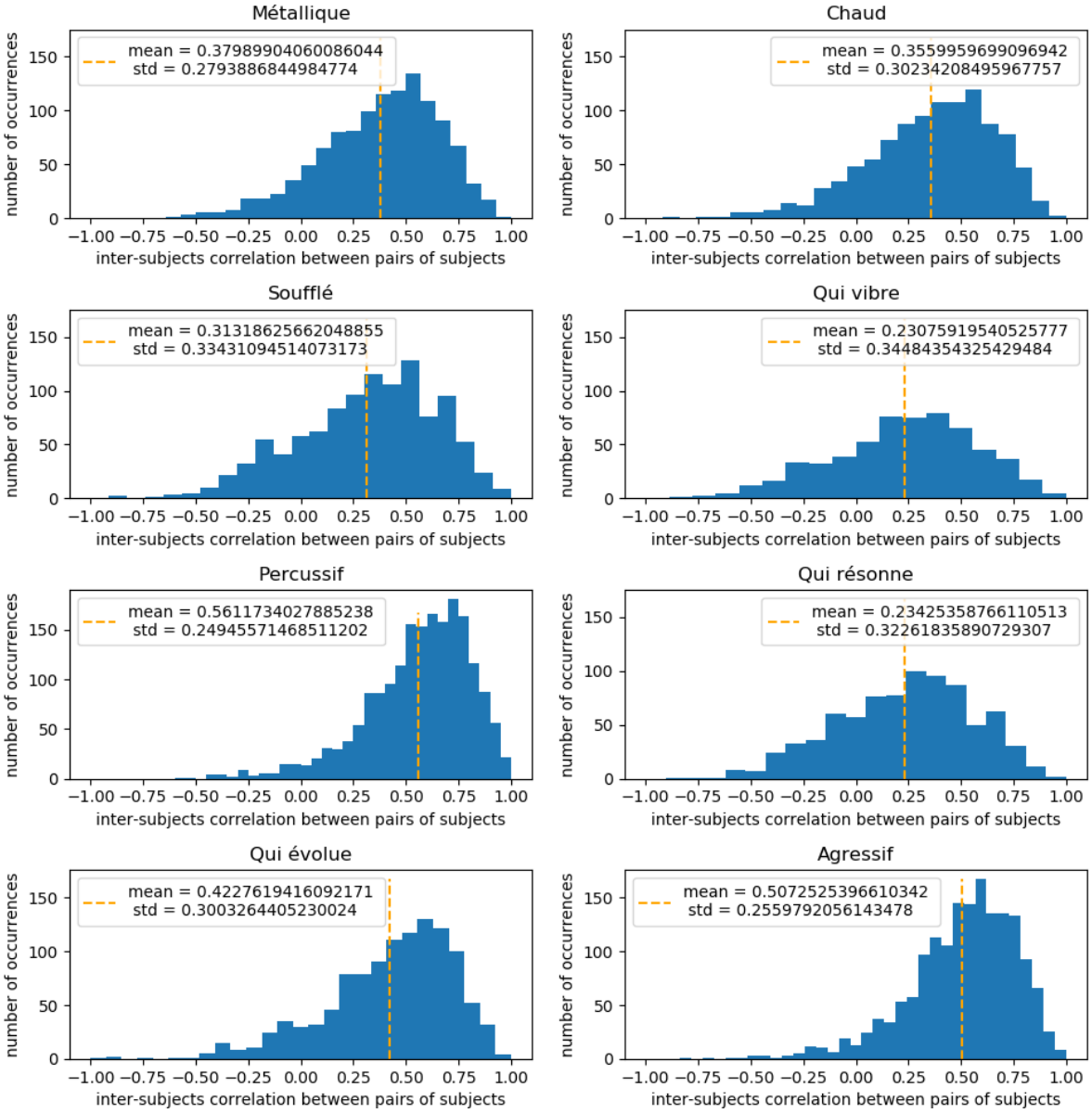


Figure 2.11: Histogram of inter-subject correlation coefficients given the semantic scales.

will be converted to distances in $[0, 2]$ where the 0 corresponds to the perfect correlation case (with a correlation coefficient of 1) and the 2 to the anti-correlation (i.e. a correlation coefficient of -1) which is well suited in our case as we want to group participants that used the scale similarly.

For each scale d we then applied a HAC initialized by $\mathbf{D}_d^{\text{inter}}$ using the same parametrization as previously (see Section 2.2.4.3). In order to find the optimal threshold for each dendrogram, we selected the average distance value that corresponded to the grouping of clusters that created the most important gap of distances, i.e. a change in the homogeneity of the clusters. The resulting dendrograms are presented in Figure 2.12 and the participants repartition per scale is given Figure 2.13. For each scale, the participants were then divided into different

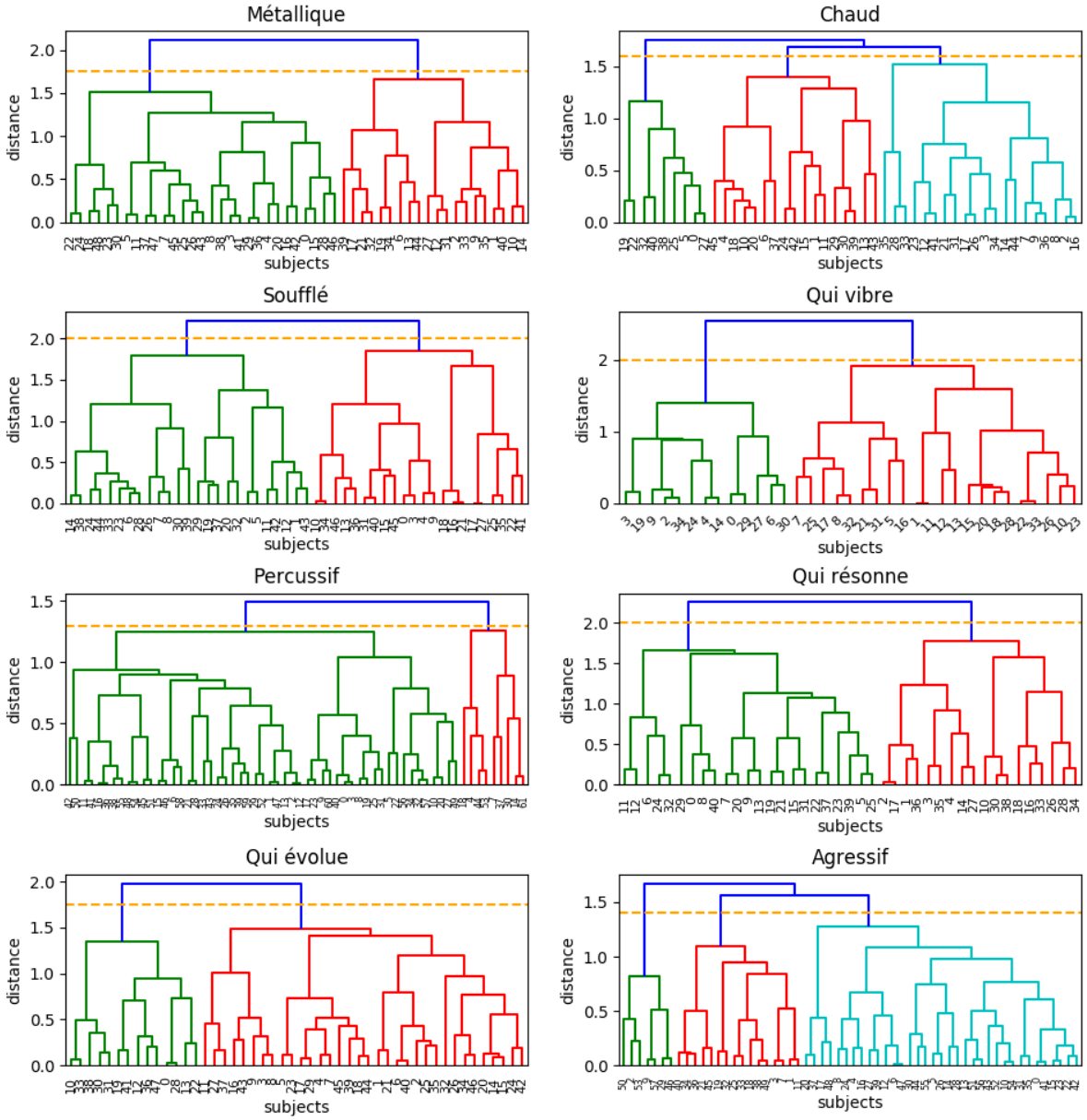


Figure 2.12: Resulting dendrograms for the inter-subject correlation coefficients for each semantic scale.

clusters, from 2 to 3 groups (for *chaud* and *agressif*).

Clusters analysis From the threshold values illustrated in Figure 2.12, we can once again clearly observe a difference between the scales. Indeed, some scales present threshold that are much lower than other (e.g. *percussif* with a value of 1.3 or *agressif* with a value of 1.4 in contrast to *soufflé*, *qui vibre* or *qui résonne* which present a threshold of 2) indicating more homogeneous clusters.

From Figure 2.13 we can see that some distributions of the participants among the different

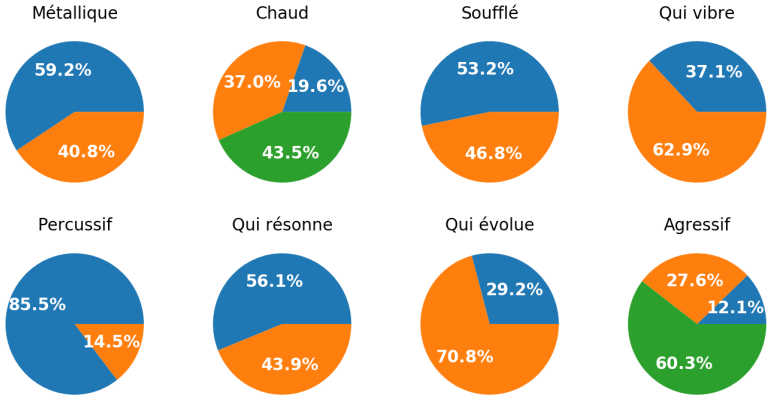


Figure 2.13: Participants repartition for the different clusters resulting from the HAC analysis for each scale.

groups are almost equally balanced (e.g. *soufflé* or *qui résonne*) whereas some present one clearly evidenced majority group (e.g. *percussif* or *qui évolue*). In order to try to explain these variations and better understand these groupings and the possibly different conceptions of the terms associated to each cluster, we plotted for each sound of the dataset the standard deviation of the given evaluations in function of their mean rating value as it was done in [Ehrette 2004], see Figure 2.14.

From Figure 2.14, the first observation worth commenting is the "bridge" – ∩ – shape of the scatter plot. This peculiar distribution that is present for every scale shows that evaluations are much more consensual in the extrema than in the midrange which corresponds to the more neutral zone of the scales. This result could have been expected as it is most likely that different persons agree on their prototypical representation of what is an "extremely metallic" or "not metallic at all" sound rather than on a sound that is "not very metallic but still a little metallic". This also confirms prior results that were obtained for other perceptual studies on very different topics but using perceptual scale ratings [Kreiman et al. 1993; Ehrette 2004]. More importantly, from this figure we can see that, for all the scales, the standard deviation for the extrema mean ratings gets low (in particular for some scales such as *agressif* or *percussif*) which indicates the existence of a consensus within the group, i.e. that participants share a same conception of the term. Although this phenomenon is observable for every scale, the differences between them are still very present in this graph. Indeed, some of the scales present clear ∩-shape with a very low standard deviation for the extrema (again *agressif*, *percussif* or *chaud* depending on the group) whereas for others the distribution is less obvious and the standard deviation for the extrema is higher (e.g. *qui vibre*, *qui résonne* or *métallique*). Interestingly, these observations on subsets of participants are in agreement with the results previously presented concerning the intra-subject consensus and inter-subject correlations computed on the entire set of participants, i.e. that some scales (*qui vibre* and *qui résonne*) appear less consensual than others.

Then, so as to further investigate the discrepancy between clusters, we tried to qualitatively compare the 10 prototypical samples of the different clusters for a same semantic scale by listening to them, i.e. to the 5 sounds with the highest mean values and the 5 sounds with the lowest mean values (which are the also the most consensual ones). When comparing the

clusters, even though the prototypical samples differ most of the time, they globally lie in the same region (i.e. half part) of the scales. In other words, if a sound has been rated as a prototype of a scale, it is very infrequent to have ratings that lie on the opposite side of the scale for the other clusters. This evidenced slight variations in the conception of the terms between the clusters but that there still exists some form of consensus, which is very positive from the perspective of using these descriptors as controls of a synthesizer.

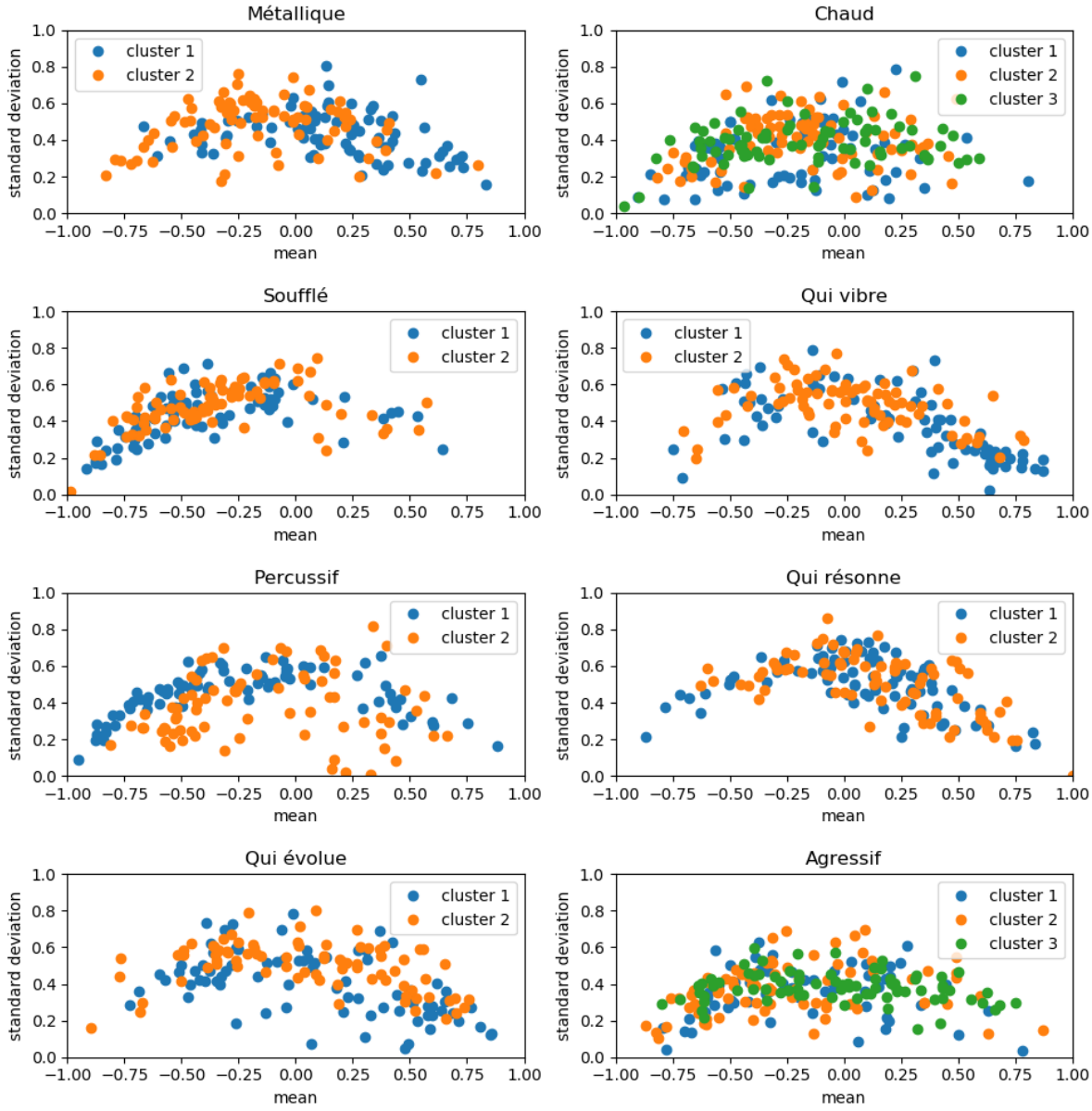


Figure 2.14: Scatter plot of the standard deviation of the ratings of each sound in function of their corresponding mean value for each cluster depending on the perceptual scale.

As a final step towards understanding the variations of the participants given the scales, we examined the comments that were given at the end of the perceptual test for each participant and compared the used keywords for the different clusters. Once again, we could not evidence

significant differences between clusters. Some interesting results for some scales are worth noting however. First, by comparing the comments of participants, a dual aspects for both *qui vibre* and *qui résonne* scales has ben evidenced. In the first case, this duality consisted in a distinction between a low-frequency oscillation of the signal (vibrato, tremolo) and a high-frequency modulation of the timbre (sizzle, distortion). For the second scale, as already explained in Section 2.2.4.4, the difference was between the presence of echo or reverberation in the sound, or in a totally different manner, the presence of the distinctive characteristics of a resonant filter[16]. This dual aspect could have had an important impact on the results as a few participants explained in their comment that they used both possible understandings of the scales to rate the sounds. Conversely, all participants agreed unanimously on the sudden, short and brutal nature of the very *percussif* sounds and the unpleasant, strident and loud aspects of the *agressif* description of a sound. Surprisingly, the participants were very consensual on this latter term that is more related to subjectivity as it refers to personal judgment. Concerning the other scales, the comments were rather consensual and the used keywords did not differ much from one participant to another independently of the group they have been assigned to. A possible explanation for these produced results might be that these scales refer more to the personal experience of the different participants and thus their evaluations on a particular scale depend on the prototypical reference of their cognitive representation of this scale which varies a lot from participant to participant.

2.3.4.5 Final label vectors computation

Finally, once the participants that poorly understood the scales were removed and the clustering of the remaining participants done, the last step of the analysis was to quantify the subset of samples of the Arturia dataset that were evaluated during the perceptual study.

In order to get relevant and consistent projections of the samples into the perceptual space constituted of the 8 dimensions represented by the semantic scales, we decided, for each scale, to focus on the participants who were associated to the clusters with the largest number of members, see Figure 2.13.

From the evaluations of these particular participants, we took the mean value of the ratings of each sound on the scale and constituted the annotated dataset that will be used later for regularizing the neural model, see Chapter 4.

2.3.5 Conclusion

To summarize, in this section we presented the second perceptual test we realized in order to evaluate the consensus of the 8 verbal descriptors evidenced by the free verbalization study so as to analyze whether they could be suited as control parameters of a synthesizer, and get a quantitative evaluation of some sound samples using these scales in prevision for the weak supervision of a deep learning model. To perform the consensus analysis, we first evaluated the intra-subject consensus, i.e. how the evaluations of a participant were consistent. From this evaluation we could evidence that some participants could not get a clear conception of some of the terms and in order not to degrade the results, we decided to remove them

[16]This observation indicates that changing the syntactical form of the term was not enough to guide participants towards the evaluation of this term with regards to its temporal aspect.

scale-wise (i.e. only for the poorly understood scale) for the rest of the analysis. We then performed an inter-subject consensus analysis to evaluate if there existed a shared conception of the terms among the participants and grouped them accordingly. Finally, we evaluated quantitatively on each scale the subset of the Arturia dataset that was used for the perceptual test by averaging the ratings of a group of participants that shared a same conception and use of this scale.

A few observations clearly emerged from the different analyses we conducted from the results of the semantic scales perceptual study. First, from both intra-subject and inter-subject consensus analyses, it can be seen that there exist discrepancies in the way participants understood and used the different scales, highlighting the difficulty to get a clear conception for some scales in comparison to others. Very interestingly, the level of consensus of each scale is consistent across both analyses. Indeed, we can distinguish three different types of scales. The consensual scales: *percussif* and *agressif* where participants were both very consistent with themselves and with other participants. This can be evaluated by the number of remaining participants (see Table 2.4), the mean value of inter-subject correlation coefficients that is above 0.5 with a low standard deviation (see Figure 2.11) or the low average distance between clusters selected as threshold for the dendrograms (see Figure 2.12). The second type is the least consensual scales: *qui vibre* and *qui résonne*. And finally the other scales that present a decent agreement among participants, i.e. *métallique*, *chaud*, *soufflé* and *qui évolue*. Then, it is worth noting that, although some scales appear less consensual and show more variability in the ratings given by the participants, there still exists some consensus between the clusters of participants. In view of using these descriptors as controls of a synthesizer however, we have to be very careful about the dual aspects of both the *qui vibre* and the *qui résonne* scales which could lead to unexpected results for the potential users.

2.4 Conclusions and perspectives

In this chapter were presented and detailed two perceptual studies.

The first perceptual study was a free verbalization test whose aim was to evidence frequently used terms to characterize synthesizer sounds. From these terms, we selected 8 verbal descriptors that constitute now the 8 dimensions of our perceptual space: *métallique*, *chaud*, *soufflé*, *qui vibre*, *percussif*, *qui résonne*, *qui évolue* and *agressif* (English closest translations: "metallic", "warm", "breathy", "vibrating", "percussive", "resonating", "evolving" and "aggressive"). These descriptors extracted from a dataset of purely synthetic sounds are somehow related to terms that were evidenced as frequently used by other studies on orchestral instruments. For example, *métallique* or "metallic" was evidenced in [Traube 2004] (specifically on guitar sounds) or [Faure 2000; Cheminée et al. 2005] together with *percussif* or *chaud* ("warm" for English studies). The latter has also been pointed out in [Zacharakis et al. 2014] in which the time-related dimension *qui évolue* ("evolving") was identified as another important perceptual dimension. In [Faure 2000], the terms *soufflé*, *vibré* (which is very close to *qui vibre*) and *résonnant*, that we chose to present to the participants as *qui résonne*, did already come out of a free verbalization test on samples from an orchestral dataset. Finally, *agressif* was also a term that has been brought to light by several studies but, most of the

time, through its opposite[17] *doux* or "soft" [Faure 2000; Traube 2004; Cheminée et al. 2005; Zacharakis et al. 2014]. However, to our knowledge, no previous study had already identified the most frequent and consensual terms used to describe musical synthesizers sounds. This constitutes the first contribution of this thesis.

In the second perceptual test, we asked the participants to evaluate sound samples using scales labeled with these 8 selected perceptual verbal descriptors. This last study allowed us to analyze how these dimensions can be consensual or not, and to evidence possible variations in the meaning of the terms or at least in the way participants understood and used them. A clear asymmetry between the scales emerged from the results showing a discrepancy in both the consistency of evaluations of each participants (i.e. participants were consistent in their evaluations using some scales and lack of consistency while rating with others) and inter-subject agreement (i.e. participants largely agreed on some scales whereas others were not very consensual). The use of these scales as control parameters of a possibly commercialized synthesizer is thus largely open to questions, especially as it has been shown that some scales can be understood in several very different manners (i.e. *qui vibre* and *qui résonne*). By wisely selecting participants for each scales in order to keep only consensual ratings for the sounds samples, we managed to get a quantitative description of these sounds within this perceptual space. We thus created a perceptually annotated dataset of synthesizer sounds, which is another contribution of this thesis.

In order to analyze deeper the results of these perceptual studies and to evaluate if the dimensions that emerged are genuinely adapted for controlling a synthesizer, it would be interesting to conduct a correlation or redundancy analysis of the scales in order to evaluate whether the selected descriptors are "independent". Indeed, if we observe that each time a sound is considered as extremely *chaud* it has also been rated as not *agressif* at all for example, then we have two controls that will encode somehow the same information which is something we want to avoid.

Also, just as in [Garnier et al. 2007] or [Zacharakis et al. 2014], it could be interesting to search for the underlying acoustic correlates of the perceptual dimensions we evidenced for synthesizer sounds and compare them to the results that were obtained for operatic voices and/or orchestral instruments.

[17] As proven anti-correlated by a t-test, see Section 2.2.4.5.

Chapter 3

Unsupervised extraction of a high-level control space for audio synthesis

Contents

3.1 Motivation **60**
3.2 Analysis-transformation-synthesis methodology **60**
3.2.1 Global methodology 60
3.2.2 AE-based models 62
3.2.2.1 Linear AE: PCA 62
3.2.2.2 AE and deep AE 62
3.2.2.3 Recurrent AE 63
3.2.2.4 Variational AE 63
3.2.3 Data representation 67
3.2.3.1 Representation in the time-frequency domain 67
3.2.3.2 Statistical modeling and implications for VAE training 68
3.2.3.3 Phase spectrogram reconstruction 70
3.3 Comparative study of different AE-based models on two datasets **72**
3.3.1 Datasets 73
3.3.1.1 NSynth dataset 73
3.3.1.2 Arturia dataset 73
3.3.2 Data pre-processing 74
3.3.3 AE-based models implementations 74
3.3.4 Experimental results 75
3.3.4.1 Analysis-synthesis 75
3.3.4.2 Cross-correlation of latent dimensions 82
3.3.4.3 AE-based sound morphing 86
3.3.5 Conclusion 89
3.4 Max/MSP prototype **89**
3.4.1 Main principle 89
3.4.2 Qualitative observations and comments 91
3.5 Conclusion **91**

In the previous chapter, we evidenced 8 perceptually relevant dimensions for characterizing synthetic sounds, which was an answer to the first main challenge of this research project. In this chapter, we will focus on the second main challenge which consists in finding a suited machine learning model that would allow to generate high-quality synthetic sounds while offering a new high-level control space for synthesis, optimistically somehow linked to the evidenced perceptual dimensions.

After introducing the motivation for investigating deep models, we will detail the global methodology applied throughout this chapter. Two comparative studies between several machine learning models we conducted on two different datasets will then be presented along with a non real-time Max/MSP prototype we developed to qualitatively validate our methodology and assess the link between the high-level space extracted by the various implemented models and the perceptual dimensions evidenced in Chapter 2.

3.1 Motivation

As stated in the introduction of this manuscript, one of the main challenges of this research project was to find a well-suited machine learning model and algorithm to extract a high-level representation space with interesting interpolation and extrapolation properties from a dataset of sounds. It would allow to navigate the space continuously and smoothly while generating new timbres within the sonic space of the dataset but also beyond its limits, exploring thus new sonorities. In addition, if this algorithm could extract a space that is well-adapted for control, i.e. a low-dimensional space with decorrelated dimensions, this would be a great advantage.

As a starting point, in order to answer this main challenge, we focused on unsupervised methods as it does not imply restrictions with regards to the data compared to supervised learning which requires gathering a labeled dataset, which could be hard and expensive to manage. We thus tried to answer the following questions: is it possible to extract automatically a high-level representation space with interesting properties directly from a low-level representation of signals? Can this space be suitable for controlling audio synthesis while presenting dimensions possibly linked to the perceptual dimensions evidenced in Chapter 2 or at least perceptually relevant?

Following the path of previous studies interested in using (deep) machine learning for audio synthesis, see Section 1.3.3.1, we focused on AE-based models[1]. In the next section we will present in details the methodology we applied during this thesis.

3.2 Analysis-transformation-synthesis methodology

3.2.1 Global methodology

The global methodology we applied to answer the questions raised in the previous section is in line with the studies mentioned in Section 1.3.3.1 [Sarroff and Casey 2014; Colonel

[1]The studies using GANs for audio synthesis were published only very recently (2019).

et al. 2017; Blaauw and Bonada 2016; Hsu et al. 2017; Esling et al. 2018b] and follows an analysis-transformation-synthesis process as illustrated in Figure 3.1.

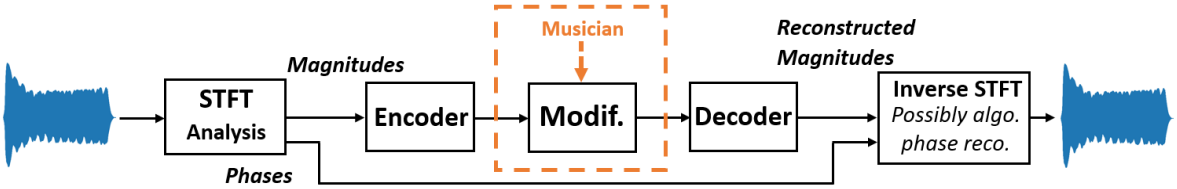


Figure 3.1: Global diagram of the analysis-transformation-synthesis process.

The first step of the process consists in converting the original signal into a low-level representation, usually in the time-frequency domain using a short-term Fourier transform (STFT). The choice of the data representation to feed the model will be detailed further in Section 3.2.3. Then, only the magnitudes of the spectrogram are kept and input to the encoder part of the AE-based model on a frame by frame basis, i.e. each frame is encoded into a latent vector. The input spectrogram is thus encoded into trajectories of latent vectors.

Then, these trajectories can possibly be modified by the musician. For examples this can be interpolating between two latent trajectories encoded from two different samples, cross-fading between two different trajectories or even just enlarging or reducing the trajectories around their mean value, see Figure 3.2. This step of the process is not done during the training of the models where they learn by reconstructing the input at the output without any modification.

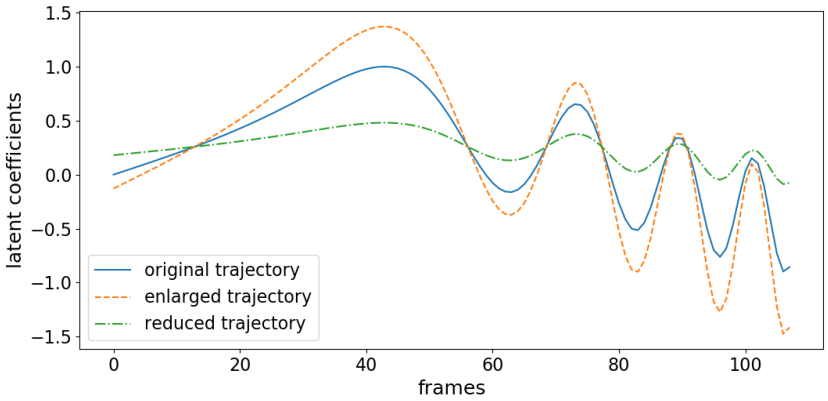


Figure 3.2: Example of modification of one particular latent dimension trajectory by the users.

The final step of the process is thus to decode these trajectories in order to reconstruct a magnitude spectrogram. The output audio signal is then synthesized by combining the decoded magnitude spectrogram with the phase spectrogram and applying the inverse STFT (ISTFT) with overlap-add (OLA). If the latent coefficients are not modified in between the encoding and decoding steps, the decoded magnitude spectrogram is close to the original one and the original phase spectrogram can be directly used for good quality waveform reconstruction. Otherwise, if the latent coefficients are modified so that the decoded magnitude

spectrogram becomes too different from the original, then a well-chosen phase reconstruction algorithm is used before applying ISTFT for reconstructing the time-domain signal. The different algorithms for phase reconstruction from a magnitude spectrogram we investigated will be described later in Section 3.2.3.3.

3.2.2 AE-based models

As explained in Section 1.3.2, AE-based models are machine learning models which functioning is twofold: first the input data are encoded into a lower-dimensional latent vector and then decoded back to the original space. In can thus be decomposed into two sub-models: the encoder and the decoder. For each of these sub-models, there exist plenty of possibilities.

In this section, we will present and detail the several types of AE-based models we investigated in order to extract a high-level low-dimensional space from a dataset of sound samples.

3.2.2.1 Linear AE: PCA

Principal component analysis (PCA) is the optimal linear orthogonal transformation that provides a new coordinate system (i.e. the latent space) in which basis vectors follow modes of greatest variance in the original data [Bishop 2006].

As a baseline for our study, we investigated the use of PCA as a linear encoder to reduce the dimensionality of the input vector $\mathbf{x}$ followed by an inverse PCA to reconstruct it from its encoding (latent) vector $\mathbf{z}$.

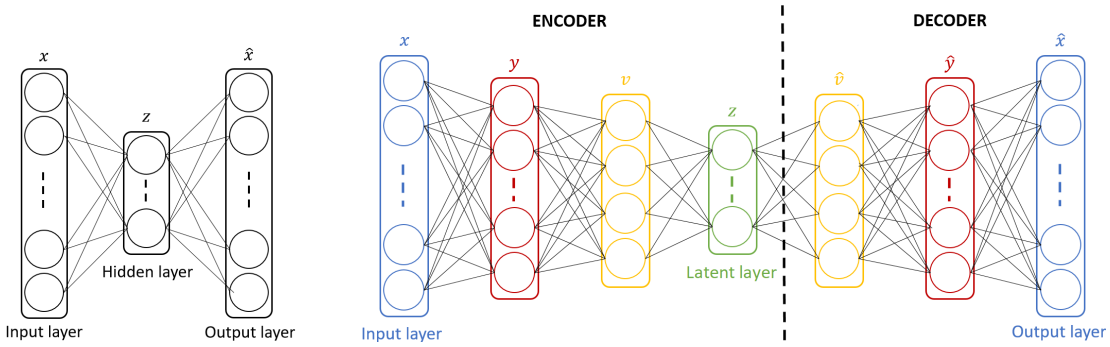
3.2.2.2 AE and deep AE

An autoencoder (AE) is a specific type of ANN that is commonly used for dimensionality reduction tasks thanks to its diabolo shape [Goodfellow et al. 2016], see Figure 3.3. It is composed of an encoder and a decoder. The encoder maps a high-dimensional low-level input vector $\mathbf{x}$ into a higher-level low-dimensional latent vector $\mathbf{z}$ which is assumed to nicely encode properties or attributes of $\mathbf{x}$. Similarly, the decoder reconstructs an estimate $\hat{\mathbf{x}}$ of the input vector $\mathbf{x}$ from the latent vector $\mathbf{z}$. The model can be written as:

$$\begin{cases} \mathbf{z} = f_{\text{enc}}\Big(\mathbf{W}_{\text{enc}}\ \mathbf{x} + \mathbf{b}_{\text{enc}}\Big), \\ \hat{\mathbf{x}} = f_{\text{dec}}\Big(\mathbf{W}_{\text{dec}}\ \mathbf{z} + \mathbf{b}_{\text{dec}}\Big), \end{cases}$$

where f_{enc} and f_{dec} are (entry-wise) non-linear activation functions, $\mathbf{W}_{\text{enc}}$ and $\mathbf{W}_{\text{dec}}$ are weight matrices and $\mathbf{b}_{\text{enc}}$ and $\mathbf{b}_{\text{dec}}$ are bias vectors. For regression tasks (such as the one considered in this study), a linear activation function is generally used for the output layer.

At training time, the weight matrices and the bias vectors are learned by minimizing a cost function over a training dataset using the usual back-propagation technique [Rumelhart et al. 1986]. Here we consider the mean squared error (MSE) between the input $\mathbf{x}$ and the output $\hat{\mathbf{x}}$. AEs are thus a particular type of unsupervised neural models as they are actually self-supervised.



(a) Shallow AE. (b) Deep AE.

Figure 3.3: General architecture of shallow and deep autoencoders.

The model can be extended by adding hidden layers in both the encoder and the decoder and create a so-called deep autoencoder (DAE), as illustrated in Figure 3.3b. This kind of architecture can be trained globally (end-to-end) or layer-by-layer by considering the DAE as a stack of shallow AEs [Hinton and Salakhutdinov 2006; Bengio et al. 2007].

3.2.2.3 Recurrent AE

In a general manner, a recurrent neural network (RNN) is an ANN where the output of a given hidden layer does not depend only on the output of the previous layer (as in feed-forward architectures) but also on the internal state of the network. Such internal state can be defined as the output of each hidden neuron when processing the previous input observations. They are thus well-suited to process time series of data and capture their time dependencies. Such networks are here expected to extract latent representations that encode some aspects of the sound dynamics. Among different existing RNN architectures, in this study we used the Long Short-Term Memory (LSTM) network [Hochreiter and Schmidhuber 1997] which is known to tackle correctly the so-called vanishing gradient problem in RNNs [Bengio et al. 1994]. In order not to loose too much information, we used the LSTM network with a many-to-many architecture for both encoder and decoder, i.e. encoding an input sequence $\mathbf{x}$ into a latent sequence $\mathbf{z}$, see Figure 3.4, and then decoding it back into an output sequence $\hat{\mathbf{x}}$.

The structure of the model depicted in Figure 3.3 still holds while replacing the classic neuronal cells by LSTM cells, leading to a LSTM-AE. The cost function to optimize remains the same, i.e. the MSE between the input $\mathbf{x}$ and the output $\hat{\mathbf{x}}$. However, the model is much more complex and has more parameters to train [Hochreiter and Schmidhuber 1997].

3.2.2.4 Variational AE

Principle of VAE Variational autoencoders (VAEs) have been introduced by [Kingma and Welling 2014] and [Rezende et al. 2014]. As mentioned in Section 1.3.2, a VAE can be seen as a probabilistic AE which delivers a parametric model of the data distribution [Kingma and Welling 2014], such as:

$$p_\theta(\mathbf{x}, \mathbf{z}) = p_\theta(\mathbf{x}|\mathbf{z})p_\theta(\mathbf{z}), \tag{3.1}$$

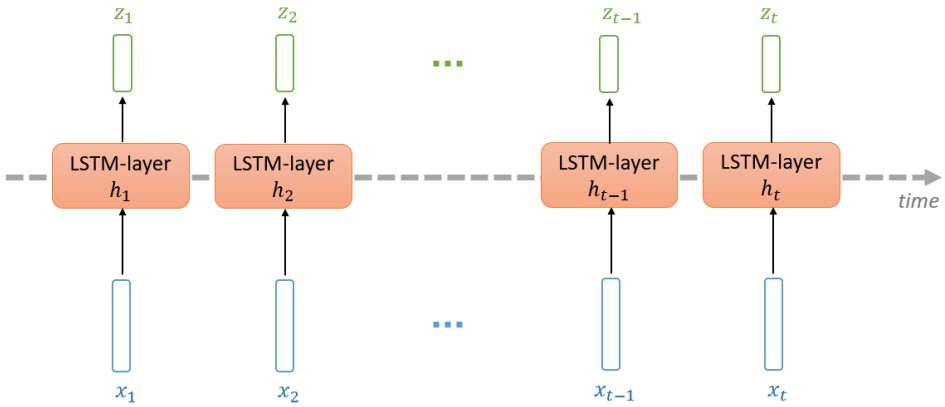


Figure 3.4: Structure of a many-to-many LSTM network for an input sequence $\mathbf{x}$ and a corresponding output sequence $\mathbf{z}$ with the same length, h_t being the internal state of the LSTM layer at time t.

where θ denotes the set of distribution parameters, $\mathbf{x} \in \mathbb{R}^F$ is a vector of observed data and $\mathbf{z} \in \mathbb{R}^L$ is a corresponding vector of latent data, with $L \ll F$. The likelihood function $p_\theta(\mathbf{x}|\mathbf{z})$ plays the role of a probabilistic decoder which models how the generation of observed data $\mathbf{x}$ is conditioned on the latent data $\mathbf{z}$. The prior distribution $p_\theta(\mathbf{z})$ is used to structure (or regularize) the latent space. Typically a standard Gaussian distribution is used: $p_\theta(\mathbf{z}) = \mathcal{N}(\mathbf{z}; \mathbf{0}, \mathbf{I}_L)$, where $\mathbf{I}_L$ is the identity matrix of size L [Kingma and Welling 2014]. This encourages the latent coefficients to be orthogonal and within a similar range. The likelihood function $p_\theta(\mathbf{x}|\mathbf{z})$ is usually defined as a Gaussian density:

$$p_\theta(\mathbf{x}|\mathbf{z}) = \mathcal{N}(\mathbf{x}; \boldsymbol{\mu}_\theta(\mathbf{z}), \boldsymbol{\sigma}^2_\theta(\mathbf{z})), \tag{3.2}$$

where $\mathcal{N}(\mathbf{x}; \boldsymbol{\mu}, \boldsymbol{\sigma}^2)$ denotes the probability density function (PDF) of the multivariate Gaussian distribution which is defined in the Appendix C.1, $\boldsymbol{\mu}_\theta(\mathbf{z}) \in \mathbb{R}^F$ and $\boldsymbol{\sigma}^2_\theta(\mathbf{z}) \in \mathbb{R}^F_+$ are the outputs of the decoder network, see Figure 3.5, and the parameter set θ is composed of its weight matrices and bias vectors (hence $\theta = \{\mathbf{W}_{\text{dec}}, \mathbf{b}_{\text{dec}}\}$). Note that the entries of $\mathbf{x}$ are assumed independent as common in VAEs, so the vector $\boldsymbol{\sigma}^2_\theta(\mathbf{z})$ contains the diagonal coefficients of a diagonal covariance matrix.

The exact posterior distribution $p_\theta(\mathbf{z}|\mathbf{x})$ corresponding to the above model is intractable. It is approximated by $q_\phi(\mathbf{z}|\mathbf{x})$, a tractable parametric model that plays the role of the corresponding probabilistic encoder. This model generally has a form similar to the decoder:

$$q_\phi(\mathbf{z}|\mathbf{x}) = \mathcal{N}(\mathbf{z}; \tilde{\boldsymbol{\mu}}_\phi(\mathbf{x}), \tilde{\boldsymbol{\sigma}}^2_\phi(\mathbf{x})),$$

where $\tilde{\boldsymbol{\mu}}_\phi(\mathbf{x}) \in \mathbb{R}^L$ and $\tilde{\boldsymbol{\sigma}}^2_\phi(\mathbf{x}) \in \mathbb{R}^L_+$ are the outputs of the encoder network, see Figure 3.5. The parameter set ϕ is composed of the weight matrices and bias vectors ($\mathbf{W}_{\text{enc}}$ and $\mathbf{b}_{\text{enc}}$) of this encoder network. As before, $\tilde{\boldsymbol{\sigma}}^2_\phi(\mathbf{x})$ is a vector containing the diagonal entries of a diagonal covariance matrix.

VAE training The training of the VAE model, i.e. the estimation of θ and ϕ, is done by optimizing the lower-bound of the marginal log-likelihood $\log p_\theta(\mathbf{x})$ over a large dataset

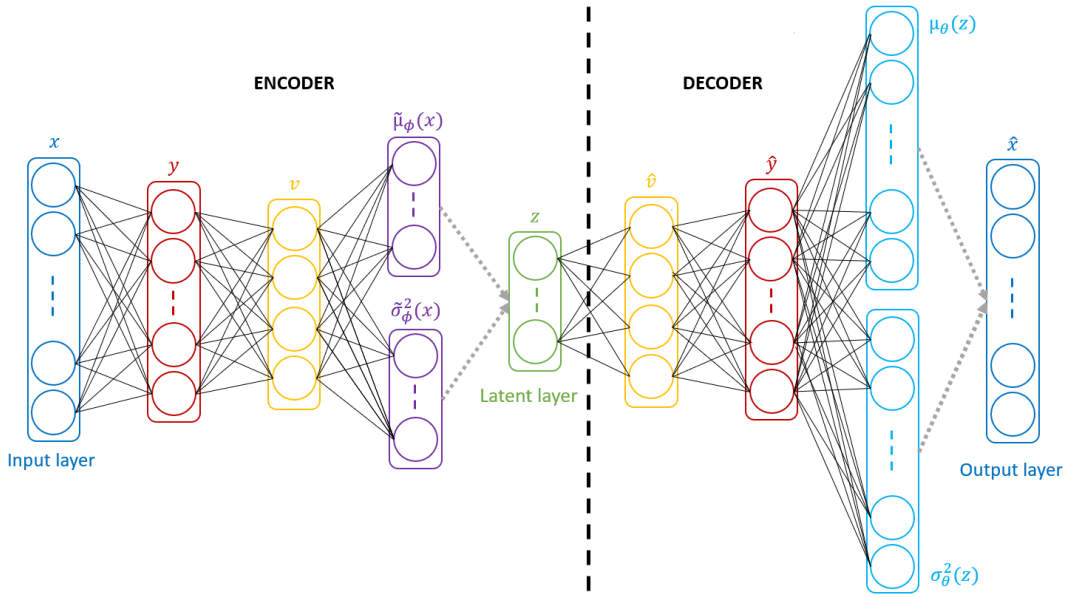


Figure 3.5: General architecture of a VAE. Grey dotted arrows represent sampling process.

of vectors $\mathbf{x}$. It can be shown (see Appendix C.2.1) that the marginal log-likelihood for an individual vector $\mathbf{x}$ can be written as:

$$\log p_\theta(\mathbf{x}) = D_{\mathrm{KL}}\Big(q_\phi(\mathbf{z}|\mathbf{x})|p_\theta(\mathbf{z}|\mathbf{x})\Big) + \mathcal{L}(\phi, \theta, \mathbf{x}), \tag{3.3}$$

where $D_{\mathrm{KL}} \geq 0$ denotes the Kullback-Leibler (KL) divergence and $\mathcal{L}(\phi, \theta, \mathbf{x})$ is the variational lower bound (VLB) given by:

$$\mathcal{L}(\phi, \theta, \mathbf{x}) = \underbrace{\mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}[\log p_\theta(\mathbf{x}|\mathbf{z})]}_{\text{reconstruction accuracy}} - \underbrace{D_{\mathrm{KL}}\Big(q_\phi(\mathbf{z}|\mathbf{x})|p_\theta(\mathbf{z})\Big)}_{\text{regularization}}. \tag{3.4}$$

As the KL divergence is always nonnegative, maximizing the marginal log-likelihood is equivalent to maximizing the VLB. Hence, in practice, the model is trained by maximizing $\mathcal{L}(\phi, \theta, \mathbf{x})$ with respect to the parameters ϕ and θ over a set of training samples, as detailed below. As we can see in (3.4), the lower-bound is composed of two terms: the first represents the average reconstruction accuracy and the second acts as a regularizer, encouraging $q_\phi(\mathbf{z}|\mathbf{x})$ to be close to the prior $p_\theta(\mathbf{z})$. They both need to be differentiable with respect to the parameters above.

For the regularization term, the result is rather straightforward as $q_\phi(\mathbf{z}|\mathbf{x})$ and $p_\theta(\mathbf{z})$ are both Gaussian distributions and the term can be written as:

$$\mathrm{D}_{\mathrm{KL}}\Big(q_\phi(\mathbf{z}|\mathbf{x})|p_\theta(\mathbf{z})\Big) = -\frac{1}{2}\sum_{l=1}^{L}(1 + \log\tilde{\sigma}_{\phi,l}(\mathbf{x}) - \tilde{\mu}^2_{\phi,l}(\mathbf{x}) - \tilde{\sigma}^2_{\phi,l}(\mathbf{x})), \tag{3.5}$$

where L is the dimension of the latent vectors and subscript l denotes the l^{th} entry of a vector, see development in Appendix C.2.2 for more details.

Regarding the reconstruction accuracy term, since the expectation taken with respect to $q_\phi(\mathbf{z}|\mathbf{x})$ is analytically intractable, it is approximated using a Monte Carlo estimate with R samples $\mathbf{z}^{(r)}$ independently and identically drawn from $q_\phi(\mathbf{z}|\mathbf{x})$:

$$\mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}[\log\, p_\theta(\mathbf{x}|\mathbf{z})] \approx \frac{1}{R}\sum_{r=1}^{R} \log p_\theta(\mathbf{x}|\mathbf{z}^{(r)}). \tag{3.6}$$

Then, to make it differentiable, we can apply the reparametrization trick introduced in [Kingma and Welling 2014]: $\mathbf{z}^{(r)} = \tilde{\boldsymbol{\mu}}_\phi(\mathbf{x}) + \tilde{\boldsymbol{\sigma}}_\phi(\mathbf{x}) \odot \boldsymbol{\epsilon}^{(r)}$, where $\odot$ represents element-wise multiplication and $\boldsymbol{\epsilon}^{(r)} \sim \mathcal{N}(0, \mathbf{I})$. The sampling is then turned into a deterministic mapping differentiable with respect to ϕ. Using (3.2), the reconstruction term of (3.4) writes (up to a constant factor):

$$\log\, p_\theta(\mathbf{x}|\mathbf{z}) \stackrel{c}{=} -\sum_{f=0}^{F-1} \Big(\log\, \sigma^2_{\theta,f}(\mathbf{z}) + \frac{(x_f - \mu_{\theta,f}(\mathbf{z}))^2}{2\sigma^2_{\theta,f}(\mathbf{z})}\Big), \tag{3.7}$$

where f denotes the f^{th} entry of a vector and F the dimension of the vectors of observed data.

In practice, a training dataset $\mathbf{X} = \{\mathbf{x}_n\}_{n=1}^{N_{tr}}$ is used for the training of the VAE. Under the hypothesis of independent and identically distributed (i.i.d.) training vectors, the VAE training is done by maximizing the total VLB, which is the sum of individual VLBs over the training vectors. If we consider only one Monte Carlo sample per training vector (which is common practice provided that the batch size is sufficiently large [Kingma and Welling 2014]), or if we consider several Monte Carlo samples as additional training data, then the summation over the $\mathbf{z}^{(r)}$ can be omitted (as $R = 1$) and we can write the total VLB as:

$$\mathcal{L}(\phi, \theta, \mathbf{X}) = \sum_{n=1}^{N_{tr}} \log p_\theta(\mathbf{x}_n|\mathbf{z}_n) - \sum_{n=1}^{N_{tr}} D_{\text{KL}}\Big(q_\phi(\mathbf{z}_n|\mathbf{x}_n)|p_\theta(\mathbf{z}_n)\Big) \tag{3.8}$$

where $\mathbf{z}_n$ actually represents $\mathbf{z}_n^{(r)}$ with $r = 1$.

Thus, by using (3.5) and (3.7), the VLB in (3.8) becomes:

$$\begin{aligned}\mathcal{L}(\phi, \theta, \mathbf{X}) = &-\sum_{n=1}^{N_{tr}} \sum_{f=0}^{F-1} \left(\log\, \sigma^2_{\theta,f}(\mathbf{z}_n) + \frac{\big(x_{fn} - \mu_{\theta,f}(\mathbf{z}_n)\big)^2}{2\sigma^2_{\theta,f}(\mathbf{z}_n)}\right) \\ &+ \frac{1}{2}\sum_{n=1}^{N_{tr}} \sum_{l=1}^{L} \Big(\log\, \tilde{\sigma}^2_{\phi,l}(\mathbf{x}_n) - \tilde{\mu}^2_{\phi,l}(\mathbf{x}_n) - \tilde{\sigma}^2_{\phi,l}(\mathbf{x}_n)\Big).\end{aligned} \tag{3.9}$$

The maximization of this VLB is done, as for (D)AE and LSTM-AE, by using the back-propagation method [Rumelhart et al. 1986].

In addition, as discussed in [Blaauw and Bonada 2016; Higgins et al. 2017], for some applications it is necessary to find an appropriate balance between the reconstruction term and the regularization term in order to get a nice tradeoff between latent space organization and signal quality. Indeed, if the regularization term is too important compared to the reconstruction accuracy term, then the signal will be very poorly reconstructed due to the

fact that the model can get stuck in a state where most latent coefficients are saturated or inactive. Conversely, a weak regularization will not force the latent space to be close to the prior $p_\theta(\mathbf{z})$ and somehow turn the VAE into an AE. The tradeoff was realized using a weighted version of (3.8):

$$\begin{aligned}\mathcal{L}(\phi,\theta,\beta,\mathbf{X}) &= \sum_{n=1}^{N_{tr}} \log p_\theta(\mathbf{x}_n|\mathbf{z}_n) - \beta \sum_{n=1}^{N_{tr}} D_{\mathrm{KL}}\Big(q_\phi(\mathbf{z}_n|\mathbf{x}_n)|p_\theta(\mathbf{z}_n)\Big) \\ &= -\sum_{n=1}^{N_{tr}}\sum_{f=0}^{F-1}\left(\log\,\sigma^2_{\theta,f}(\mathbf{z}_n) + \frac{\big(x_{fn}-\mu_{\theta,f}(\mathbf{z}_n)\big)^2}{2\sigma^2_{\theta,f}(\mathbf{z}_n)}\right) \\ &\quad + \frac{\beta}{2}\sum_{n=1}^{N_{tr}}\sum_{l=1}^{L}\Big(\log\,\tilde{\sigma}^2_{\phi,l}(\mathbf{x}_n) - \tilde{\mu}^2_{\phi,l}(\mathbf{x}_n) - \tilde{\sigma}^2_{\phi,l}(\mathbf{x}_n)\Big),\end{aligned} \tag{3.10}$$

where β is a weighting coefficient set empirically to maintain both the reconstruction accuracy and the KL divergence terms in the same range and tunable according to the targeted tradeoff between reconstruction and regularization.

3.2.3 Data representation

Most of the papers introduced in Section 1.3.3.1 dealing with AE-based deep neural models for audio synthesis use a time-frequency domain (magnitudes only) representation of the data to feed the models on a frame by frame basis. To the best of our knowledge, one paper investigated the use of raw waveform representation as input [Engel et al. 2017]. Although this low-level representation is very convenient as it corresponds to the desired final output and allows to get rid of the phase reconstruction issues that come with the magnitude-STFT representation, it involves extensive computations due to the use of WaveNet model [van den Oord et al. 2016b] for synthesizing a few seconds long signal, which takes too much time for our application (more than 15 minutes for generating a signal of a duration of 4 seconds according to [Engel et al. 2019])[2].

In this section, we will present the different data representations in the time-frequency domain that have been used in the different previously cited studies and can possibly be used, in particular for maintaining a consistent statistical framework with VAE (see Section 3.2.3.2), and the investigated phase reconstruction algorithms that are necessary to reconstruct time-domain signals from spectrograms.

3.2.3.1 Representation in the time-frequency domain

As already stated before, most of the studies using AE-based models for synthesizing audio are dealing with a representation of the data in the time-frequency domain, discarding the phase information when feeding the models. Several different possibilities of representations exist in time-frequency. For example, in [Sarroff and Casey 2014], [Colonel et al. 2017] and

[2]However, we did try to apply AE-based synthesis on raw waveforms by investigating the generation of single-cycle waves (i.e. very short signal of exactly one period duration) allowing us to use regular (V)AE-based models for a possibly interesting new oscillator. As the results were not very promising, we dropped the raw waveform representation and followed all the other studies by using a STFT-domain representation of the data.

[Esling et al. 2018b], the authors applied the model to the linear magnitude coefficients of the STFT whereas in [Hsu et al. 2017] they used the log-magnitude coefficients. In [Hsu et al. 2017], they also investigated mel-scale time-frequency representation of the spectrogram, where frequency bins are computed using filter banks that match human perceptual sensitivity of frequencies. Other representations such as non-stationary Gabor transform (NSGT) [Balazs et al. 2011] or the discrete cosine transform (DCT) [Diniz et al. 2010] were also explored in [Esling et al. 2018b] as alternative to the STFT.

In the context of this study, we chose to focus on a magnitude STFT representation of the data, either the log-magnitude coefficients or the power spectrogram (i.e. squared magnitude) depending on the model and the loss function chosen for the VAE model. In all these representations of the STFT, the information about the signal are the same and it is easy to switch representations by applying a square transformation, a log-transformation or its inverse.

3.2.3.2 Statistical modeling and implications for VAE training

In the VAE framework, the minimization of the VLB with respect to the reconstruction accuracy term amounts to the optimal estimation of the model parameters (actually the decoder parameters) in the maximum-likelihood (ML) sense which depends on the form of the underlying statistical model. The precise choice of the encoder and decoder conditional distributions, i.e. the assumption made on the underlying statistical model of the input data, thus have an impact on the loss function to minimize when training the model and have to be wisely chosen.

In most papers dealing with VAE-based spectrogram modeling [Blaauw and Bonada 2016; Hsu et al. 2017; Akuzawa et al. 2018; Esling et al. 2018b], MSE is used as the reconstruction error. However, this choice has important implications on the underlying statistical model of the signal.

Indeed, taking the MSE between the input vector $\mathbf{x}$ and the output $\hat{\mathbf{x}}$ is equivalent to setting the reconstruction accuracy of (3.7) to:

$$\log p_\theta(\mathbf{x}|\mathbf{z}) \stackrel{c}{=} -\sum_{f=0}^{F-1} \Big(x_{fn} - \mu_{\theta,f}(\mathbf{z}_n)\Big)^2, \tag{3.11}$$

corresponding thus to maximizing the likelihood function under a "fixed-variance free-mean" Gaussian model hypothesis. Even if this assumption provides some nice theoretical interpretation of the process, this is poorly discussed in the papers, and raises many questions about the limitations and the relevance of this interpretation with the chosen data representation.

Yet, a consistent theoretical framework enabling to justify and interpret the choice of data representation, likelihood function and reconstruction term of the loss function, plus how these points are related does exist. It has been evidenced for spectrogram modeling with nonnegative matrix factorization (NMF) in the seminal papers [Févotte et al. 2009; Févotte and Cemgil 2009]. In [Girin et al. 2019][3] , we presented how this framework can be applied to VAE. We recall here the main results of this study, which present the three major cases of

[3]The full paper is available in Appendix F.

loss functions associated with different underlying distribution of the STFT.

Euclidian distance case: the first assumption that is usually done and that we already developed in (3.11) is the fixed-variance free-mean Gaussian assumption on the spectrogram magnitude coefficients x_{fn} such that $x_{fn} \sim \mathcal{N}(x_{fn}; \mu_{\theta,f}(\mathbf{z}_n), \sigma^2)$, where $\sigma^2_{\theta,f}(\mathbf{z}_n) = \sigma^2, \forall(f,n)$. Maximizing the parameters of the model in the ML sense then leads to minimizing the squared Euclidian distance between the input data x_{fn} and the output parameter $\mu_{\theta,f}(\mathbf{z}_n)$, i.e. using the MSE as reconstruction accuracy loss function. Here x_{fn} can represent indifferently (linear or log) magnitude or power spectrum.

Itakura-Saito divergence case: the second classically used framework is to make an assumption of a circularly symmetric complex Gaussian model for the STFT complex coefficient $s_{fn} \in \mathbb{C}$ (i.e. $s_{fn} \sim \mathcal{N}_c(s_{fn}; 0, \sigma^2_{\theta,f}(\mathbf{z}_n))$, see PDF of a complex Gaussian distribution $\mathcal{N}_c$ in Appendix C.1.1). This assumption implies a Gamma distribution for the power spectrum coefficients $|s_{fn}|^2$. We thus have $x_{fn} \sim \mathcal{G}(x_{fn}; \alpha, \alpha/\sigma^2_{\theta,f}(\mathbf{z}_n))$, see probability density function (PDF) in Appendix C.1.2, where $\mathbb{E}[x_{fn}] = \mathbb{E}[|s_{fn}|^2] = \sigma^2_{\theta,f}(\mathbf{z}_n)$ with $x_{fn} \in \mathbb{R}_+$. In this case, maximizing the model parameters in the ML sense corresponds to minimizing the Itakura-Saito (IS) divergence between the input data x_{fn} and the output model parameter $\sigma^2_{\theta,f}(\mathbf{z}_n)$:

$$D_{\text{IS}}\left(x_{fn}|\sigma^2_{\theta,f}(\mathbf{z}_n)\right) = \frac{x_{fn}}{\sigma^2_{\theta,f}(\mathbf{z}_n)} - \log \frac{x_{fn}}{\sigma^2_{\theta,f}(\mathbf{z}_n)} - 1.$$

Kullback-Leibler divergence case: the final usual case is the assumption of a Poisson distribution of $x_{f,n}$: $x_{fn} \sim \mathcal{P}(x_{fn}; \sigma_{\theta,f}(\mathbf{z}_n))$, see PDF in Appendix C.1.3. Now, maximizing the model parameters in the ML sense is equivalent to minimizing the KL divergence between x_{fn} and $\sigma_{\theta,f}(\mathbf{z}_n)$:

$$D_{\text{KL}}\left(x_{fn}|\sigma_{\theta,f}(\mathbf{z}_n)\right) = x_{fn} \log \frac{x_{fn}}{\sigma_{\theta,f}(\mathbf{z}_n)} - x_{fn} - \sigma_{\theta,f}(\mathbf{z}_n).$$

As there is no direct underlying statistical model on complex valued STFT coefficients, any representation can be used for the x_{fn}. However, historically, KL-based NMF has always been applied on linear-scale magnitude spectra [Smaragdis and Brown 2003; Virtanen 2007].

In summary, we have thus, depending on the loss function:

- Euclidian distance: $p_\theta(\mathbf{X}|\mathbf{Z}) = \prod_{f,n} \mathcal{N}(x_{fn}; \mu_{\theta,f}(\mathbf{z}_n), \sigma^2)$;
- IS divergence: $p_\theta(\mathbf{X}|\mathbf{Z}) = \prod_{f,n} \mathcal{G}(x_{fn}; \alpha, \alpha/\sigma^2_{\theta,f}(\mathbf{z}_n))$ and $p_\theta(\mathbf{S}|\mathbf{Z}) = \prod_{f,n} \mathcal{N}_c(s_{fn}; 0, \sigma^2_{\theta,f}(\mathbf{z}_n))$ with $x_{fn} = |s_{fn}|^2$;
- KL divergence: $p_\theta(\mathbf{X}|\mathbf{Z}) = \prod_{f,n} \mathcal{P}(x_{fn}; \sigma_{\theta,f}(\mathbf{z}_n))$.

The usual assumption that is made for audio (in particular for source separation and speech enhancement [Duong et al. 2010; Vincent et al. 2011; Liutkus et al. 2011; Bando et al. 2018; Leglaive et al. 2018; Leglaive et al. 2019a]) is the circularly symmetric complex Gaussian model for the STFT complex coefficients, which makes the IS divergence loss function more

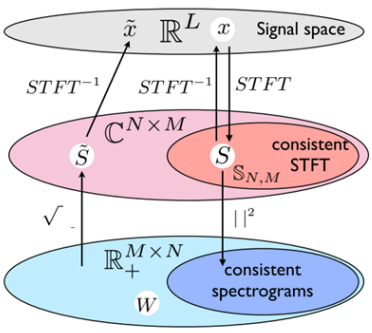
adapted for our application as we work with power spectrogram. However, historically, the studies dealing with VAE for audio did rely on the Gaussian assumption for x_{fn} thus implying the minimization of the squared Euclidian distance between the input data x_{fn} and the output parameter when optimizing the parameters of the model in the ML sense. In the first experiments we realized we therefore used this assumption and we kept the same setting for all the experiments that will be presented in this manuscript. For future work however, it would be interesting to head towards the use of the IS divergence within the VAE lower bound.

3.2.3.3 Phase spectrogram reconstruction

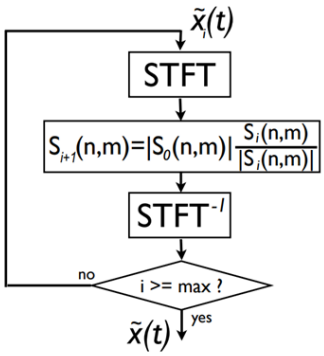
In order to reconstruct a time-domain signal from STFT magnitude (or power or log-magnitude) spectrogram, the phase information is necessary. Thus, if the phase spectrogram corresponding to a magnitude spectrogram is not available, it has to be reconstructed from the magnitude coefficients. This is generally the case when the STFT magnitude spectrogram is modified by the model (e.g. by changing the latent coefficients values in an AE). There exist several methods to do so.

In this section, we will present and detail the methods we applied during this project.

Griffin and Lim method One of the most used method for reconstructing the phase spectrogram from the magnitude spectrogram is the Griffin and Lim algorithm (G&L) [Griffin and Lim 1984]. It is an iterative algorithm based on a consistency approach. The basic idea of this approach is that any complex matrix does not necessarily correspond to the STFT of a signal (see Figure 3.6a). The purpose here is then to iteratively compute successive STFT and ISTFT starting from the given magnitude spectrogram in order to converge towards a complex matrix that is consistent with the spectrogram of a signal (i.e. the most plausible one given the magnitude spectrogram), see Figure 3.6b.



(a) Domains involved when processing STFT and magnitude spectrograms.



(b) The iterative framework of Griffin and Lim algorithm.

Figure 3.6: Illustration of the concepts behind the Griffin and Lim algorithm: consistency and iterative framework. These two figures are taken from [Sturmel and Daudet 2011].

One of the main issues with this algorithm is that it does not perform well if the given magnitude spectrogram is not a genuine magnitude STFT but, for instance, a spectrogram that is reconstructed by a neural network and was potentially subjected to modifications as in the context of this thesis. The algorithm can also converge towards unsatisfactory solutions, where there are some phase signs inversions that sound unnatural for example. Another drawback is that this algorithm relies on the redundancy of information in the spectrogram and iterates each time on the whole spectrogram using STFT and ISTFT, which makes it rather complicated to use for real-time applications. Some real-time implementations of the G&L algorithm have been recently proposed though, but they are still quite complex in terms of computations.

Linear phase unwrapping method Another approach for reconstructing the phase of a spectrogram we investigated is the so-called linear phase unwrapping. This method based on phase trajectory reconstruction dates back to the seventies. Its general principle is to work in time-frequency domain and to interpolate both the magnitudes and the phases between frames. There are two historical schools of thoughts that converged on this principle. The first is the phase vocoder [Allen 1977; Portnoff 1980; Dolson 1986] where the interpolation is applied for all the frequency bins. The other method is called the sinusoidal model [McAulay and Quatieri 1986; McAulay and Quatieri 1995; George and Smith 1997] and consists in representing the signal as a mixture of sine waves. Using this representation, the interpolations in amplitude and phase are no longer performed on every frequency bin but only on the peaks (corresponding to the harmonics of the signal or the component sine waves) that are first detected for each frame. The sinusoidal model has then been extended to a sinusoidal plus noise model to take into account the aperiodic components of natural sounds, where the noise is obtained by subtracting the analyzed sinusoids from the signal [Serra and Smith 1990; Serra 1997]. The obtained residual noise can then be modeled separately with appropriate noise models, possibly transformed, and then added back to the (modified and resynthesized) sinusoidal part.

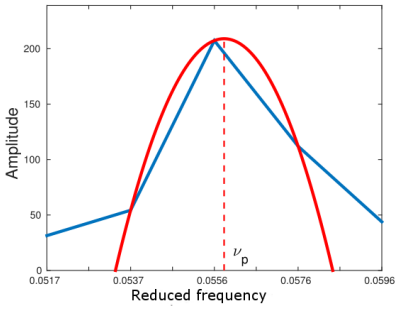
This method has been recently updated by [Magron 2016] through a slightly different framework. Similarly as for the sinusoidal model, his method consists first in modeling the signal as a mixture of sinusoids and exploits the natural relationship between adjacent frequency bins. Then, the sinusoidal components are selected by detecting the peaks for each STFT frame of the signal (with quadratic interpolation) and each frequency bin is associated to a particular peak. Finally, the phases are computed by interpolation of every bin using these peaks dependencies.

Technically, for each frame, the peaks are first detected and then the instantaneous frequency of each one of them is computed using quadratic interpolation, see Figure 3.7a. Finally, depending on its amplitude, each peak will be associated with an influence zone defined by the following equation, see Figure 3.7b.

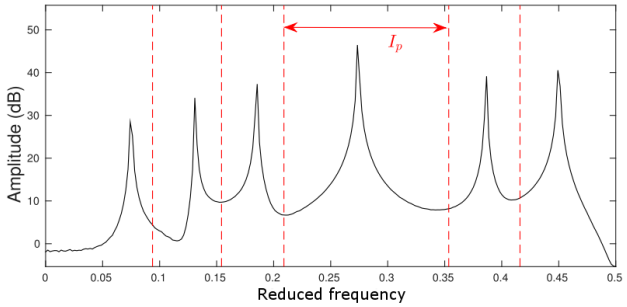
$$I(p) = \left\lfloor \frac{A_p f^*_{p-1} \ + \ A_{p-1} f^*_p}{A_p \ + \ A_{p-1}} \right\rfloor, \tag{3.12}$$

where p is the peak index, f^*_p its instantaneous frequency and A_p its respective amplitude.

For each frequency bin, the phase can then be linearly unwrapped with respect to time



(a) Quadratic interpolation for instantaneous frequency estimation of the peaks.



(b) Peaks influence zones repartition.

Figure 3.7: Phase unwrapping processing of spectral peaks. Both figures are taken from [Magron 2016].

using the sinusoidal model with the frequency of the peak associated to the influence zone it belongs to [Magron 2016] using the following formula:

$$\Phi(f,t) = \Phi(f,t-1) \; + \; 2\pi f_p^* \; H \quad \forall f \in I(p),$$

where f is the frequency bin and $I(p)$ the current influence zone associated with the p^{th} peak of instantaneous frequency f_p^*, t is the index of the frame and H is the hop-size between consecutive frames.

The time dependency of this method is reduced to only one frame instead of using the whole STFT as for the basic version of the G&L algorithm. This is thus usable for real-time applications, and the signal can be generated on-the-fly using overlap-add (OLA) synthesis.

An important remark that can be done is that this method can be combined with the G&L algorithm. Indeed, the linear phase unwrapping provides a well-adapted initialization for the G&L algorithm which is more general as there is no distinction between the prevailing sinusoidal components and the (weaker or "noisier") other components.

3.3 Comparative study of different AE-based models on two datasets

In order to compare the different AE-based models presented earlier (see Section 3.2.2), we realized several experiments aiming at evidencing their strengths and weaknesses with respect to our application. First we thus compared them in terms of reconstruction accuracy depending on the encoding dimension to see how well they can compress the information without losing too much quality in the synthesized/decoded signals. Then we compared the extracted latent spaces in terms of correlation of the dimensions to see if some models are more adapted to extract a control space than others. Finally, as a first step towards the use of these latent spaces for navigating through the sound space and creating new sounds, we investigated how they can be used to interpolate between sounds in the spirit of what was done for instrument hybridization in [Engel et al. 2017], and compared them qualitatively.

In the following, we will present and detail these experiments we performed on two different datasets[4] that will be first introduced.

3.3.1 Datasets

3.3.1.1 NSynth dataset

First, in order to have a benchmark on standardized data, we used the NSynth dataset introduced in [Engel et al. 2017]. This is a large publicly available database (more than 30 GB) of 4s long monophonic music sounds sampled at 16 kHz. They represent 1,006 different instruments generating notes with different pitches (from MIDI 21 to 108) and different velocities (5 different levels from 25 to 127). To generate these samples, different methods were used: some acoustic and electronic instruments were recorded and some others were synthesized. The dataset is labeled with:

- instrument family (e.g., keyboard, guitar, synth_lead, reed),
- source (acoustic, electronic or synthetic),
- instrument index within the instrument family,
- pitch value, and
- velocity value.

Some other labels qualitatively describe the samples, e.g. brightness or distortion, but they were not used in our work.

To train our models, we used a subset of 10,000 different sounds randomly chosen from this NSynth database[5], representing all families of instruments, different pitches and different velocities. We split this dataset into a training set (80%) and testing set (20%). During the training phase, 20% of the training set was kept for validation. In order to have a statistically robust evaluation, a k-fold cross-validation procedure with $k = 5$ was used to train and test all different models (we divided the dataset into 5 folds, used 4 of them for training and the remaining one for test, and repeated this procedure 5 times so that each sound of the initial dataset was used once for testing).

3.3.1.2 Arturia dataset

Then, we focused on a dataset containing sound samples representing the target sounds of our application. Actually, this dataset has already been introduced in Section 2.2.2.1 and has been referred to as "Arturia dataset".

As a reminder, this dataset is constituted of 1,233 sound samples of a duration of between 2 and 2.5 seconds generated from every preset of the Arturia software synthesizers sampled at 44.1 kHz and normalized in pitch and in loudness, see Section 2.2.2.1.

Similarly to NSynth, the database was split into a training set (80%) and a testing set (20%). Plus, during the training phase, 20% of the training set was kept for validation and,

[4]Parts of the results introduced in this section (the one obtained with the NSynth dataset) were presented at the 2019 Sound & Music Computing (SMC) conference [Roche et al. 2019], see full published paper in Appendix F.

[5]This is this subset of 10,000 samples that will be referred to as "NSynth dataset" for the rest of this manuscript.

in order to have a statistically robust evaluation, we used a k-fold cross-validation procedure with $k = 5$ for all the investigated models.

3.3.2 Data pre-processing

As already introduced in Section 3.2.1, the first step for data pre-processing is magnitude and phase short-term spectra extraction. To do so, for the NSynth dataset, we applied a 1024-point STFT to the input signals using a sliding Hamming window with 50% overlap. Frames corresponding to silence segments were removed. Our dataset of 10,000 samples thus resulted in a dataset containing 1,157,310 (513-point frame) vectors.

Regarding the Arturia dataset, we applied the exact same process except that, considering the sample rate and the fundamental frequency of the data, the magnitude and phase short-term spectra were extracted by applying a 2048-point STFT instead of 1024. This resulted in 133,164 (1025-point frame) vectors.

Then, for both dataset, the corresponding (513 or 1025-point) positive-frequency magnitude spectra were converted to log-scale and normalized in energy: we fixed the maximum of each log-spectrum input vector to 0 dB (the energy coefficient was stored to be used for signal reconstruction). Then, the log-spectra were thresholded, i.e. every log-magnitude below a fixed threshold was set to the threshold value. Finally they were normalized between -1 and 1, which is a usual procedure for ANN inputs. Three threshold values were tested: -80 dB, -90 dB and -100 dB. Corresponding denormalization, log-to-linear conversion and energy equalization were applied after the decoder, before signal reconstruction with transmitted phases and ISTFT with overlap-add.

3.3.3 AE-based models implementations

We tried different types of AE-based models as introduced earlier: AE, DAE, LSTM-AE and VAE. For all the models we investigated several values for the encoding dimension, i.e. the size of the bottleneck layer / latent variable vector, from $enc = 4$ to 100 (with a fine-grained sampling for $enc \leq 16$).

For both datasets, we investigated several architectures the DAEs and VAEs. However, considering the differences in the size of their respective input vectors, the architectures we explored for each dataset varied slightly. For NSynth we tested the following architectures of DAEs:

- $[513, 128, enc, 128, 513]$,
- $[513, 256, enc, 256, 513]$,
- $[513, 256, 128, enc, 128, 256, 513]$;

and for the VAEs, we considered only one architecture: $[513, 128, enc, 128, 513]$.

For the Arturia dataset, we tried the same architectures for both DAEs and VAEs:

- $[1025, 128, enc, 128, 1025]$ and
- $[1025, 512, 128, enc, 128, 512, 1025]$.

Several different values of the weight factor β were tested for the VAEs for both datasets.

Finally, concerning the LSTM-AE, our implementation used two 1-layer feed-forward LSTM layers (one for the encoder and one for the decoder) with non-linear activation functions giving the following architecture: $[F, enc, F]$ (where $F = 513$ for NSynth and 1025 for the Arturia dataset). Both LSTM layers were designed for many-to-many sequence learning, meaning that a sequence of inputs, i.e. of spectral magnitude vectors, is encoded into a sequence of latent vectors of same temporal size and then decoded back to a sequence of reconstructed spectral magnitude vectors as explained in Section 3.2.2.3 and illustrated in Figure 3.4.

For all the neural models, we tested different pairs of activation functions for the hidden layers and output layer, respectively:

- (tanh, linear),
- (sigmoid, linear),
- (relu, linear),
- (tanh, sigmoid).

AE, DAE, LSTM-AE and VAE models were implemented in Python using the *Keras* toolkit [Chollet 2015] (we used the *scikit-learn* [Pedregosa et al. 2011] toolkit for the PCA). Training was performed using the Adam optimizer [Kingma and Ba 2015] with a learning rate of 10^{-3} over 600 epochs with early stopping criterion (with a patience of 30 epochs) and a batch size of 512. The DAEs were trained in two different ways, with and without layer-wise training (see Section 3.2.2.2).

3.3.4 Experimental results

The results of all the experiments presented earlier on the two datasets we just introduced will be presented in the next sections.

3.3.4.1 Analysis-synthesis

In this section, for the sake of clarity, we only present the results obtained for: a threshold of -100 dB applied on the log-spectra and a restricted set of the tested AE-based models (listed in the legends of the figures). The results obtained for other threshold values and other tested architectures did not significantly differ.

As already explained in the introduction, the first step towards comparing the different models is to evaluate their accuracy on an analysis-synthesis task depending on the encoding dimension, and compare them. For that purpose we evaluated the models on both datasets using two different measures. First we measured the RMSE which provides a global measure of magnitude spectra reconstruction but can be insufficiently correlated to perception depending on which spectral components are correctly or poorly reconstructed. To address this issue in audio processing, we thus also calculated objective measures of the perceptual audio quality, namely PEMO-Q scores [Huber and Kollmeier 2006].

In addition, for all the figures of this section, the reconstruction error (respectively PEMO-Q score) for each considered dimension of the latent space and each model are displayed with a 95% confidence interval obtained by conducting paired t-test considering every sound (i.e. every audio file) of the test set as an independent sample.

For more convenience, in this section, the results obtained on the two different dataset will be presented separately.

NSynth dataset Figure 3.8 and Figure 3.9 show respectively the reconstruction error (RMSE in dB) and the PEMO-Q scores obtained with PCA, AE, DAE, LSTM-AE and VAEs model (for different β values) on the NSynth test set (averaged over the 5 folds of the cross-validation procedure) as a function of the dimension of the latent space.

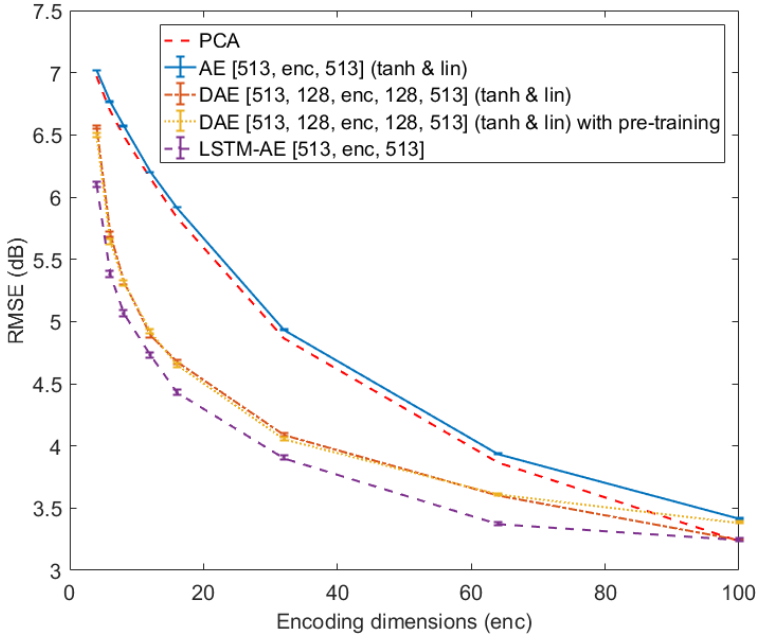
As expected, the RMSE decreases with the increase of the dimension of the latent space for all methods. Interestingly, PCA systematically outperforms (or at worst equals) shallow AE, see Figure 3.8a. This somehow contradicts studies on image compression for which a better reconstruction is obtained with AE compared to PCA [Hinton and Salakhutdinov 2006]. To confirm this unexpected result, we replicated our PCA vs. AE experiment on the MNIST image dataset [LeCun et al. 1998], using the same AE implementation and a standard image pre-processing (i.e. vectorization of each 28×28 pixels gray-scale image into a 784-dimensional feature vector). In accordance with the literature, the best performance was systematically obtained with AE (for any considered dimension of the latent space). This difference of AE's behavior when considering audio and image data was unexpected and, to our knowledge, it has never been reported in the literature.

Then, contrary to (shallow) AE, DAEs systematically outperform PCA (and thus AE), with up to almost 20% improvement (for $enc = 12$ and $enc = 16$). Our experiments did not reveal notable benefit of layer-by-layer DAE training over end-to-end training. Importantly, for small dimensions of the latent space (e.g. smaller than 16), RMSE obtained with DAE decreases much faster than with PCA and AE. This is even more the case for LSTM-AE which shows an improvement of the reconstruction error of more than 23% over PCA (for $enc = 12$ and $enc = 16$). These results confirm the benefits of using a more complex architecture than shallow AE, here deep or recurrent, to efficiently extract high-level abstractions and compress the audio space. This is of great interest for sound synthesis for which the latent space has to be kept as low-dimensional as possible (while maintaining a good reconstruction accuracy) in order to be "controlled" by a musician.

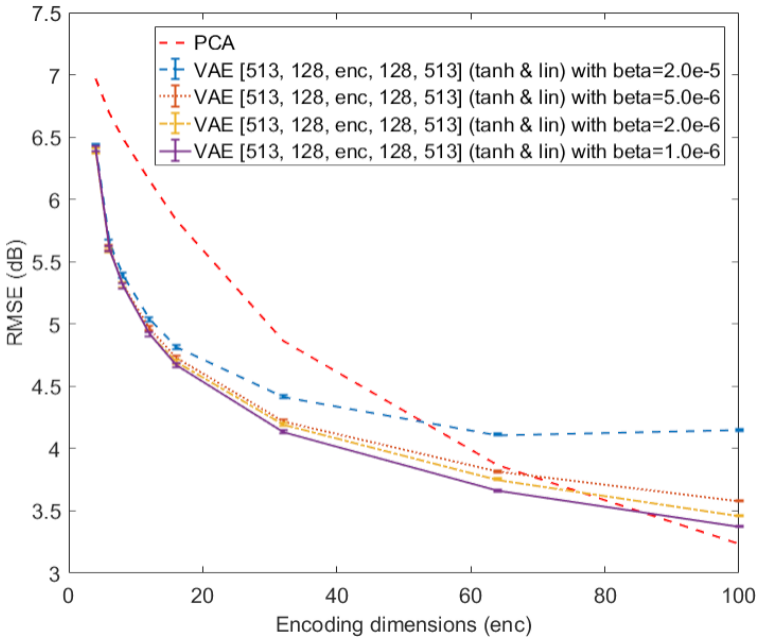
In Figure 3.8b it can be seen that the overall performance of VAEs is in between the performance of DAEs (even equals DAEs for lower encoding dimensions, i.e. lower than 12) and the performances of PCA and AE. Let us recall that minimizing the reconstruction accuracy is not the only goal of VAE which also aims at constraining the distribution of the latent space. As shown in Figure 3.8b, the parameter β, which balances regularization and reconstruction accuracy in equation (3.10), see Section 3.2.2.4, plays a major role. As expected, high β values foster regularization at the expense of reconstruction accuracy. However, with $\beta \leqslant 2.10^{-6}$ the VAE clearly outperforms PCA, e.g. up to 20% for $enc = 12$.

It can also be noticed that when the encoding dimension is high ($enc = 100$), PCA seems to outperform all the other models. Hence, in that case, the simpler (linear model) seems to be the best. We can conjecture that achieving the same level of performance with AEs would require more training data, since the number of free parameters of these model increases drastically. However, using such high-dimensional latent space as control parameters of a music sound generator is impractical.

Similar conclusions can be drawn in terms of audio quality, see Figure 3.9. Indeed, in a



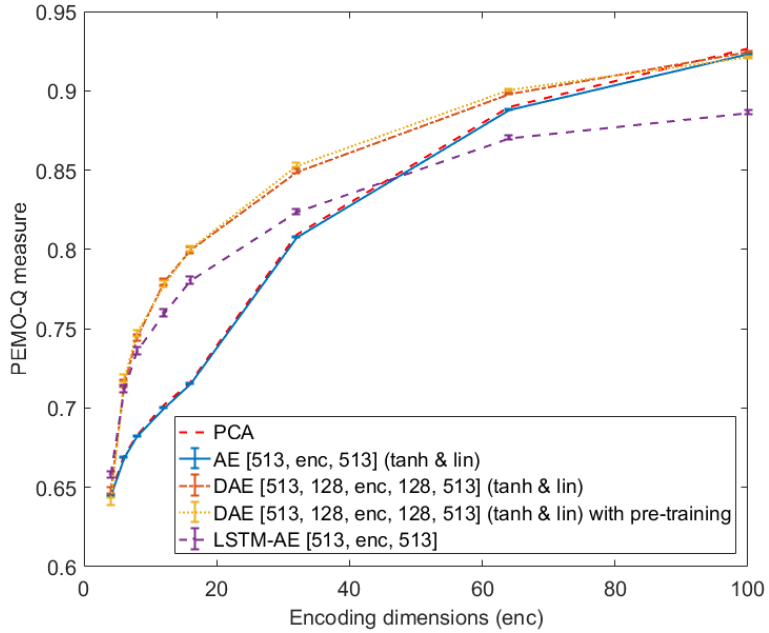
(a) PCA, AE, DAE (with and without layer-wise training) and LSTM-AE.



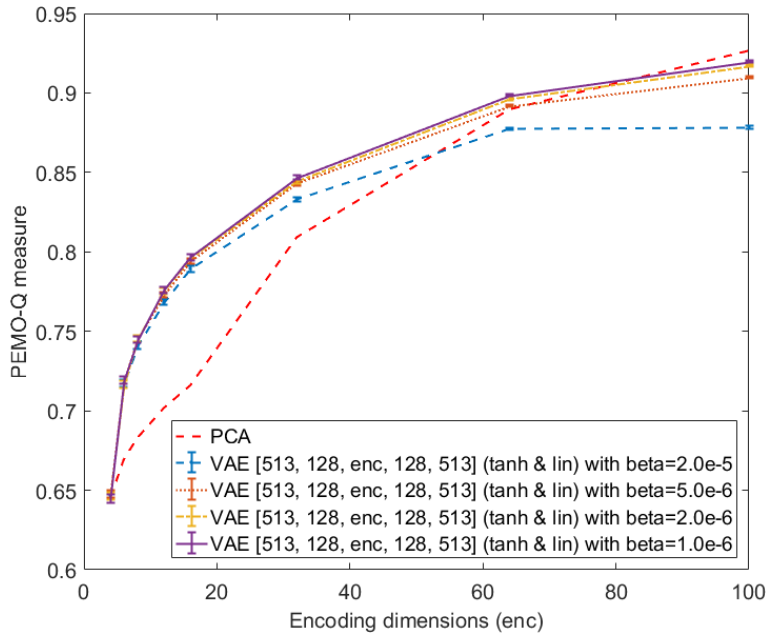
(b) VAE for different values of β (RMSE obtained with PCA is also recalled).

Figure 3.8: Reconstruction error (RMSE in dB) obtained with different AE-based models as a function of latent space dimension (trained on the NSynth dataset).

general manner, the PEMO-Q scores are well correlated with RMSE measures in our experiments. PEMO-Q measures for PCA and AE are very close, but PCA still slightly outperforms



(a) PCA, AE, DAE (with and without layer-wise training) and LSTM-AE.



(b) VAE for different values of β (measures obtained with PCA are also recalled).

Figure 3.9: PEMO-Q measures obtained with different AE-based models as a function of latent space dimension (trained on the NSynth dataset).

the shallow AE. The DAEs and the VAEs both outperform the PCA (up to about 11% for $enc = 12$ and $enc = 16$) with the audio quality provided by the DAEs being a little bet-

ter than for the VAEs. Surprisingly, and contrary to RMSE scores, the LSTM-AE led to a (slightly) lower PEMO-Q scores for all considered latent dimensions. Further investigations will be done to assess the relevance of such differences at the perceptual level.

Audio examples of samples reconstructed by the different AE-based models can be found in the companion webpage.

Arturia dataset Figure 3.10 and Figure 3.11 respectively illustrate the results obtained on the Arturia dataset in terms of RMSE and PEMO-Q measures for PCA, AE, several architectures of DAEs (displayed in the legends of the figures), LSTM-AE and two different architectures of VAEs with different β values (also listed in the legend of the figures), all models having (tanh, lin) as pair of activation functions.

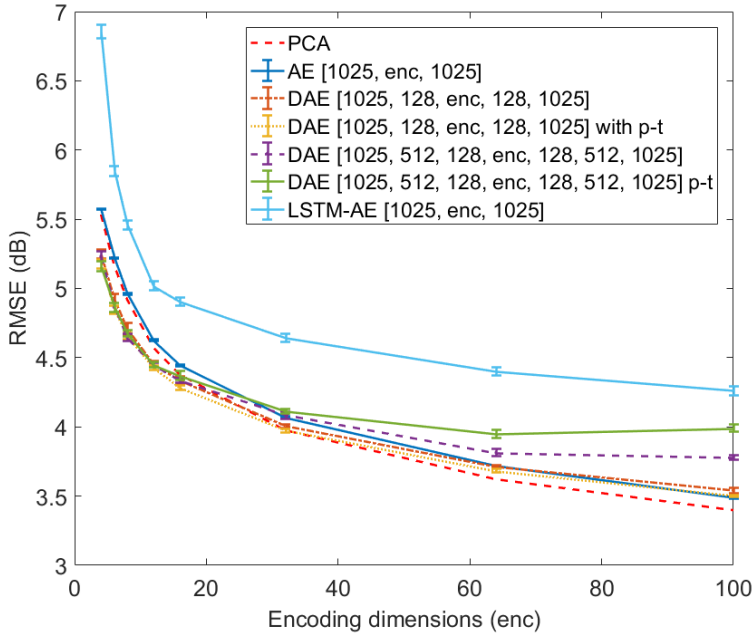
If we compare these two figures with the ones obtained on the NSynth dataset (respectively Figure 3.8 and Figure 3.9), a few observations are worth commenting. First, globally, the results are better for the Arturia dataset than for NSynth, i.e. the RMSE is lower for encoding dimensions lower than 32 and the PEMO-Q significantly higher (except for the LSTM-AE that seems to perform slightly worse in terms of RMSE).

Then, in accordance with the results on NSynth, the RMSE (respectively the PEMO-Q measures) decreases (respectively increases) with respect to the dimension of the latent space, and the greater the β, the lower the reconstruction accuracy and quality in the case of VAEs. We can also notice that here again, the benefit of layer-by-layer pre-training strategy cannot be clearly evidenced.

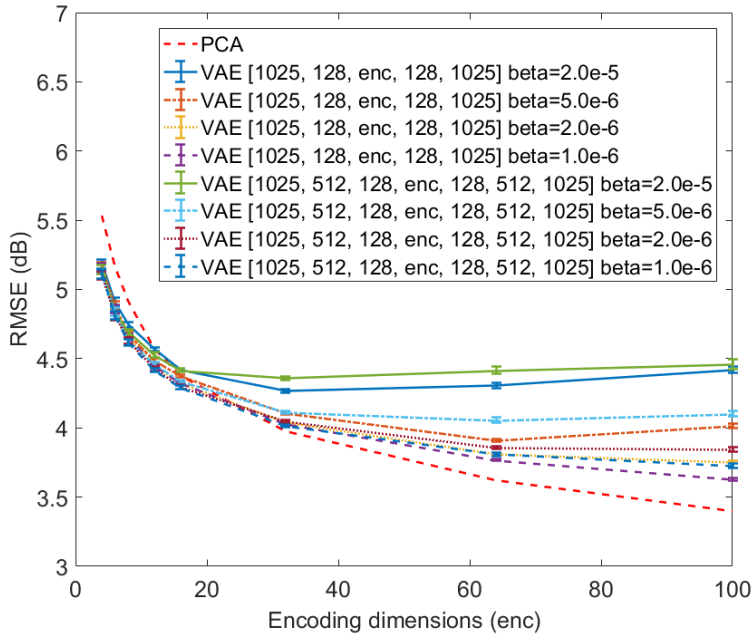
Interestingly, the phenomenon observed for RMSE at $enc = 100$ for the NSynth dataset (see Figure 3.8) is observable at $enc = 32$ on this dataset, see Figure 3.10, i.e. that below this dimension of the latent space, the deeper models perform significantly better than PCA (up to 7.7% for $enc = 6$) and then, for greater values, PCA clearly outperforms the other models. Concerning the PEMO-Q measures, the PCA seems to always slightly outperform every other model, but the values are close and the perceptual significance of these differences in the measures can be subject to discussion.

One explanation for the fact that PCA outperforms all the more complex models for an encoding dimension larger than 32 could be the ratio between the number of parameters to train the models and the size of the dataset. This hypothesis is comforted by the fact that the deeper the models, the less improvement in accuracy we have and the larger the gap between the model and PCA. Indeed, the Arturia dataset is a lot smaller than the subset of the NSynth dataset we used (133,164 against 1,157,310 for NSynth) and the models have much more parameters as the window length for the STFT is twice longer. For example, the LSTM-AE model presents more than 5,000,000 parameters to train and the deepest models of DAEs and VAEs have more than 1,000,000 which is too much for the training to be successful given the size of our dataset. Plus, concerning the deep models with the following architecture: $[1025, 128, enc, 128, 1025]$, the number of parameters remains below twice the number of input vectors while $enc \leq 32$, which could explain the change in behavior. The size of the dataset does not thus seem sufficient for large encoding dimensions and the models cannot learn properly and reach the wanted accuracy that PCA is able to achieve.

In order to confirm this hypothesis and show if reducing the complexity of the models can lead to better results, we tried to extract the exact same dataset but using a window length



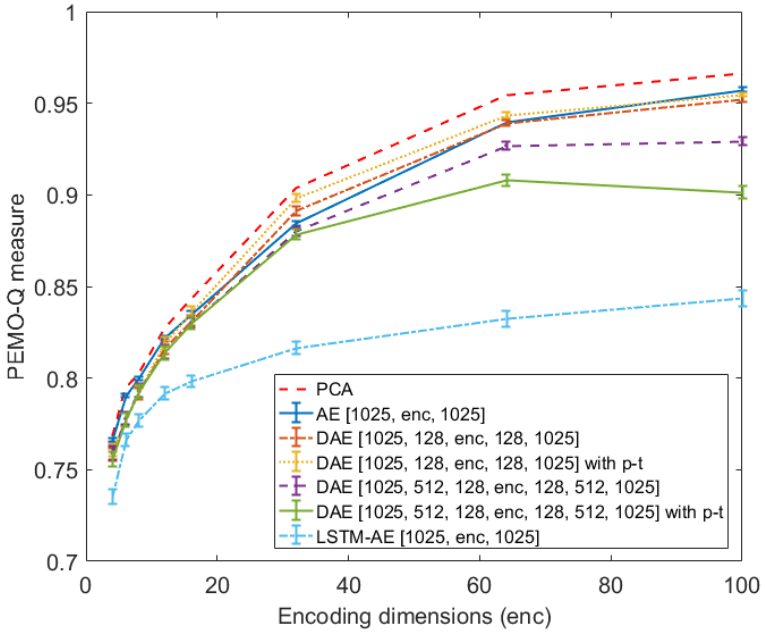
(a) PCA, AE, DAEs (two different architectures with and without layer-wise training) and LSTM-AE.



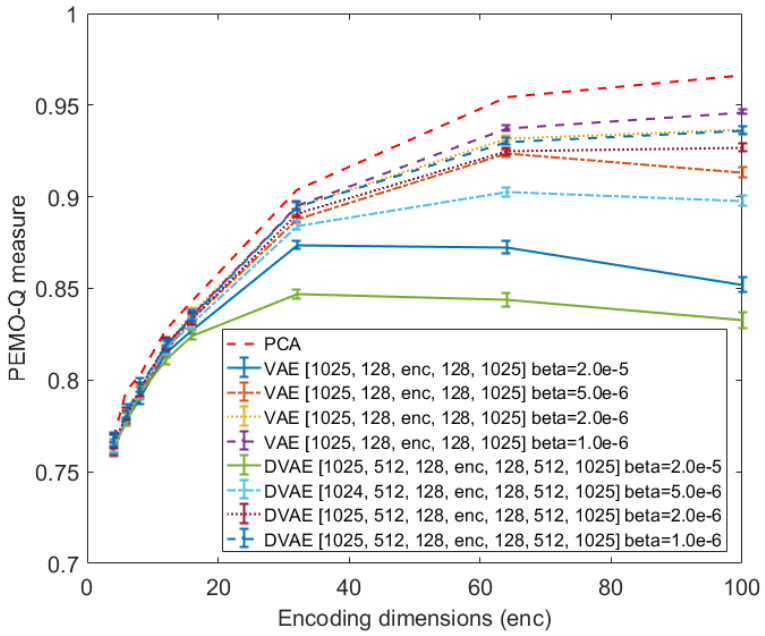
(b) VAEs with different architectures and different values of β (RMSE obtained with PCA is also recalled).

Figure 3.10: Reconstruction error (RMSE in dB) obtained with different AE-based models as a function of latent space dimension (trained on the Arturia dataset).

of 1024 samples instead of 2048, resulting thus in STFT length of 513 which is the input size



(a) PCA, AE, DAEs (two different architectures with and without layer-wise training) and LSTM-AE.



(b) VAEs with different architectures and different values of β (measures obtained with PCA are also recalled).

Figure 3.11: PEMO-Q measures obtained with different AE-based models as a function of latent space dimension (trained on the Arturia dataset).

of the models. This thus doubles the number of input vectors in the dataset and allows to

greatly reduce the number of parameters to train in the models. Figure 3.12 and Figure 3.13 report the results obtained with this new setting in terms of respectively RMSE and PEMO-Q measures. For more convenience and a better readability of the curves, Figure 3.12a has been rescaled, hiding parts of the results for the LSTM-AE model. The complete figure is available in Appendix D.1.

The first observation we can make is that, despite the reduction of the number of trainable parameters of the LSTM-AE, there are still too numerous (more than 1,300,000) and the accuracy of the model remained almost the same. Nonetheless, from these two new figures we can identify several differences compared to the previous ones. Indeed, in terms of RMSE, it is clear that PCA does not outperform DAEs anymore for $enc = 32$ but that this happens for encoding dimensions greater than 64. For VAEs, the results are less obvious but we can still notice some improvements. Concerning the PEMO-Q measures, the observations are quite similar and we can see that DAEs as well as VAEs perform much better than for a temporal window of 2048 samples, DAEs presenting results almost indistinguishable from PCA.

From these results, we can thus say that, by increasing the size of the dataset (e.g. by generating more examples of synthesizer sounds, or maybe by slightly modifying the presets in order to get more various sounds) it is possible to further improve the quality of the synthesis achieved by AE-based models, which is a very promising result for the future.

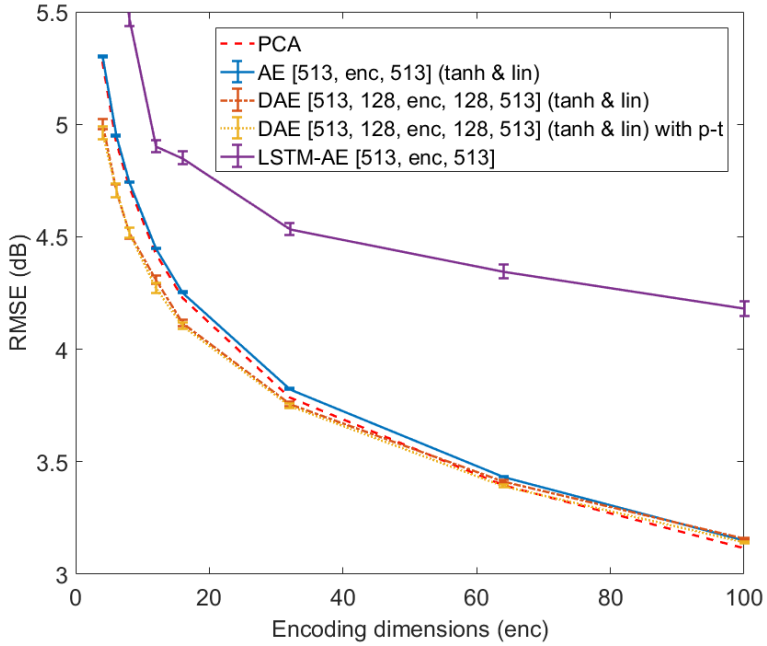
Audio examples of reconstructed samples from these two versions of the Arturia dataset using the different AE-based models are also available on the companion webpage.

3.3.4.2 Cross-correlation of latent dimensions

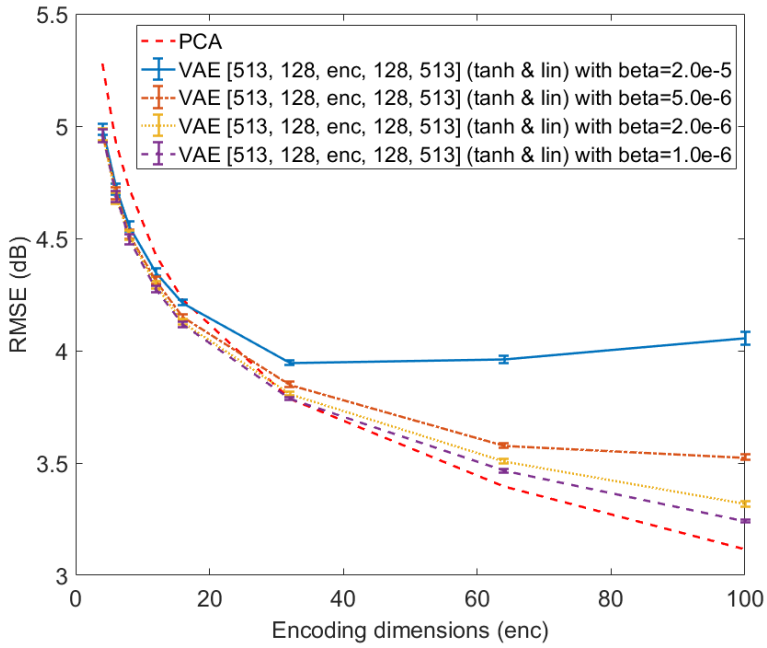
As already introduced earlier, now we will report further analyses aiming at investigating how the extracted latent dimensions may be used as control parameters by a musician. In the present sound synthesis framework, such control parameters are expected to respect (at least) the following constraints: to be as decorrelated as possible in order to limit the redundancy in the spectrum encoding, and to have a clear and easy-to-understand perceptual meaning.

In the present study, we focused on the first constraint by comparing PCA, DAEs, LSTM-AE and VAEs in terms of correlation of the latent dimensions. More specifically, the absolute values of the correlation coefficient matrices of the latent vector $\mathbf{z}$ were computed on each sound from the test datasets. Figure 3.14 reports the mean values averaged over all the sounds of the NSynth test dataset and Figure 3.15 on the Arturia test dataset. For the sake of clarity, we present here the results obtained for a latent space of dimension 16 only for one model of DAE ($[F, 128, 16, 128, F]$ (tanh & lin) with end-to-end training) and for VAEs with the same architecture ($[F, 128, 16, 128, F]$ (tanh & lin)) and different values of β (from 1.10^{-6} to 2.10^{-5}).

As could be expected from the complexity of its structure, we can see that, for both datasets, the LSTM-AE extracts a latent space where the dimensions are significantly correlated with each other. Such additional correlations may come from the sound dynamics which provide redundancy in the prediction. We can also see that PCA and VAEs present similar behaviors with much less correlation of the latent dimensions, which is an implicit property of these models. Interestingly, and in accordance with equation (3.10), we can notice that the higher the β, the more regularized the VAE and hence the more decorrelated the latent dimensions. Importantly, both Figure 3.14 and Figure 3.15 clearly show that for

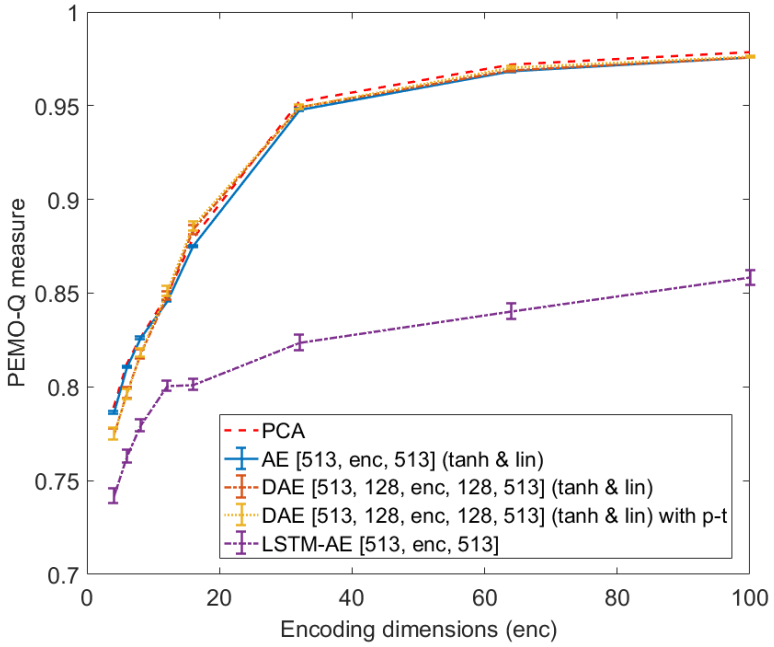


(a) PCA, AE, DAEs (with and without layer-wise training) and LSTM-AE.

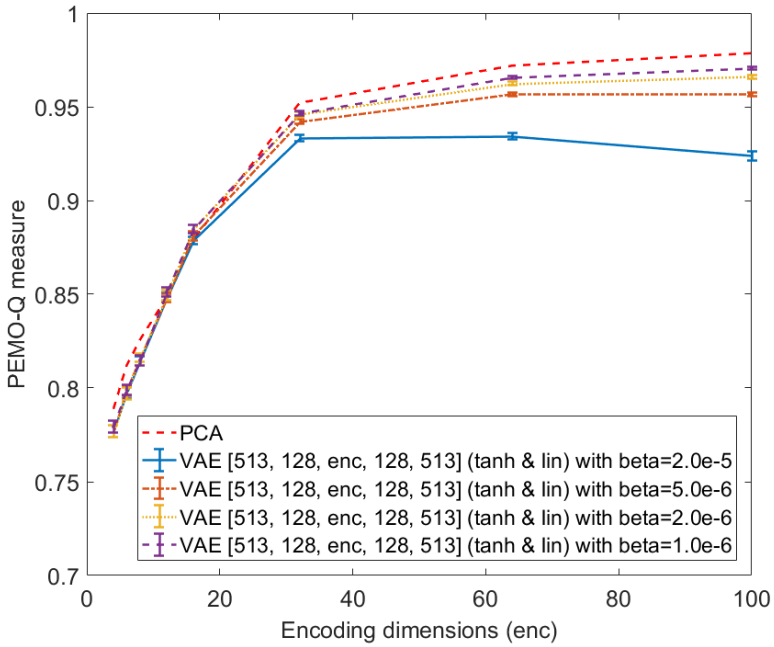


(b) VAEs with different values of β (RMSE obtained with PCA is also recalled).

Figure 3.12: Reconstruction error (RMSE in dB) obtained with different AE-based models as a function of latent space dimension (trained on the Arturia dataset computed with smaller temporal windows - 1024-point STFT).



(a) PCA, AE, DAEs (with and without layer-wise training) and LSTM-AE.



(b) VAEs with different values of β (measures obtained with PCA are also recalled).

Figure 3.13: PEMO-Q measures obtained with different AE-based models as a function of latent space dimension (trained on the Arturia dataset computed with smaller temporal windows - 1024-point STFT).

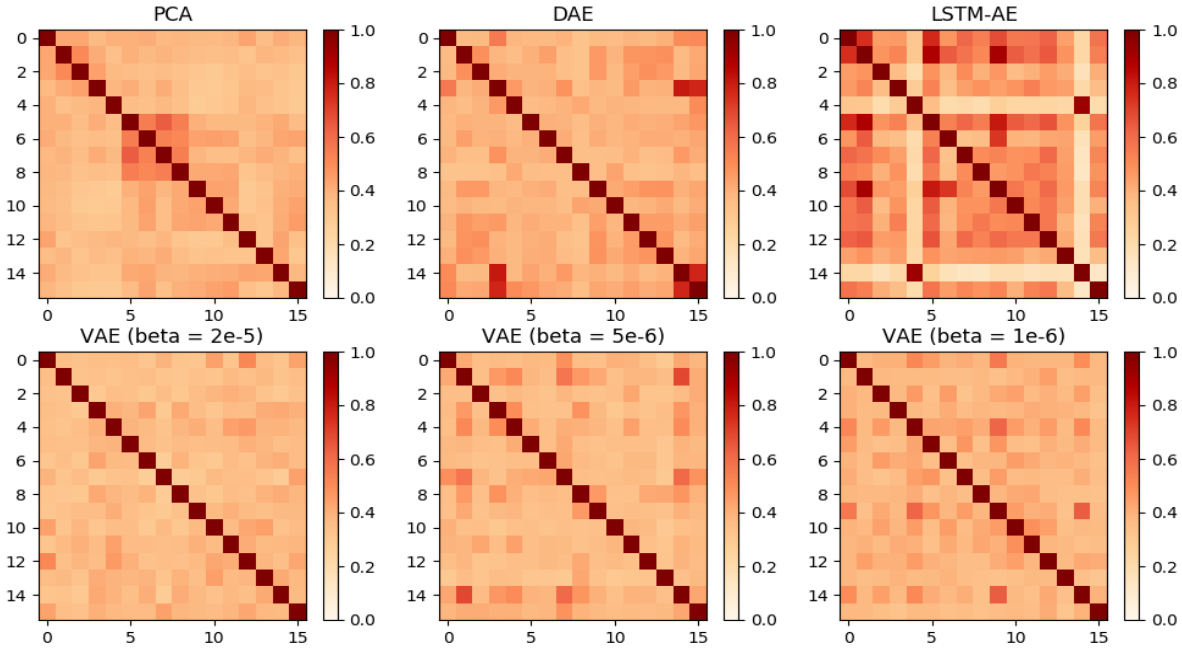


Figure 3.14: Correlation matrices of the latent dimensions (average absolute correlation coefficients) for PCA, DAE, LSTM-AE and VAEs trained on the NSynth dataset.

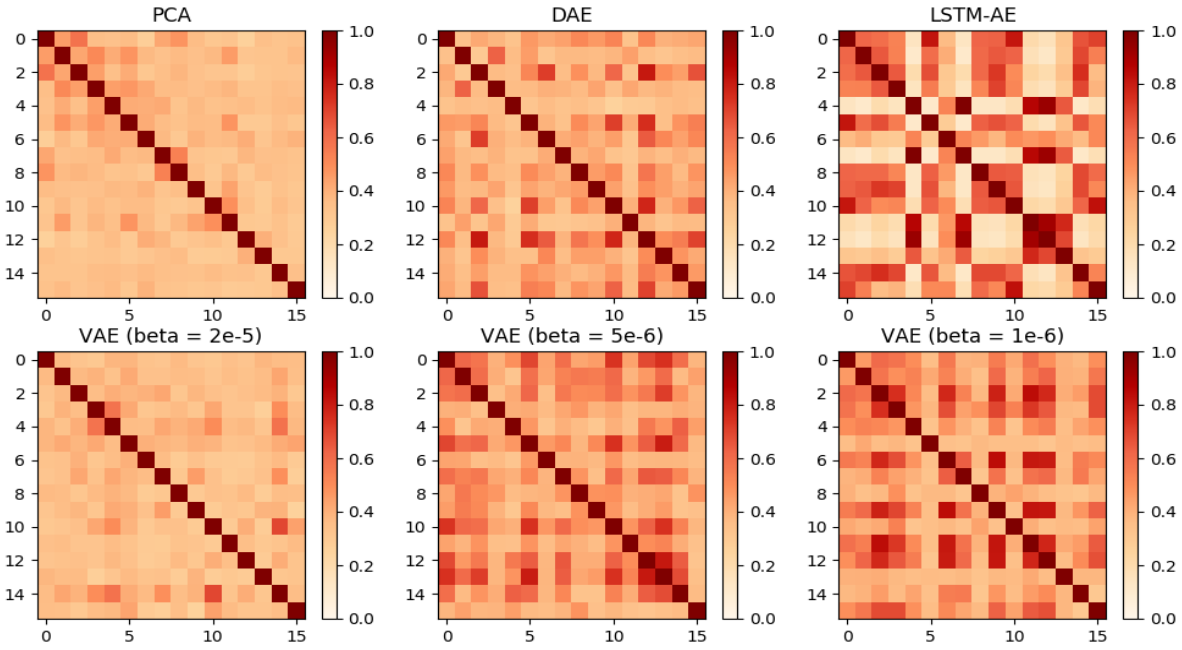


Figure 3.15: Correlation matrices of the latent dimensions (average absolute correlation coefficients) for PCA, DAE, LSTM-AE and VAEs trained on the Arturia dataset.

a well-chosen β value, the VAE can both extract latent dimensions that are much less correlated than for corresponding DAEs, which makes it a better candidate for extracting good control parameters, while allowing fair to good reconstruction accuracy, see Figure 3.8b and

Figure 3.10b. The β value has thus to be chosen wisely in order to find the optimal tradeoff between decorrelation of the latent dimensions and reconstruction accuracy.

Moreover, the similarity between the two figures seems to indicate that the organization of the latent space is independent of the dataset that has been used for training the model.

3.3.4.3 AE-based sound morphing

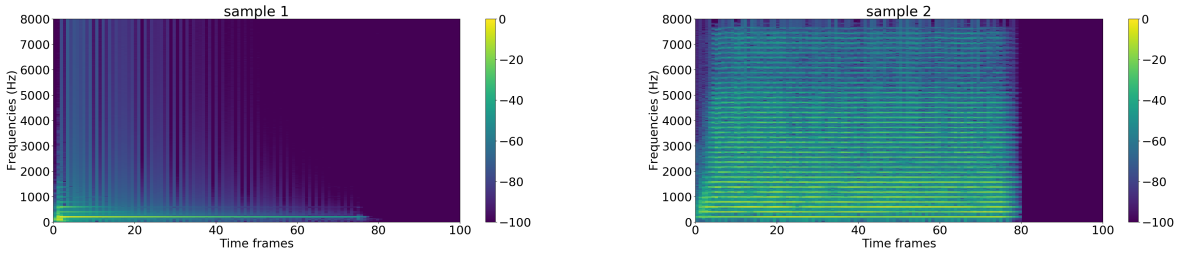
Finally, as a prior experiment to exploring the sound space by navigating through the latent space, we investigated how these latent spaces could be used for sound morphing, i.e. interpolation between two sound samples from both datasets.

To do so, we selected a series of pairs of sounds from one dataset (either NSynth or the Arturia dataset) where the two sounds in a pair presented different characteristics. For each pair, we proceeded to separate encoding, entry-wise linear interpolation of the two resulting latent vectors, decoding, and finally individual signal reconstruction with ISTFT and the G&L algorithm to reconstruct the phase spectrogram [Griffin and Lim 1984]. We experimented different degrees of interpolation between the two sounds: $\hat{\mathbf{z}} = \alpha\, \mathbf{z}_1 + (1-\alpha)\, \mathbf{z}_2$, with $\mathbf{z}_i$ the latent vector of sound i, $\hat{\mathbf{z}}$ the new interpolated latent vector, and $\alpha \in \{0, 0.25, 0.5, 0.75, 1\}$ (this interpolation is processed independently on each pair of vectors of the time sequence). For every interpolation, we initialized the G&L algorithm using the original phase spectrogram of the sample from the pair with the highest weight coefficient (either α or $(1-\alpha)$). The same process was applied using the different AE models we introduced earlier.

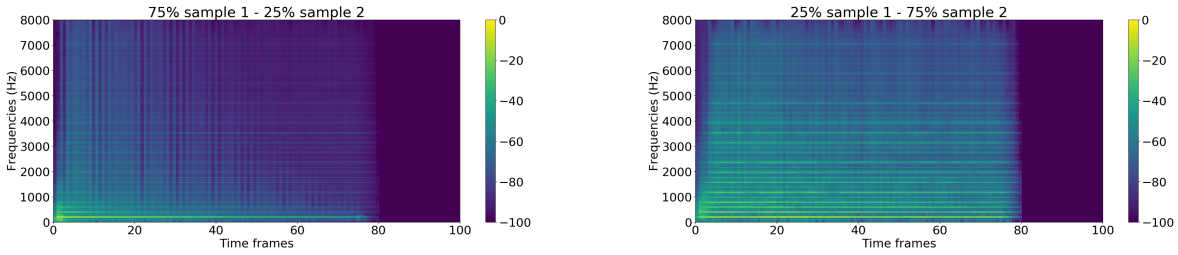
In Figure 3.16 and Figure 3.17 are displayed two examples of results obtained with PCA, DAE (with layer-wise pre-training), LSTM-AE and VAE (with $\beta = 1.10^{-6}$), with an encoding dimension of 32 for respectively the NSynth dataset and the Arturia dataset.

From Figure 3.16, qualitatively we note that interpolations in the latent space lead to a smooth transition between source and target sound. By increasing sequentially the degree of interpolation, we can clearly go from one sound to another in a consistent manner, and create interesting hybrid sounds. The results obtained using PCA interpolation are (again qualitatively) below the quality of the other models. The example spectrogram obtained with interpolated PCA coefficients is blurrier around the harmonics and some audible artifacts appear. On the opposite, the LSTM-AE seems to outperform the other models by better preserving the note attacks (see comparison with VAE in Figure 3.16). A more detailed interpolation figure (showing all the α values) can be found in Appendix D.2.1. More interpolation examples along with corresponding audio samples can also be found in the companion webpage.

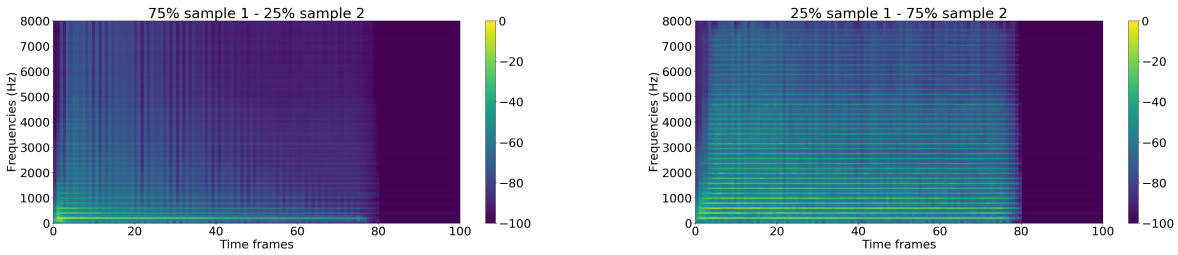
From Figure 3.17 we can see that for Arturia dataset also the transition from one sample to another seems (qualitatively) to be done very smoothly and in a consistent manner, creating interesting hybrid sounds (see figure in Appendix D.2.2 for a more detailed transition between the two samples for the listed models, other examples are also avaible in the companion webpage). As expected from the observations in Section 3.3.4.1, the LSTM-AE shows the poorest results in terms of quality and accuracy. The spectrograms are blurrier than with other models and the resulting audio signal lacks many details. However, the other models demonstrate very promising results. PCA and DAE are almost indistinguishable and marginally outperform VAE which seems to produce slightly noisier signals. What is interesting to note is that all the models generated hybrid sounds while interpolating that



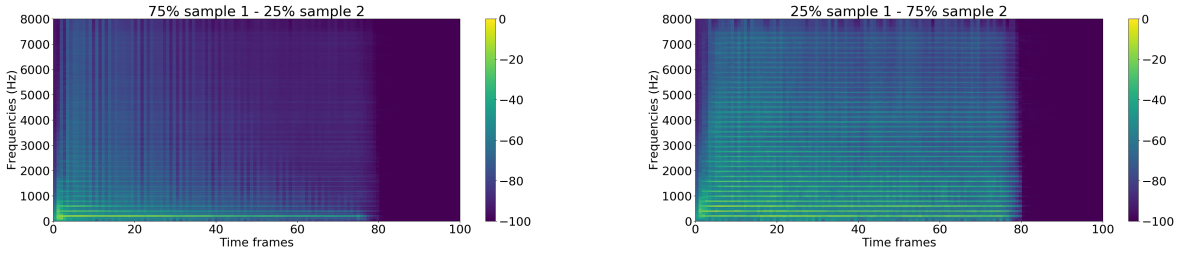
(a) Original samples – Left: bass_electronic_010-055-100, Right: brass_acoustic_050-055-100



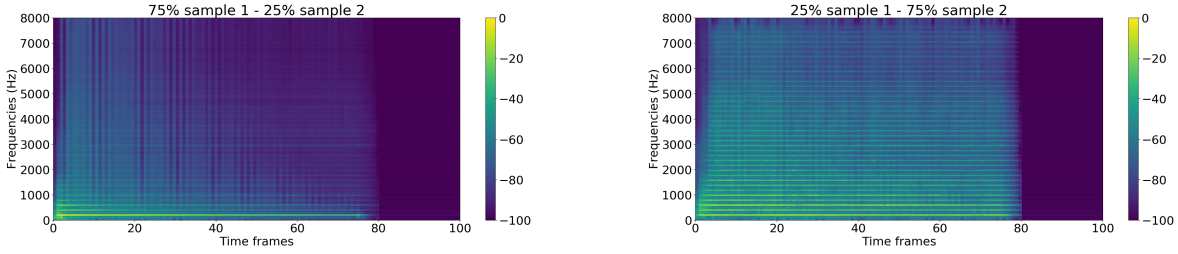
(b) PCA



(c) DAE

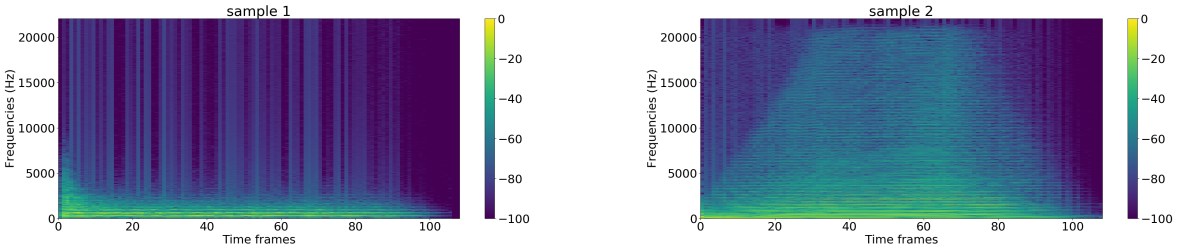


(d) LSTM-AE

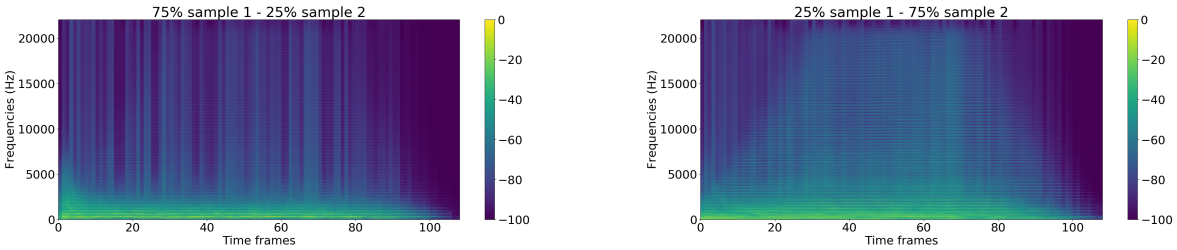


(e) VAE

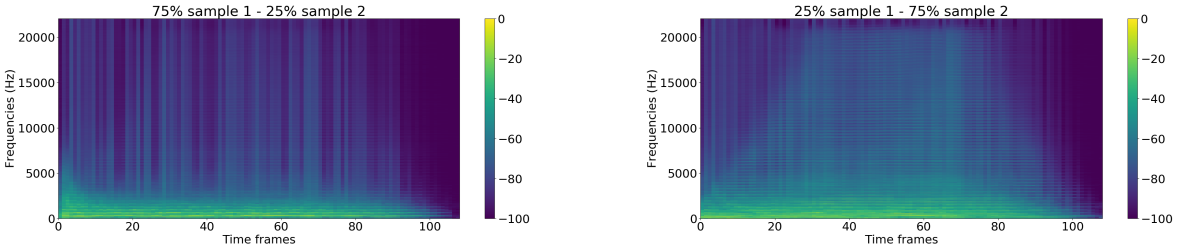
Figure 3.16: Examples of decoded magnitude spectrograms after sound interpolation of 2 NSynth samples (top) in the latent space using respectively PCA (2nd row), DAE (3rd row), LSTM-AE (4th row) and VAE (bottom). A more detailed version of the figure can be found in Appendix D.2.1 or at http://www.gipsa-lab.fr/~fanny.roche/PhD_thesis.html.



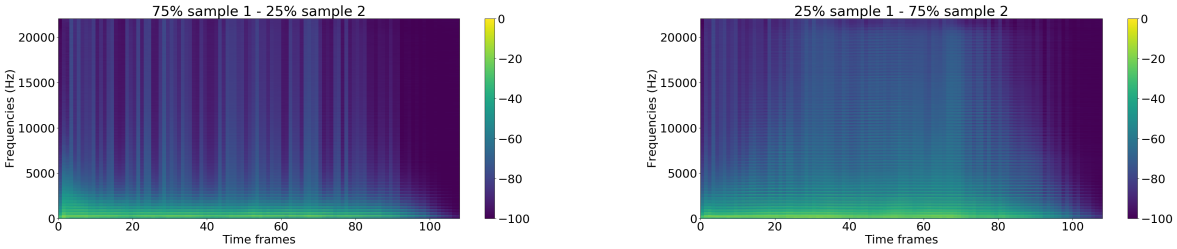
(a) Original samples



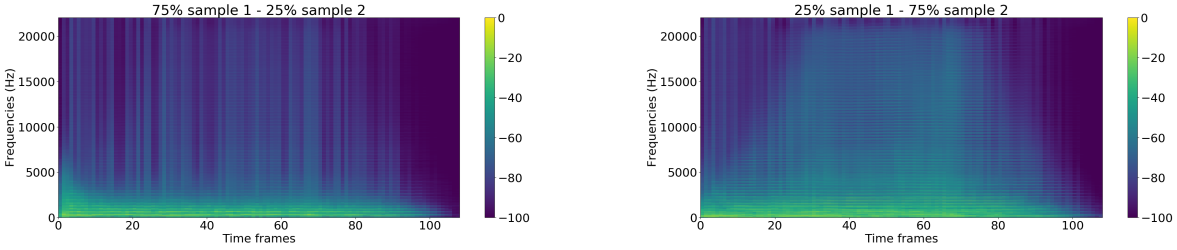
(b) PCA



(c) DAE



(d) LSTM-AE



(e) VAE

Figure 3.17: Examples of decoded magnitude spectrograms after sound interpolation of 2 Arturia samples (top) in the latent space using respectively PCA (2nd row), DAE (3rd row), LSTM-AE (4th row) and VAE (bottom). A more detailed version of the figure can be found in Appendix D.2.2 or at http://www.gipsa-lab.fr/~fanny.roche/PhD_thesis.html.

sound quite similar or at least have a lot of sonic characteristics in common. In other words, linearly interpolating with the different models seems to lead to almost the same hybrid sounds.

3.3.5 Conclusion

From the experiments conducted on both the publicly available database NSynth and our Arturia dataset of synthesizer sounds, we can draw several conclusions. First, and contrary to the literature on image processing, shallow AEs do not systematically outperform PCA in terms of reconstruction accuracy. Then, the best performance in terms of signal reconstruction is always obtained with more complex architectures such as DAEs or LSTM-AE (if the size of the dataset is appropriate). Finally, the VAEs lead to fair-to-good reconstruction accuracy while constraining the statistical properties of the latent space, insuring some amount of decorrelation across latent coefficients and limiting their range. These latter properties make the VAEs good candidates for our targeted sound synthesis application.

Moreover, the AE-based models have all shown, to some extent, that they are able to extract representation spaces that lead to smooth and consistent transitions from a source to a target sound, creating thus interesting hybrid sounds. This is very promising for the target application which consists in creating new sounds by navigating through this high-level representation space.

However, we evidenced the fact that the size of the dataset needs to be controled wisely in order to be able to train the models correctly and reach a good reconstruction accuracy, and consequently a good sound synthesis audio quality, but that this requirement can be satisfied without too many difficulties.

So far, no evidence of any connection between these extracted latent spaces and the perceptual dimensions introduced in Chapter 2 has been found. In order to investigate this potential link, we developed a non real-time Max/MSP prototype that will be presented in the next section.

3.4 Max/MSP prototype

As stated before, in order to explore the latent space and its effects on the reconstructed signal, we implemented a Max/MSP non real-time prototype reproducing the global methodology presented in Figure 3.1. Max/MSP is a visual programming tool initially developed by IRCAM in Paris and now maintained and made available by Cycling 74 (https://cycling74.com/). It allows to combine together audio processing units from a large library of analysis, synthesis and processing modules in order to synthesize sound in real-time. For the prototype we used the Open Sound Control protocol (OSC) which is a Python package that allows to send data from a graphical interface (here Max/MSP, see Figure 3.18) to a Python executing program.

3.4.1 Main principle

The graphical interface (GUI) displays a "Load" button in the top left corner (see Figure 3.18) that allows to load a sound file (.wav) in the Python program where it is instanta-

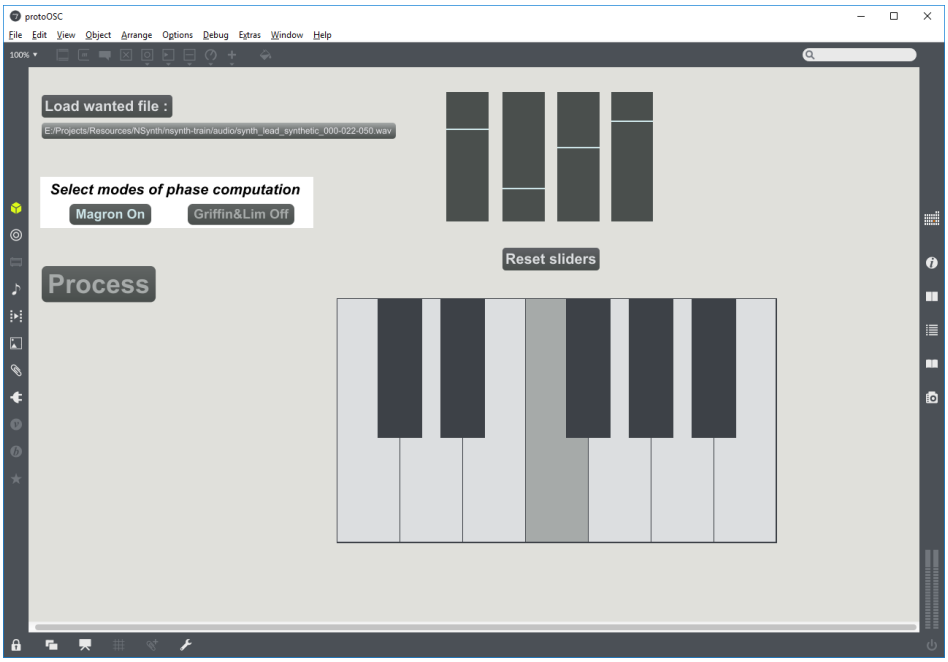


Figure 3.18: Graphical user interface of the non real-time Max/MSP prototype.

neously converted into normalized log-magnitude spectra (as explained in Section 3.3.2) and encoded using one of the various AE-based models presented in Section 3.2.2. In the top right corner, four sliders are presented corresponding to the four main latent dimensions extracted by the model according to the explained variance. The main idea behind this is to organize the dimensions so that the ones presented to the user are the ones which have the largest impact on the reconstructed signal, either in the sense of the largest variability of the latent coefficients or in the sense of the largest variance of the reconstructed spectra (just like PCA). These sliders are meant for the user to be able to modify the trajectories of these particular latent dimensions (as illustrated in Figure 3.2 and detailed in corresponding explanations). The new trajectories of the latent dimensions are computed using the sliders values (going from 0.1 to 10 times the standard deviation of the latent dimension, 1 corresponding to the original trajectory):

$$\tilde{\mathbf{z}}_k = (\mathbf{z}_k - \boldsymbol{\mu}_k)\ s_k\ +\ \boldsymbol{\mu}_k \tag{3.13}$$

where $\tilde{\mathbf{z}}_k$ is the new trajectory of the k^{th} latent coefficient, $\mathbf{z}_k$ its original trajectory and $\boldsymbol{\mu}_k$ the mean of the variable $\mathbf{z}_k$ over all the training set, s_k being the value of the corresponding slider.

Then the "Process" button launches the computation of the new trajectories and the whole synthesis process. For that, first, the signal is reconstructed using the chosen AE-based model and the original phase spectra with ISTFT and OLA. The residue between this reconstructed signal and the original one is then computed in order to store what could not be captured by the model. Afterward, if the values of the sliders are not all set to their neutral position on the interface (corresponding to $s_k = 1$ and implying no trajectory modification and thus to directly output the original signal), the modified trajectories of the latent dimensions (3.13) are sent to the decoder and the phase spectra are reconstructed using the resulting decoded

magnitude spectra as described in Section 3.2.3.3. For the phase reconstruction, the user can select the method thanks to the buttons in the white box in Figure 3.18, i.e. either the linear phase unwrapping (referred to as *Magron* in the GUI), the Griffin and Lim algorithm or a combination of both (the phase unwrapping method being used as an initialization of G&L). After applying ISTFT and OLA, the residue computed earlier is eventually summed to generate the final reconstructed signal. The purpose of adding the residue at the end is to reintegrate what the model has not been able to learn and only see the impact of the latent changes on the reconstructed sound.

Finally, the user can listen to the reconstructed signal by clicking on the keys of the keyboard which allows to play different notes of the same instrument (if the selected AE-based model has been trained on the NSynth dataset and the .wav a part of it, otherwise only one note can be played as the Arturia dataset is mono-pitch).

3.4.2 Qualitative observations and comments

The main objective of this prototype was to evaluate qualitatively and at a very small scale (no formal perceptual study was done at this stage) the quality of the sound synthesis using the various reduction dimension models and if a connection between the main dimensions extracted by these AE-based models and the perceptual dimensions evidenced in Chapter 2 could be made.

What we noticed from experimenting the prototype on various sound samples was that, in accordance with the observations of Section 3.3, the AE-based models are adapted to quality sound synthesis in a context of an analysis-synthesis process (i.e. no user modification). We could also validate the two phase reconstruction methods as they give very good sound quality results. However, playing with the sliders turned out not to be perceptually meaningful. Modifying the trajectories led to important perceptual changes of the generated sound in a continuous manner, but with no perceptual relevance. This hence highlighted the difficulty to match the extracted dimensions with perceptual dimensions as direct control and evidenced the need for some kind of supervision of the models during training.

3.5 Conclusion

In this chapter, we introduced and detailed the different unsupervised machine learning methods we explored and compared for the extraction of a representation space with interesting interpolation and extrapolation properties while allowing the generation of new sounds with good quality.

First, we presented the global analysis-transformation-synthesis methodology we applied for trying to answer the second main challenge of this thesis project. Inspired by the previous studies investigating the use of deep learning for music sound synthesis, we focused on AE-based models. We detailed their principle, both theoretically and practically, and the potential implications in the data representation necessary to train the different models.

We then realized two extensive comparative studies of the proposed models on two different datasets: the NSynth dataset which is a multi-pitch and multi-instrument dataset of 4 seconds long sound samples and the Arturia dataset, which contains mono-pitch synthesizer sounds of between 2 and 2.5 seconds long. The different models were compared in terms of accuracy of

the reconstructed signal in the context of an analysis-synthesis process (no modification of the latent space) using both an objective physical measure (RMSE) and an objective perceptual measure (PEMO-Q), in terms of the correlation of the latent dimensions extracted and finally in terms of sound morphing, i.e. how the models were adapted for extracting a latent space in which interpolation between sounds leads to interesting hybrid sounds.

Finally, in order to establish qualitatively and at a very small scale, a possible connection between the dimensions of the latent spaces extracted and the perceptual dimensions evidenced in Chapter 2, we experimented a non real-time Max/MSP prototype we developed and described in a last section.

From all the experiments we realized, several conclusions can be drawn and the questions raised in Section 3.1 can be answered. First, whether it is on a standardized multi-instrument dataset or on a more realistic dataset considering our application, AE-based models proved to be well adapted for proposing a good quality sound synthesis while extracting a latent space in which it is possible to navigate smoothly and without discontinuities. In particular, such models demonstrated a good capacity to create possibly interesting hybrid sounds by interpolating between two different sounds of the dataset in the latent space, which is an evidence of their ability to generate new sounds. Moreover, thanks to their probabilistic framework, VAE-based models insure some amount of decorrelation across the latent coefficients which makes them good candidate for control. However, no link between these latent spaces and the evidenced perceptual dimensions, nor dimensions with any perceptually relevant sense emerged naturally, highlighting thus the need for an additional perceptual supervision while training the model. This will be the main focus of the next chapter.

Chapter 4

Towards weak supervision of autoencoder models using timbre perception

Contents

4.1 Motivation **94**
4.2 Perceptual regularization methodology **94**
4.2.1 Timbre-based regularization methodology 94
4.2.2 Weakly supervised learning 96
4.2.3 Proposed methodology 97
4.2.3.1 2-step learning procedure 97
4.2.3.2 Perceptual regularization metric 98
4.3 Regularizing VAE-based models using perceptually meaningful continuous labels **98**
4.3.1 Datasets 98
4.3.2 Data pre-processing 99
4.3.3 Regularized AE-based models implementation 99
4.3.4 Experimental results 99
4.3.4.1 Analysis-synthesis 100
4.3.4.2 Latent space organization 103
4.4 Perceptual evaluation of regularized AE-based models **105**
4.4.1 Stimuli 105
4.4.2 Protocol of the perceptual study 106
4.4.3 Results analysis 107
4.4.3.1 Applied methodology 107
4.4.3.2 Results 109
4.4.4 Discussion 111
4.5 Conclusions and perspectives **111**

In the previous chapter, we compared and evaluated different unsupervised AE-based methods and showed that they were relevant tools, in particular the VAE, to extract a representation space to navigate smoothly into the sonic space created by a dataset of samples. This opens the way to create new sounds. However, from these experiments, no link with the perceptual dimensions evidenced in Chapter 2 has naturally emerged. This motivated the addition of supervision to the model in order to obtain perceptually relevant control parameters

for synthesis. In this chapter we will focus on a weakly supervised method we investigated in order to encourage some of the dimensions of the latent space to match these perceptual dimensions.

First we will present the proposed weakly supervised training methodology used to train the VAE. Then we will present the experimental protocol and results.

4.1 Motivation

One of the main objectives of this research project was to allow the perceptual control of the synthesis. We therefore found it necessary to add perceptual meaning to the dimensions extracted by the models.

In Chapter 2, the second perceptual test we conducted allowed us to get a quantified subjective evaluation of a subset of the Arturia dataset along 8 perceptual dimensions. These evaluated stimuli actually represent a new (smaller) dataset of samples that are annotated with respect to these perceptual dimensions and which will be useful to supervise the training of the model.

Given the results obtained in Chapter 3, we decided to focus on the VAE model and attempt to add a perceptual supervision during the training by using these annotated samples. The central questions of this chapter are: how can we add a perceptual supervision during the training of the model in order to "force" the meaning of the latent dimensions? And is it possible to change the behavior of the extracted latent space using a very limited set of annotated samples?

4.2 Perceptual regularization methodology

4.2.1 Timbre-based regularization methodology

As already stated in Section 1.3.3.1, in [Esling et al. 2018b], the authors started to investigate how to incite a VAE model to extract a latent space that matches the topology of the perceptual timbre space. To do so, they added a perceptual regularization term to the weighted VLB of the VAE (3.10) that encourages the distances in the latent space $\mathbf{z}$ to have the same structure as the distances in the timbre space $\mathcal{T}$:

$$\mathcal{L}(\phi, \theta, \beta, \mathbf{x}) = \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}[\log\, p_\theta(\mathbf{x}|\mathbf{z})] - \beta\ \ \mathrm{D}_{\mathrm{KL}}\Big(q_\phi(\mathbf{z}|\mathbf{x})|p_\theta(\mathbf{z})\Big) + \alpha\ \mathcal{R}(\mathbf{z}, \mathcal{T}) \tag{4.1}$$

where $\mathcal{R}(\mathbf{z}, \mathcal{T})$ is the additional perceptual regularization term and α its corresponding weighting factor which has the same role as β for the classic VAE regularization term that we introduced previously. Figure 4.1 illustrates this approach. As depicted in the figure, the used timbre space $\mathcal{T}$ relies on several MDS studies that were realized using datasets of orchestral instruments, see details in the paper.

In [Esling et al. 2018b], taking inspiration from the t-SNE algorithm that has the same purpose[1] [van der Maaten and Hinton 2008], the authors assumed the relationships between

[1] Two samples that are close in the first high-dimensional space will have close coordinates in the new (low-dimension) space.

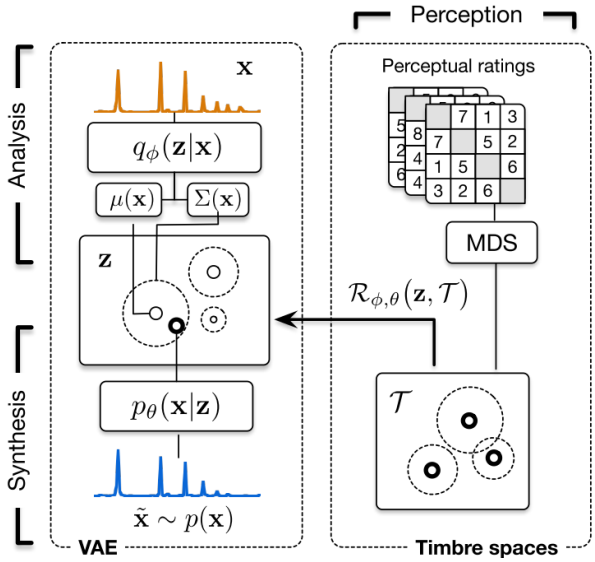


Figure 4.1: Diagram of the perceptually-regularized VAE extracted from [Esling et al. 2018b].

two different samples i and j in the latent space $\mathcal{D}^{\mathbf{z}}_{i,j}$ to be computed using a conditional Gaussian density and the distances $\mathcal{D}^{\mathcal{T}}_{i,j}$ in the timbre space $\mathcal{T}$ using a Student-t distribution:

$$\begin{cases} \mathcal{D}^{\mathbf{z}}_{i,j} = \dfrac{\exp(-||\mathbf{z}_i - \mathbf{z}_j||^2/2\sigma_i^2)}{\sum_{k\neq i}\exp(-||\mathbf{z}_i - \mathbf{z}_k||^2/2\sigma_i^2)} \\ \mathcal{D}^{\mathcal{T}}_{i,j} = \dfrac{(1+||\mathcal{T}_i - \mathcal{T}_j||^2)^{-1}}{\sum_{k\neq i}(1+||\mathcal{T}_k - \mathcal{T}_i||^2)^{-1}} \end{cases}$$

where σ_i^2 is the variance of the Gaussian centered in $\mathbf{z}_i$. The $\mathcal{R}(\mathbf{z},\mathcal{T})$ regularization term is then calculated using the sum of KL divergences between the two different distributions of distances in the two different spaces. In another paper they published on the same topic the same year, [Esling et al. 2018a], they also experimented other distances for the different spaces. First, they tested Euclidian distance for both spaces where the $\mathcal{R}(\mathbf{z},\mathcal{T})$ term is also computed using a Euclidian distance between the two distances terms obtained in the different spaces. Then, they explored the modeling of the perceptual ratings using a univariate Gaussian distribution. For more details about the used distances and regularization, the reader is referred to the two corresponding papers [Esling et al. 2018a; Esling et al. 2018b].

For training the models, they used the Studio On Line (SOL) database of orchestral instruments[2] from which they extracted, for every sample, a single non-stationary Gabor transform (NSGT) frame to serve as input. During the training, they explored two values of α (0.1 and 1) and a 2-step procedure, first starting without the perceptual regularization during a certain number of epochs and then inserting the additional regularization during another number of epochs.

This method seemed very promising and was inspirational for us, however there are several important differences between their target application and ours that have to be taken into

[2]The SOL database is the supporting database of the IRCAM's Orchid sofwtare https://forum.ircam.fr/projects/detail/orchids/.

account. First, the structure of the latent space extracted with their model is expected to follow the structure of the timbre space of orchestral instruments evidenced with the MDS studies. This is a very interesting property when the application is to wander the latent space smoothly with some perceptual relevance. However, this does not give perceptually meaningful controls for the synthesis. In other words, the different dimensions extracted by the VAE do not match individual perceptual dimensions from the timbre space. The only property we can be sure of is that if two samples are close in the timbre space, then their latent representations will be close to each other, which is not our primary goal in this thesis.

Secondly, for these studies the authors had at their disposition a fully labeled dataset of samples, i.e. each sample was given with its corresponding coordinates in the timbre space. In our context, unfortunately, only a very small subset of the dataset has been perceptually annotated thanks to the listening test and is insufficient for training a deep model correctly. We thus had to investigate weakly supervised methods in order to see how to handle the training of deep models using a very small quantity of labeled samples together with a great amount of unlabeled data.

4.2.2 Weakly supervised learning

Weakly supervised learning is another type of learning method that lies in between supervised and unsupervised learning. Gathering a completely labeled dataset in order to solve a particular task in a fully supervised framework can be very difficult and time consuming, and learning a supervised model with very few samples is often not possible. Some studies thus investigated the use of weakly labeled or unlabeled data together with a small amount of fully labeled data in order to overcome this issue, see studies presented in [Zhou 2017].

There exist different types of weak supervision [Zhou 2017]: the inexact supervision, where the available labels are coarse grained labels (e.g. image description instead of objects description for image labeling); inaccurate supervision where the labels are not necessarily the ground truth (e.g. noisy labels or labels that come from unreliable annotators); and the incomplete supervision where only a subset of the data is labeled. In the context of our project, our objectives and available data match perfectly the scope of the incomplete supervision and from the different methods dealing with incomplete supervision we will focus on semi-supervised learning where the model learns from both labeled and unlabeled data without any human intervention [Zhou 2017; Chapelle et al. 2006].

Most of the methods dealing with semi-supervised learning are used for classification problems [Chapelle et al. 2006; Zhou 2017], however such learning strategy seems to also be suitable for regression and some studies have investigated this issue, see the studies presented in [Kostopoulos et al. 2018]. Although there exist several methods for both classification and regression that can be very different from each other, they all rely on the smoothness assumption on the structure of the underlying distribution of data which states that if data samples are close, then their labels should also be close [Chapelle et al. 2006].

From the various semi-supervised techniques, two main trends for handling the mix of labeled and unlabeled data emerge. The first one consists in training a model by optimizing a 2-fold objective function (containing both a supervised criterion and an unsupervised criterion) using both the labeled ($\mathcal{X}_L$) and unlabeled ($\mathcal{X}_U$) datasets. This can be done either by inferring the missing labels of $\mathcal{X}_U$ [Nigam et al. 2000; Chapelle and Zien 2005; Zhou and

Li 2005; Hueber et al. 2015; Girin et al. 2017] or by considering them as missing data and not taking them into account when optimizing the objective function [Ranzato and Szummer 2008]. The other trend is to perform a 2-step learning procedure by first training the model in an unsupervised way on all the data (both $\mathcal{X}_L$ and $\mathcal{X}_U$) and then "fine-tuning" the model on the labeled dataset only ($\mathcal{X}_L$) [Hinton and Salakhutdinov 2007; Bengio et al. 2007; Chapelle et al. 2006; Oliveira et al. 2005].

From all these techniques, we got interested in a 2-step learning procedure inspired by the approach of [Hinton and Salakhutdinov 2007] where they first pre-train the model in an unsupervised manner using all the available data in order to extract "sensible, high-level features" that will then be refined using a fine-tuning stage on the labeled data only. This proposed process will be developed in the next sections.

4.2.3 Proposed methodology

In order to answer the last challenge of this thesis and get perceptually relevant dimensions for the control of the synthesis, we applied the exact same methodology as in Chapter 3 except that we used the results of the perceptual test described in Section 2.3 to train the model by means of a semi-supervised method.

As stated in Section 4.1, we focused on the VAE model and in this section we will present the semi-supervised technique we applied.

4.2.3.1 2-step learning procedure

In line with the approaches of [Hinton and Salakhutdinov 2007] and [Esling et al. 2018a], we investigated the use of a 2-step learning procedure to add perceptual supervision to our VAE model.

To do so, just as in [Esling et al. 2018a] or [Hinton and Salakhutdinov 2007], the idea was to start by training the model in an unsupervised fashion on both the unlabeled and the labeled datasets using the standard weighted VLB introduced in (3.10).

Then, so as to encourage the latent space to match perceptual dimensions, we inserted an additional regularization term in the VLB of the VAE, just as in [Esling et al. 2018a], see (4.1), and trained this "regularized" VAE using the annotated dataset only.

Finally, we investigated the repetition of these 2 steps in a sequential way to evaluate whether it could have an impact on the accuracy and the effectiveness of the perceptually regularized VAE and, in a way, improve the obtained results.

The proposed methodology can thus be summarized as:

- repeat the 2 steps N times:
  1. unsupervised pre-training:
     optimizing $\mathcal{L}(\phi, \theta, \beta, \mathbf{x}) = \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}[\log p_\theta(\mathbf{x}|\mathbf{z})] - \beta \, \mathrm{D}_{\mathrm{KL}}\Big(q_\phi(\mathbf{z}|\mathbf{x})|p_\theta(\mathbf{z})\Big)$
     on $\mathcal{X}_U \cup \mathcal{X}_L$,
  2. supervised fine-tuning:
     optimizing $\mathcal{L}(\phi, \theta, \beta, \mathbf{x}) = \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}[\log p_\theta(\mathbf{x}|\mathbf{z})] - \beta \, \mathrm{D}_{\mathrm{KL}}\Big(q_\phi(\mathbf{z}|\mathbf{x})|p_\theta(\mathbf{z})\Big) + \alpha \, \mathcal{R}(\mathbf{z}, \mathcal{P})$

on $\mathcal{X}_L$, where $\mathcal{R}(\mathbf{z}, \mathcal{P})$ is a regularization term based on the distance between the latent vector $\mathbf{z}$ and a vector gathering the individual perceptual dimensions resulting from our perceptual test, and defined in the next subsection.

4.2.3.2 Perceptual regularization metric

The goal we want to reach using these experiments is to get some dimensions of the latent space to match the perceptual dimensions evidenced in Chapter 2. To do so, our first very naive intuition was to try to force the 8 first dimensions of the latent space to match the ratings of the subjects on the different scales.

Yet, no assessment has been made on the linearity or non-linearity of the relationship between the use of the perceptual scales and the actual perception of the potential users. It was thus difficult to evaluate which metric was the most adapted to serve as perceptual regularization function. We therefore decided to keep it as simple as possible and use the MSE. We then have:

$$\mathcal{R}(\mathbf{z}, \mathcal{P}) = \text{MSE}(\mathbf{z}_{1:8}, \mathbf{l}) \tag{4.2}$$

where $\mathbf{z}_{1:8}$ corresponds to the vector of the 8 first latent dimensions extracted by the VAE and $\mathbf{l}$ the label vector corresponding to the average ratings of the input sound on the 8 perceptual scales.

4.3 Regularizing VAE-based models using perceptually meaningful continuous labels

In order to validate our proposed methodology aiming at perceptually regularize a VAE model, we investigated several configurations of the regularization and the learning procedure, and systematically compared objectively these newly implemented models to the classic VAE implementation. In this section, we will present these experiments and the results we obtained.

4.3.1 Datasets

For these experiments, we needed two different sets of samples, a labeled dataset $\mathcal{X}_L$ and an unlabeled dataset $\mathcal{X}_U$. As already introduced before, the labeled dataset consisted of the 80 samples annotated during the perceptual test with semantic scales we presented in Section 2.3.4.5. Concerning the unlabeled dataset, we used the Arturia dataset already introduced in Section 2.2.2.1 and containing 1,233 synthesizer sound samples.

Similarly as for these previous experiments we realized on AE-based models, we split the unlabeled dataset into a training set (80%) and a testing set (20%). In order to respect the 2-step methodology presented in the previous section, when splitting the database, we made sure that all the samples from the labeled dataset (i.e. the 80 samples that were annotated during the perceptual test with scales) were contained in the training set and not in the testing set. We thereby insured the unsupervised pre-training to be realized on $\mathcal{X}_U \cup \mathcal{X}_L$ and the supervised fine-tuning on samples from $\mathcal{X}_L$ that have already been seen by the model.

4.3.2 Data pre-processing

Unlabeled data We applied the exact same pre-processing of the data as in Section 3.3.2 except that we applied a Hamming window with a length of 1024 samples for computing the STFT as for the last experiments presented in Section 3.3.4.1. We also focused on a threshold value of -100 dB for the log-spectrograms that will be given as inputs to the model (after being normalized between -1 and 1).

Labeled data The labeled data that will be given to the model during the supervised fine-tuning stage are composed of two elements. The first component consists of the signal representation data, i.e. the normalized log-spectrograms exactly as for the unlabeled data, and the second the perceptual labels extracted from the listening test. Each sample of the dataset consists in a sequence of normalized spectral vectors that are given one by one to the model whereas its corresponding label consists in only one single vector of ratings. We thus decided, in order to be consistent, to associate the corresponding label vector to each normalized spectral vector, finally resulting for each samples in $[\mathbf{x}_{i,t};\ \mathbf{l}_i]$ where i corresponds to the index of the sample, $\mathbf{l}_i$ its associated labeled vector (i.e. the mean ratings of the participants of the test on the 8 evidenced perceptual dimensions) and $\mathbf{x}_{i,t}$ the normalized 513-point spectrum of its t^{th} frame.

4.3.3 Regularized AE-based models implementation

For our experiments, we decided to focus on the VAE model with an architecture of $[513, 128, enc, 128, 513]$ with (tanh, linear) pair of activation functions and a classic regularization weighting coefficient $\beta = 1.10^{-6}$ (3.10). As regards with the perceptual regularization weighting coefficient α, we tested several values: 1 and 0.1 as in [Esling et al. 2018a; Esling et al. 2018b] and also 0.01 in order to significantly extend the range of values to evaluate the behavior of the model with respect to the new constraint.

As in Chapter 3, we investigated different values for the encoding dimension, but this time going from 8 to 100, the size of the perceptual space evidenced before being of 8 and thus preventing us from using lower sizes for the bottleneck.

As introduced in Section 4.2.3.1, we also wanted to evaluate the impact of repeating the 2-step procedure several times during the training stage on the results achieved by the model. For that we thus experimented 3 different learning processes: applying only once the 2-step procedure (i.e. pre-training and then fine-tuning), applying this process twice, or repeating it three times.

All the models were implemented in Python using the *Keras* toolkit [Chollet 2015] and the training was performed using the Adam optimizer [Kingma and Ba 2015] with a learning rate of 10^{-3} over 600 epochs with a batch size of 512. For the unsupervised pre-training phase, an early stopping criterion with a patience of 30 epochs was used whereas during the fine-tuning stage the model was forced to train until the end (i.e. the 600 epochs).

4.3.4 Experimental results

In this section we present the results obtained from the experiments we realized aiming at validating objectively our methodology for the perceptual control of a VAE. In other words,

we wanted to investigate if this method still allowed us to generate sounds with good quality while constraining the extracted latent space.

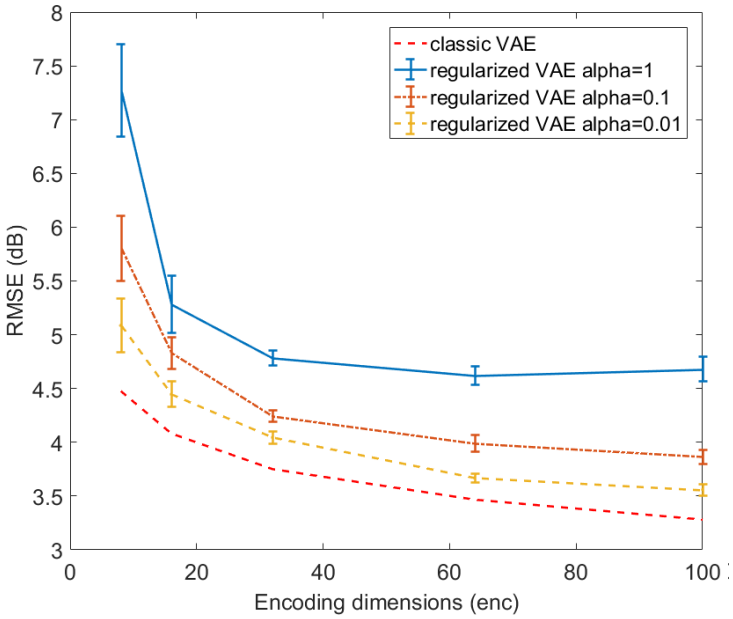
4.3.4.1 Analysis-synthesis

As a first step towards the evaluation of the perceptually-regularized VAE, we compared the different versions of the model (including the non perceptually-regularized model baseline) objectively in an analysis-synthesis framework by computing their RMSE and PEMO-Q scores for different values of the encoding dimension. The aim of this analysis is to evaluate how the addition of the perceptual constraint during training deteriorates the quality of the reconstructed signal.

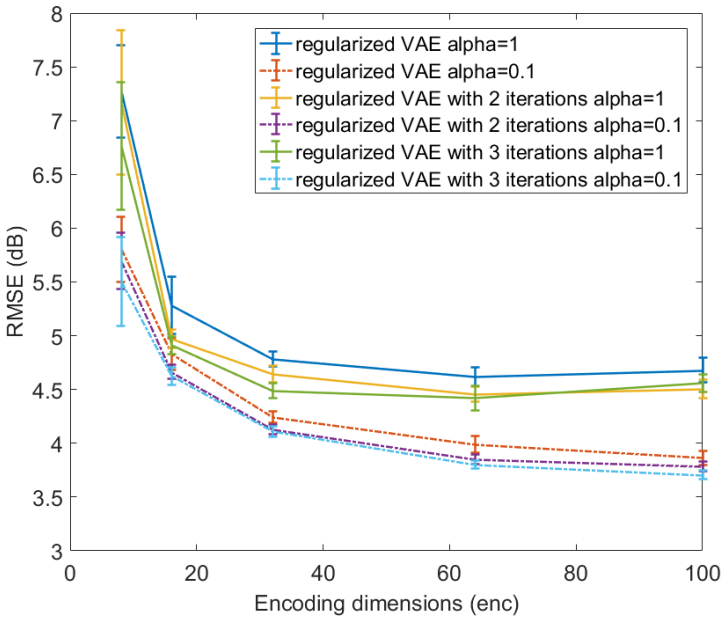
The results are illustrated in Figure 4.2 (RMSE) and Figure 4.3 (PEMO-Q scores). As for the results presented in Section 3.3.4.1, for each considered dimension, a 95% confidence interval of each metric was obtained by conducting a paired t-test considering each sound (i.e. each audio file) of the test set as an independent sample and taking the classic VAE baseline as reference.

From the figures, one of the first observations we can do is that the results reported for RMSE and PEMO-Q present similar behavior and that, as we expected, increasing the weight of the perceptual regularization (i.e. α) deteriorates the quality of the reconstructed signal. In particular, for an encoding dimensions of 8, the regularization highly deteriorates the quality of the signal reaching up to 62% worsening in RMSE (respectively 25% in PEMO-Q score) for an α value of 1. This result is not very surprising as this dimension corresponds to the size of the perceptual space and thus, by fine-tuning, we forced the model to encode constant values for each latent dimensions (corresponding to the given labels on each scale) while the signal we want to reconstruct is not static and evolves with respect to time. Indeed, as the model is fed on a frame-by-frame basis, it means that the dynamics of the input signal are encoded through the dynamics of the latent trajectories, which is not the case if we force them to be constant, affecting thus severely the quality of the output signal. This can also explain the large size of the confidence interval. For other encoding dimensions, especially greater than 16, the results are statistically more accurate and we can observe that an α value going from 0.01 to 1 causes an increase in RMSE of 6% up to 33% (respectively 1.6% to 16% in PEMO-Q) for an $enc = 64$. This shows that the α weighting coefficient can have an important impact on the quality of the reconstructed signals and thus has to be chosen wisely.

Concerning the number of repetitions of the 2-step procedure for training the VAE, from Figure 4.2b and Figure 4.3b we can see that, even if performing a second iteration of the process seems to significantly improve the quality of the reconstruction (up to 6% decrease in RMSE for $\alpha = 1$ and $enc = 16$ and only 3.6% for $enc = 64$, respectively 2.8% and 4.2% in PEMO-Q), it seems that there is no interest in performing it again as the increase in accuracy is not very compelling. Moreover, the gap in either RMSE or PEMO-Q that it creates is clearly below the gain in accuracy brought by using a lower α value (which can reach up to 17% improvement in RMSE for $enc = 100$, and almost 14% for $enc = 64$, respectively around 10% for PEMO-Q). Qualitatively however, the repetition of the learning procedure seems to have an impact on the perceptual regularization. From informal listening test, we noticed that the more repetitions of the 2-step procedure we performed to train the model,

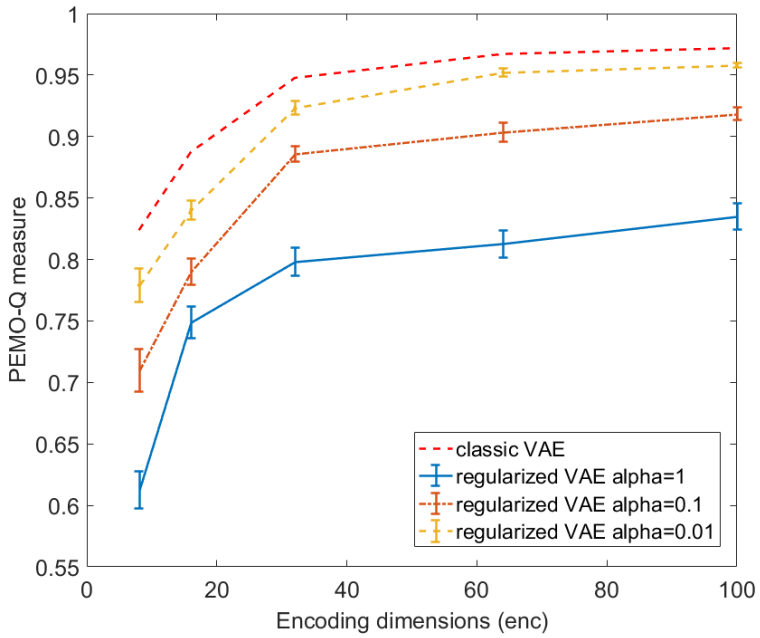


(a) Test of perceptually-regularized VAE with different values of α and comparison with the classic VAE baseline.

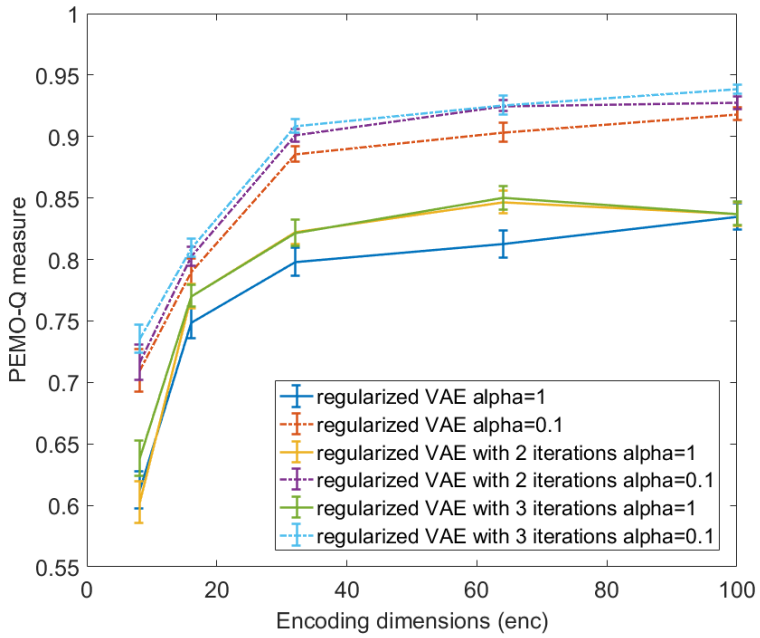


(b) Test of perceptually-regularized VAE for different numbers of iterations of the 2-step procedure, and for two values of α.

Figure 4.2: Reconstruction error (RMSE in dB) obtained with different versions of the perceptually-regularized VAE.



(a) Test of perceptually-regularized VAE with different values of α and comparison with the classic VAE baseline.



(b) Test of perceptually-regularized VAE for different numbers of iterations of the 2-step procedure, and for two values of α.

Figure 4.3: PEMO-Q measures obtained with different versions of the perceptually-regularized VAE.

the less influence there was on the regularization (lower perceptual influence of the dimensions for the generation of sounds when applying an offset on the latent trajectory of a particular sound[3]). This point had thus to be investigated further, see next sections.

A final remark we can make about these results is that, in terms of both RMSE and PEMO-Q for the perceptually-regularized models, we are slightly below the quality that has been achieved with the purely unsupervised models. However, the results are close to the ones obtained on the NSynth dataset for VAEs with a β value between 5.10^{-6} and 2.10^{-5} (see Figure 3.8b and Figure 3.9b) which makes the model suitable, to a certain extent, for our application if the parameters are wisely chosen.

4.3.4.2 Latent space organization

Now that we have evaluated the proposed model in terms of reconstruction accuracy, a first step towards the evaluation of the regularization effectiveness for fine-tuning the latent space is to investigate if the structure of the new latent space has been modified. To do so, we focused on the perceptually-regularized model with an encoding dimension of 64 and an α of 0.1 (as it appeared to achieve a nice tradeoff between the regularization and the reconstruction accuracy). Then we computed the latent vectors of all the samples of the annotated dataset and applied the t-SNE algorithm [van der Maaten and Hinton 2008] to obtain a 2-dimensional projection of its 8 first latent dimensions. We applied the exact same method on the classic VAE model in order to compare the structures of their respective latent spaces. Also, so as to investigate further the impact of the repetition of the 2-step procedure during the training, we computed the t-SNE projection of the perceptually-regularized models for 2 and 3 iterations of the process. The results are illustrated in Figure 4.4.

On Figure 4.4, each point on the graph represents one frame of one of the samples of the annotated dataset that has been encoded. For the sake of clarity, we displayed the projections of the latent spaces obtained for the different models with only one label (i.e. the color) that we chose to represent the ratings on the "Métallique" scale[4]. Importantly, these colors are only indicative and serve the purpose of highlighting a grouping of the samples that is clearly strengthened in the perceptually-regularized models compared to the classic VAE[5]. From the figure, we can also note a difference between the different number of repetition of the training procedure. Indeed, for only one iteration of the procedure, the newly extracted latent space seems to be very well-structured in groups that are well-defined and separated from others whereas the more repetitions, the less obvious the grouping structure (although it is difficult to distinguish a clear difference between the models with 2 or 3 iterations).

From these experiments, we have confirmed that the additional regularization term does have a clear impact on the structure of the latent space extracted by the model and that this impact seems to be reduced when the 2-step procedure is repeated during the training phase. However, from the results it is impossible to say if the perceptually-regularized VAE is able

[3]See Section 4.4.1 for more details about this process. Some sound examples are also available at the companion webpage.

[4]Of course, this could have been possible with every scale from the test. Similar results were observed.

[5]It is worth noting that on the left columns, the 3 rows present slightly different projections. This is only due to the fact that the t-SNE algorithm is stochastic and that the results are therefore different each time it is computed, as it can be seen in the figure.

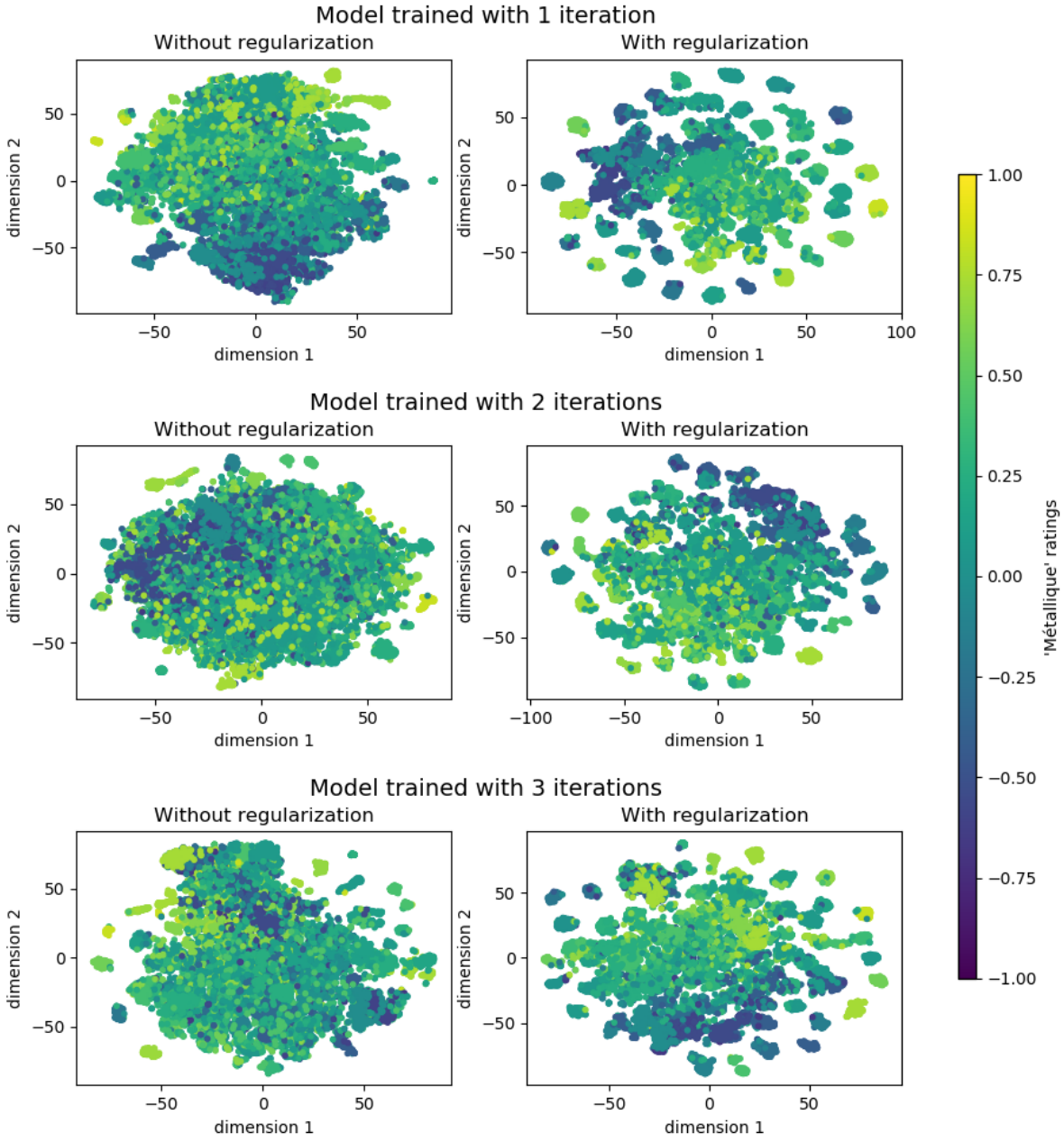


Figure 4.4: 2-dimensional projection of the 8 first dimensions of the latent space encoded by the VAE models using the t-SNE algorithm on the labeled dataset. On the left column are presented the projections corresponding to the classic VAE and the right column the perceptually-regularized VAE. From the top to the bottom are represented the models trained with respectively 1, 2 and 3 iterations of the 2-step learning procedure. On this example the color represents the mean ratings given by the participants on the "Métallique" scale.

to generalize or if it has only been overfitted. Also, these results do not inform us on whether the modifications in the structure of its latent space are indeed perceptually relevant or not. In order to tackle these issues, we conducted a preliminary perceptual evaluation of the model

by means of a new listening test that will be presented in the next section.

4.4 Perceptual evaluation of regularized AE-based models

In order to assess the performance of the perceptually-constrained VAE, we conducted a perceptual study based on the A/B testing model where the participants had to select from pairs of sounds the one that maximized one perceptual dimension in an analysis-modification-synthesis framework. In this section, we detail this listening test and the obtained results.

4.4.1 Stimuli

Selecting the perceptual scales to be evaluated As previously explained, the model we perceptually regularized is a static model, i.e. the model does not encode any temporal aspect of the input signals. Hence, there were few chances that the model could correctly capture the perceptual dimensions related to temporal concepts such as percussiveness, resonance or evolution of the signals. We thus decided to remove these scales from the evaluation of the model (i.e. *percussif*, *qui résonne* and *qui évolue*, respectively "percussive", "resonating" and "evolving"). Only five scales were then selected for the perceptual test: *métallique*, *chaud*, *soufflé*, *qui vibre* and *agressif* (respectively "metallic", "warm", "breathy", "vibrating" and "aggressive").

Selecting the source samples for each scale By means of this study, there were two different criteria that we wanted to validate in order to assess the effectiveness of the model. The first was to verify that the model was indeed able to learn the main perceptual dimensions from the annotated dataset and thus capable of modifying sound timbre along these dimensions. The second was to evaluate if the model had overfitted and was therefore only capable to handle correctly the labeled samples or if it could generalize and have the same behavior with respect to samples that it had never seen before. To do so, we selected source samples to be modified by the model from both the annotated dataset (training) and the test set.

Moreover, for each of the five selected scales, we wanted to evaluate the model on sounds that were very representative of the scale and sounds that were very unrepresentative of the scale (i.e. being able to turn a metallic sound into an even more metallic sound and also transform a sound that is not metallic at all into a sound that is more metallic for example). We thus selected 6 samples from the annotated dataset that presented the most consensual ratings among the participants of the perceptual test with scales (i.e. with the lowest standard deviation, see Section 2.3.4.4): 3 unrepresentative (with the lowest mean ratings on the scale) and 3 very representative (with the highest mean ratings on the given scale). Unnanotated samples were simply chosen randomly since we did not have much information about their positioning on the perceptual dimensions.

Synthesizing the stimuli Once the source samples for each scale had been selected, we had to establish a process to modify them in order to be suitable for the test (paired samples

both had to be reconstructed by the model and one had to be more representative of the perceptual dimension than the other). To that goal we decided to apply an analysis-modification-synthesis process where the modification consisted in applying a predefined constant offset to the trajectory of the corresponding latent coefficient (i.e. the corresponding entry of $\mathbf{z}$) encoded by the model for each sample. In other words, each sample spectrogram was split into frames that were encoded into a sequence of latent coefficients and for each encoded frame we took the latent coefficient corresponding to the perceptual scale to be evaluated and applied an offset to it. From the results obtained with the previous experiments, see Section 4.3.4, we decided to use the perceptually-regularized VAE model with an architecture of $[513, 128, 64, 128, 513]$ with $\beta = 1.10^{-6}$, $\alpha = 0.1$ and only one iteration of the 2-step learning procedure. The new modified latent sequence was then sent to the decoder. And after the reconstruction of the phase spectrogram using the G&L algorithm [Griffin and Lim 1984], the final waveform was reconstructed using the ISTFT with OLA. We chose this offset to be positive so as to stay consistent with the perceptual test with scales as the task was to evaluate the amount of "metallicity" or "aggressiveness" in the sounds, which is intrinsically positive. However, we could have tried negative offset values in order to see if the model could also transform the sounds the other way around and generate less representative samples.

All the transformations (analysis-synthesis, modification of $\mathbf{z}$ and G&L algorithm) brought a lot of artifacts in the reconstructed samples. In order to be as unbiased as possible, we thus applied the same transformation process (with different offset values) to both samples of the pairs to be compared during the test. The two different offset values to apply were defined empirically. The lowest offset was set to the smallest value for which we could notice a perceptual difference between a signal simply encoded-decoded and a signal reconstructed with an offset on its latent trajectory. Regarding the largest offset, we searched for the threshold value for which all the reconstructed signals were perceptually identical (somehow saturating the decoder) and set the offset value to be 50% of this threshold. Some examples of analyzed-transformed-synthesized samples using different offset values are available in the companion webpage.

Normalizing the stimuli in loudness Finally, once all the stimuli were generated, we had to normalize them in loudness, see explanations in Section 2.2.2.1. To do so, we normalized the perceived loudness in combination with a true-peak level measurement following the international standards ITU BS.1770-4 [International Telecommunication Union (ITU) 2015] and EBU R 128 [European Broadcasting Union (EBU) 2016] that have been developed to standardize loudness measurements based on the power of the audio signal. In particular, we followed the recommendation of the Audio Engineering Society for short form content which consists in fixing the integrated loudness at a value of -16 LUFS and set the maximum true-peak value to -1 dB TP [Audio Engineering Society (AES) 2017]. The normalization was performed in Python by using the *pyloudnorm* package [Steinmetz 2018] that implements algorithms following these norms.

4.4.2 Protocol of the perceptual study

For this final perceptual test, we focused on a very simple study based on the A/B testing model as it was faster to implement, quick and easy for the participants to perform and it

allowed us to have a simple evaluation of the model effectiveness with regards to the given task (i.e. having extracted perceptually meaningful dimensions). The basic principle of such a test is to present two different sound samples to the participants and ask them to select the sample that best matches a given criterion. By presenting two versions of a same sample modified through the perceptually-regularized model, one presumed (according to the imposed latent coefficient) more *metallic* or *aggressive* than the other, we can have a preliminary perceptual validation of the fact that the model can modify significantly a source sound in a perceptually relevant manner or not. This would be a first step towards evaluating whether it was able to capture, to a certain extent or not, the corresponding acoustic characteristics of the perceptual dimension of interest and thus fulfill the objectives of our study, at least for some of the evidenced perceptual dimensions.

In our case, after a short introduction of the task, the participants were successively presented with pairs of samples corresponding to two different versions of a same original sound sample modified by the perceptually-regularized VAE in an analysis-modification-synthesis framework using two different offset values on the corresponding dimension, see Section 4.4.1 fore more details. They were asked (in French) to select which one of the two samples sounded the most "metallic" for example, or on the 4 other dimensions ("warm", "breathy", "vibrating" and "aggressive"). The samples position (on the left – A – or on the right – B) was set randomly from one pair to another. Then, for each pair comparison, if the sample corresponding indeed to the greater VAE offset had been selected by the participant, his or her response y was set to "1" in our results table and to "0" otherwise (binary variable).

In total, the participants had 60 pairs to evaluate. The test lasted about 20 minutes with a comparable duration to our two preceding experiments (see Chapter 2). A screenshot of the interface of the test is given in Figure 4.5 and the entire protocol is detailed in Appendix E.1.

For each of the 30 participants that took the test, the pairs of samples and their corresponding question were presented in a different random order. Also, for each pair, the association of the two sounds to sample A or sample B was random so that no bias linked to a systematic order presentation was introduced.

4.4.3 Results analysis

In this section we will first present how we analyzed the perceptual data and then the obtained final results.

4.4.3.1 Applied methodology

For this test, we introduced sounds generated by means of an analysis-transformation-synthesis using the perceptually-regularized VAE applied on both samples from the training dataset and samples that the model had never seen before. The purpose of this analysis was thus two-fold. The main goal was to evaluate if participants were able to "correctly" perceive variations in VAE offset. In other words, if the "guessing" probability (i.e. $y = 1$ as explained in Section 4.4.2) is significantly greater than chance. The second objective was to investigate the influence of the type of the dataset factor (train or test) on the participants evaluations so as to assess whether the model is able to generalize or not. If a significant interaction was found between these factors, then a post-hoc analysis was conducted to evaluate the

Figure 4.5: Screenshot of the A/B perceptual test. Similarly as before, this test was implemented using the Web Audio Evaluation Tool [Jillings et al. 2015]. *Note: Here the B sound has been selected.*

"guessing" performances/scores for each separate dataset.

As for the previous perceptual test with semantic scales, we decided to perform all the analyses scale-wise and thus to evaluate the collected answers for each verbal descriptor separately.

The different steps involved in the analysis will be detailed in the following paragraphs.

Logistic random effects regression The first step was to choose the appropriate method to apply for the statistical analysis of the results together with its different parameters. Given the fact that for each scale a participant was asked to answer the same question for several sounds (6 samples from each dataset), we considered the "participant" variable into our model as a random effect. Likewise, as a same sample has been presented to every participant that took the test, we also considered a variable "sample" as a random effect in the model. Then, since we were dealing with a binary variable y and repeated measures, we chose to use a logistic random effects regression to analyze the results. The logistic regression was performed using the *glmer* function of the *lme4* R software package[6].

Nested models test Once the model established, as explained in the introduction of this section, we wanted to evaluate if the "dataset" factor had a significative impact on the binary variable y. To do so, we used a likelihood-ratio test and applied the *anova* function of the R software.

[6] https://CRAN.R-project.org/web/packages/lme4/index.html

AUC – ROC curve In order to evaluate the accuracy of the final model, we computed the area under the curve (AUC) of the ROC curve (receiver operating characteristic) that takes values between 0 and 1. The greater the AUC, the more accurate the model [Zweig and Campbell 1993]. To compute this value we used the *AUC* package of the R software[7].

Chance threshold comparison Finally, depending on the final chosen model, we had to perform one or two tests to compare the probability of $y = 1$ with chance. For that, we applied the methodology presented in [Hothorn et al. 2008] and used the *glht* function of the R software *multcomp* package. In the case where the "dataset" factor has a significative impact and that we have thus to perform two comparisons, this method insures that the risk of the first type related to the simultaneous decision of all decisions does not exceed a threshold that we set ourselves in advance (0.05) by adjusting the p-values. For the analysis, we performed one-tailed hypothesis test ($H_0 : p(y = 1) \leq 0.5$ vs. $H_1 : p(y = 1) > 0.5$).

4.4.3.2 Results

The obtained results are illustrated in Figure 4.6 and the results of the different tests introduced above are reported in Table 4.1.

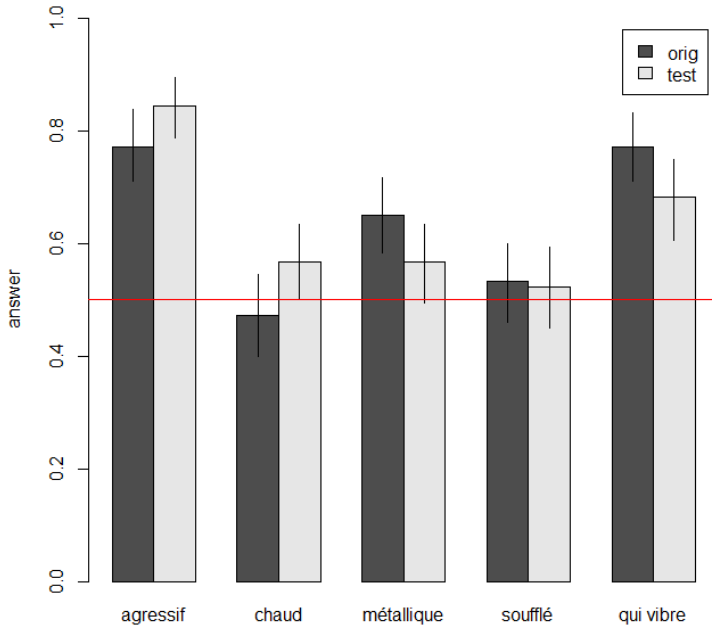


Figure 4.6: Results of the A/B testing analysis for the different scales and the 2 datasets (*orig* corresponds to the training annotated dataset and *test* to the test samples). The red line indicates chance. For each scale and the two datasets, the mean answer is illustrated with a 95% confidence interval.

From both Figure 4.6 and Table 4.1 we can make some observations. A first observation is that for every scale, the chosen model is well-adapted and present a satisfactory AUC value (from 0.83 to 0.91).

[7] https://CRAN.R-project.org/package=AUC

Scale	"dataset" factor test		Chance threshold comparison		ROC curve
	$Chisq(1)$	p-value	z	p-value	AUC
Agressif	0.026	0.871	3.388	$\ll 0.0001$	0.91
Chaud	1.07	0.30	0.37	0.355	0.84
Métallique	0.434	0.61	1.47	0.07	0.89
Soufflé	0.002	0.964	0.25	0.40	0.91
Qui vibre	0.76	0.383	3.93	$\ll 0.0001$	0.83

Table 4.1: Table of the results of the analysis for the different scales. The second column represents the results of the *anova* test aiming at evidencing a potential impact of the "dataset" factor on the answers (the *Chisq(1)* and p-value are reported), the third column presents the results of the comparison with chance (z and p-value) and the last column presents the results of the AUC – ROC curve measures.

Then, for all the scales, we can see that the "dataset" factor has no significative impact on the results which means that the perceptually-regularized model is able to generalize and change the acoustic characteristics of a sounds even if it has not been trained on it (p-values > 0.05). Given this result, the comparison with the chance threshold for all the scales was performed by taking into account the samples independently of the dataset they originated from.

Finally, regarding the comparison with chance, we can distinguish three different cases. The first case concerns the *agressif* and *qui vibre* dimensions. The test showed that the guessing score was significantly greater than chance (with a p-value $\ll 0.0001$), which means that the model clearly captured the underlying acoustic characteristics of these perceptual dimensions. Surprisingly, *qui vibre* was one of the least consensual dimensions when looking at the results of the previous perceptual test (see Section 2.3.4.4) but it appeared from this test that participants have nonetheless identified correctly and agreed on the concept modeled by the VAE for this scale. A possible reason could be that, in the group of participants selected for labeling the sounds, the ratings on this scale were very consensual and focused on one particular conception of *qui vibre* that was well-captured by the model and acknowledged by the participants of this final A/B perceptual test. Regarding the *agressif* dimension, the results support the idea that participants strongly agree on the underlying acoustic properties and conception of the term, and that the VAE preformed well to model them. Conversely, the results cannot reject the hypothesis according to which the participants answered the test randomly for the *chaud* and *soufflé* dimensions (p-value > 0.05). There can be several reasons for that. One possible explanation is that the model could not capture correctly the acoustic properties of these descriptors, possibly because the ratings were not consistent or consensual enough. This could also be explained by the fact that, as results of the SD study showed a decent agreement among participant on these scales but with some variations of conceptions, when taking the A/B test the participants focused on different aspects of the timbre and thus did not agree on the responses. Finally, concerning the last dimension (*métallique*), the graph

shows that a tendency emerged towards a guessing score greater than chance although that tendency is not statistically significant (p-value of $0.07 > 0.05$).

4.4.4 Discussion

The results presented in the previous section must be taken as preliminary results. Indeed, the test that has been designed to evaluate the model is very simple and limited. The participants cannot express themselves about the selected verbal descriptors (e.g. explaining that there is a difference between the two presented samples but that the question raised is not appropriate to describe this variation) nor quantify the difference between the two samples by using quantitative values for example. To do a more rigorous evaluation of the model, it would be necessary to increase the size of the dataset to evaluate, the number of participants and to perform a more precise test. For instance it could be a test similar to our second scaling study and using the same original stimuli, but this time complemented by slightly modified samples using the perceptually-regularized VAE. In particular, it could be interesting to investigate further the *métallique* dimension as the results obtained here are encouraging.

4.5 Conclusions and perspectives

In this chapter, we presented our proposed semi-supervised method with a 2-step learning procedure aiming at perceptually regularize a VAE. This learning procedure consists in first pre-training the model in an unsupervised way and then, by adding an extra perceptual constraint to the function to be optimized, fine-tuning the model on a labeled dataset. To evaluate our proposed perceptually-regularized VAE we then performed two sets of experiments. First we realized an objective comparison of several configurations of our model, including the classic VAE as the baseline. To do so, we compared them by evaluating both the reconstruction accuracy they could achieve using the RMSE objective physical measure and the PEMO-Q objective perceptual measure as in Chapter 3, and the structure of their extracted latent space using a 2-dimensional t-SNE projection of the latent coefficients corresponding to the samples of the training dataset. Then, we selected the best model parameters setting (corresponding to the optimal tradeoff between reconstruction accuracy and perceptual regularization) and evaluated perceptually the effectiveness of the model at capturing the acoustic characteristics of the perceptual dimensions by performing a listening test. The model being static (i.e. not having recurrent layers to represent the dynamics of the signal), we only kept the perceptual scales that were not related to the temporal dynamics of the signals (thus 5 out of 8): *métallique*, *chaud*, *soufflé*, *qui vibre* and *agressif*. We conducted a perceptual study using an A/B testing model where the participants had to select from pairs of sounds the one that maximized one perceptual dimension in an analysis-modification-synthesis framework.

From these experiments, several conclusions can be drawn. First, as expected from the results of Section 3.3.4.1, we noticed that the additional regularization degrades slightly the quality of the audio signals generated by the model but that this quality remains decent when choosing an appropriate weighting factor α. This is an issue that could be easily tackled in the future by enlarging the size of the datasets (both for the unlabeled dataset used for the unsupervised pre-training of the model $\mathcal{X}_U$ and the labeled dataset used during the supervised

fine-tuning $\mathcal{X}_L$) as we discussed in Section 3.3.4.1. We also noticed that repeating the 2-step procedure when training the model slightly improved the accuracy of the reconstructed signal. However, this improvement was done at the expense of the perceptual regularization of the latent space and the benefit from iterating the training steps appeared less compelling than finding a well-suited α value. The interest of such process was thus not clearly evidenced. Finally, thanks to the last perceptual test, we managed to perform a preliminary evaluation of the effectiveness of the model perceptual regularization. This experiment allowed us to validate our proposed methodology. The model was able to generalize and to capture the acoustic properties of some of the given perceptual descriptors although the size of the labeled dataset is very small. In particular, the results demonstrated that the model managed to capture very well the acoustic properties of the *agressif* and *qui vibre* dimensions. However, the results on *chaud* and *soufflé* dimensions showed that the perceptually-regularized VAE could not model correctly these dimensions while the results on *métallique* are not so clear and deserve further investigation. In order to have a more robust analysis of the effectiveness of the model at capturing the characteristics of all the evidenced perceptual descriptors and obtain clearer results, it could be interesting to realize a more complex test (e.g. an other SD study) and at a bigger scale, i.e. involving both more participants and more sound samples, but this is left to future work.

Conclusion and perspectives

Conclusion

During this thesis, we were interested in one of the main challenges of the synthesizer market and the research in sound synthesis which is to suggest new forms of synthesis allowing the creation of new sonorities while offering musicians new more intuitive controls to help them reach the desired sound more easily. The main objective of this PhD work was thus to develop and evaluate new data-driven machine learning methods for music sound synthesis allowing the generation of new sounds while providing high-level perceptually meaningful control parameters. As a first step towards this challenging objective, we got interested into synthesizing new timbres by modifying sounds according to such perceptual controls. For example, it can consist in making a specific sample sounds more "metallic" and less "aggressive".

To achieve this objective, we had to tackle three main challenges. The first was to determine what were the most salient perceptual dimensions and meaningful verbal descriptors used to characterize synthesizer sounds. Then, the second challenge consisted in finding a well-suited machine learning model able to perform a mapping between a high-level representation space presenting interesting interpolation and extrapolation properties and the sound space so that we can navigate smoothly while exploring timbres beyond the limits of the database and create new interesting sounds with high quality. Finally, the last challenge was to make sure the main dimensions of this representation space made sense perceptually by encouraging them to match the verbal descriptors evidenced during the first step.

In the first chapter, we presented the state-of-the-art in music sounds perception and synthesis and all the concepts necessary for the rest of the thesis. We introduced the main methods that have been applied to understand and characterize musical timbre and the commonly used (and commercialized) synthesis techniques. We also presented new synthesis techniques based on data-driven machine learning algorithms and in particular methods based on deep learning which is the current favored framework for sound synthesis.

In the second chapter, we explored the most frequently and consensually used terms to describe synthesizer sounds. To that purpose we conducted two perceptual tests. The first one was a free verbalization test whose aim was to collect vocabulary used to describe synthetic stimuli we generated using Arturia's instruments. From this study, after grouping the collected terms into perceptually relevant categories and selecting the most frequent and consensual ones, we identified 8 perceptual dimensions: *métallique*, *chaud*, *soufflé*, *qui vibre*, *percussif*, *qui résonne*, *qui évolue* and *agressif* (English closest terms: "metallic", "warm", "breathy", "vibrating", "percussive", "resonating", "evolving" and "aggressive"). Then, we performed a second perceptual test aiming at rating stimuli along these 8 dimensions. The main objectives of this test were (i) to evaluate the degree of consensus of the selected dimensions in order to assess whether they could be adapted for control, and (ii) to get a quantitative evaluation

of a subset of our Arturia dataset of samples using these scales (in order to weakly supervise the training of a VAE for resynthesis).

In chapter 3, we faced the second challenge and investigated the use of machine learning methods to extract a relevant representation space with interesting properties. Following the previous studies using data-driven deep learning models for sound synthesis, we decided to focus on autoencoders (AEs) and realized an extensive comparative study of several types of these models (linear, shallow, deep, recurrent and variational) on two different datasets. The comparison was performed by evaluating the different models using several criteria. First, we compared the different models in terms of the reconstruction accuracy they could reach in an analysis-synthesis framework by computing both RMSE and PEMO-Q scores for several sizes of the bottleneck layer. Then we compared the organization of their latent space by performing a correlation analysis on the extracted latent dimensions, and finally investigated the use of this space for sound morphing by performing interpolations between pairs of sounds. From these experiments, together with a qualitative analysis made with a non real-time prototype developed during the thesis, we have evidenced that these models, and in particular variational autoencoders (VAEs), are relevant tools for extracting a high-level latent space in which we can navigate smoothly and create new sounds. However, so far, no link between the latent dimensions extracted by these models and the perceptual dimensions evidenced by the perceptual tests was made possible.

In the last chapter, we focused on the last challenge and tried to add supervision to the model by encouraging some of the extracted latent dimensions to be perceptually relevant and to match the 8 evidenced perceptual dimensions. Inspired by different studies and given the size of our annotated dataset, we proposed a semi-supervised method based on a 2-step learning procedure to perceptually regularize a VAE model. This method consisted in first applying an unsupervised pre-training of the VAE using the complete Arturia dataset (just as in Chapter 3) and then, by adding a perceptual regularization term to the variational lower bound to be optimized, performing a supervised fine-tuning of the model on the labeled subset of synthesizer sounds (created using the ratings collected during the second perceptual test). We then evaluated our method both objectively and perceptually. First, we performed an objective evaluation of our method by comparing the classic VAE model with several perceptually-regularized VAEs presenting different parameter settings (encoding dimensions, weighting factors for the extra perceptual regularization, repetition of the learning procedure). This comparative study was performed by evaluating two different criteria: the reconstruction accuracy obtained by the different models in an analysis-synthesis framework (both in terms of RMSE and PEMO-Q scores), and the structure of their latent space by projecting them in a 2-dimensional space using the t-SNE algorithm. Then, after choosing the best tradeoff parameters setting for the perceptually-regularized VAE (with regards to both perceptual regularization and reconstruction accuracy), we closed the loop by performing a perceptual evaluation of our proposed method using a final listening test. This perceptual test enabled us to evaluate the effectiveness of the perceptual regularization using 5 of the 8 scales that were not related to temporal dynamics of the signal (the model being static). From these preliminary experiments, we could validate our methodology as appropriate for generating sounds with fair-to-good audio quality and capturing the acoustic properties of (two plus one to be investigated further) perceptual dimensions while generalizing its behavior onto never

seen samples in an analysis-transformation-synthesis framework.

Main contributions

We can summarize the main contributions of this thesis as:

1. We listed 784 verbal descriptors used by expert and non expert listeners to describe (in French) synthesizers sounds (more than half of them being classical terms used to describe the timbre of other sound categories, and less than half of them being new terms that appear to be specific to synthesizer sounds).

2. Among them, we identified 8 perceptual verbal descriptors that are both frequently, transversally and consensually used to describe synthesizer sounds and evaluated their degree of consensus (both intra-subject and inter-subject).

3. We obtained quantitative evaluations of a subset of 80 synthesizer sounds with respect to these 8 perceptual dimensions, resulting in an annotated dataset available for weakly supervised algorithms.

4. We conducted an extensive comparison of AE-based models for synthesis and sound morphing using two different samples datasets (a standardized multi-pitch and multi-instrument dataset and a dataset of mono-pitch synthesizer sounds) and concluded that these models are well adapted for navigating smoothly a timbre space while allowing to create new sonorities with good quality. Plus, thanks to the organization of its latent space, we showed that the VAE model seems to be particularly well adapted for our application.

5. We suggested a new methodology to perceptually regularize a VAE model using weak supervision learning. This 2-step learning semi-supervised method seems to be well adapted for allowing the model to generalize although a very small set of annotated samples has been used. Thanks to a simple preliminary perceptual test, we could validate to a certain extent that our methodology allowed the model to capture the acoustic properties of some perceptual dimensions and to transform sounds in a perceptually relevant manner with respect to these dimensions.

Perspectives

In the next sections, we will propose some improvements and suggestions we thought of for future investigations on this topic.

Perceptual dimensions analysis

During this thesis, we collected a large number of terms to describe synthesizer sounds that we grouped into perceptually meaningful categories using a semantic proximity analysis. It would be a good idea to study these groups further by performing correlation or redundancy analysis to evaluate whether the selected descriptors are "independent" and genuinely

appropriate for being used as control parameters, or if there are redundant and thus not very adapted for such application.

From the collected terms, it could also be interesting to explore the semantic network related to the specialized lexicon used for the perceptual characterization of the timbre (in the case of synthesizer sounds) and possibly investigate the use of sound-informed word embeddings [Lopopolo and Miltenburg 2015; Vijayakumar et al. 2017] to go deeper in the analysis of the semantic relationships between the terms of this lexicon.

In addition, in order to study deeper these selected categories, it could be interesting to investigate the underlying acoustic correlates of these perceptual dimensions so as to better understand their acoustic properties/characteristics and the impact on the sound. It would also be great to be able to compare these acoustic correlates to the ones obtained for different types of musical instruments such as orchestral instruments [Zacharakis et al. 2014], the operatic voice [Garnier et al. 2007] or the speaking voice [Ehrette 2004].

Neural models for audio synthesis

For this work, in order to extract an interesting representation space, we focused on AE-based models and in particular on VAE models that we regularized perceptually. In the view of future investigations, some improvements can be brought to both the chosen VAE model and the general framework.

Future developments on the VAE model

First, as stated in Section 3.2.3.2, the fixed-variance free-mean Gaussian assumption on the spectrogram magnitude coefficients is not the best probabilistic model. One interesting improvement to be brought would thus be to modify the actual minimized loss function (MSE) for the IS divergence whose minimization corresponds to the optimization of the parameters of the model in the maximum-likelihood (ML) sense under the assumption of a circularly symmetric complex Gaussian model for the STFT complex coefficients which is a better statistical model than the current Gaussian model for power magnitude coefficients.

There also exists another recently proposed version of VAE called the Wasserstein AE (WAE) [Tolstikhin et al. 2018] which we did not have time to explore during this work. The WAE uses a different method from the classic VAE for regularizing the latent space by relying on the maximum mean discrepancy (MMD) instead of the KL divergence. This model has shown to outperform the standard VAE on image generation by synthesizing less blurry images. It could thus be interesting to investigate the use of this model for our application.

Additionally, as explained in Section 4.2.3.2, we did not investigate the relationship between the use of the semantic scales and the actual perception of potential users and consequently chose naively to implement the perceptual extra regularization as a MSE between the latent coefficients of the VAE and the mean ratings of the participants collected during the SD test. But this relation is probably not linear and there exist several metrics that would possibly be more adapted to the application. Another perspective for the improvement of our model would be to consider different distance metrics for the extra regularization and evaluate their influence on the results.

Finally, it could be interesting to investigate, as we did for the AE, a recurrent version

of the VAE such as a RNN-VAE as proposed in [Chung et al. 2015] for instance. Indeed, although the latent space extracted by this model seemed to be very complex and correlated[8], in the experiments we conducted on a dataset with an adapted size (i.e. for NSynth dataset), see Chapter 3.3.4.1, the LSTM-AE significantly outperformed all other models in terms of reconstruction accuracy which makes it a good candidate to explore. Moreover, the presence of recurrent layers would allow us to evaluate the perceptually-regularized model also on the perceptual dimensions related to temporal dynamics of the signal, which would be of great interest for this study. However, in RNN-VAE, the two internal "states" (the deterministic state $\mathbf{h}$ of the RNN and the stochastic state of the VAE) are intertwined and it is thus not that easy to be able to encode separately the dynamics of the signal and the higher level dependencies (e.g. emotions in speech) into these two variables [Chung et al. 2015].

Future developments on the general framework

Concerning the general framework applied for this work, some improvements can also be brought to future investigations.

To begin with, as explained in Section 3.3.4.1, a first interesting step to validate our model would be to largely increase the size of our Arturia dataset in order to overcome the issue of the unbalance between the number of input vectors and the number of parameters to train in the model.

Then, when transforming the sounds by means of our perceptually-regularized VAE models, the reconstructed samples suffered from some audible artifacts that probably come from the algorithm used to reconstruct the phase spectrogram from the magnitude spectrogram (G&L algorithm and/or linear phase unwrapping). A nice perspective for future work would therefore be to investigate other phase reconstruction methods, and possibly explore some real-time robust techniques in order to implement a real-time prototype embedding our model.

Finally, as an alternative to our proposed perceptually-regularized VAE model, it could be interesting to investigate the use of conditional GANs (C-GANs) as these models are state-of-the-art in image generation and seem to show very promising results for high-quality sound synthesis [Donahue et al. 2019; Engel et al. 2019]. However, these models do not give explicit parameters such as VAEs and may not be very adapted to an analysis-transformation-synthesis framework as investigated in this PhD work, but more appropriate for the exploration of pure synthesis from perceptually relevant control parameters.

[8] Which could perhaps be overcome to some extent by the regularization constraint of the VAE model.

APPENDICES

Appendix A

Free verbalization test additional material

A.1 Participants

A.1.1 Pre-test questionnaire

- Comment évalueriez-vous votre niveau en français ? *Une seule réponse possible.*
  - Langue maternelle
  - Bilingue
  - Courant
  - Bon niveau
  - Niveau élémentaire

- Veuillez sélectionner les champs qui vous concernent. *Plusieurs choix possibles.*
  - Pratique d'un instrument de musique
  - Enregistrement et mixage audio
  - Développement de logiciels audio
  - Recherches dans le domaine de l'audio

- A quelle catégorie d'âge appartenez-vous ? *Une seule réponse possible.*
  - Moins de 15
  - 15-24
  - 25-34
  - 35-44
  - 45-54
  - 55-64
  - 65+

- Pour ce test, il est vivement conseillé de se mettre dans de bonnes conditions d'écoute (environnement calme et matériel audio correct - ne pas utiliser les enceintes d'un ordinateur portable). Durant le test pour pourrez régler le volume du son en utilisant le curseur "Master Volume Control" en haut à gauche de la page. Veuillez sélectionner le

champs ci-dessous se rapprochant le plus de vos conditions d'écoute. *Une seule réponse possible.*

- Enceintes de studio
- Enceintes domestiques
- Casque de bonne qualité
- Casque de qualité moyenne/basse
- Autres

A.1.2 Collected information

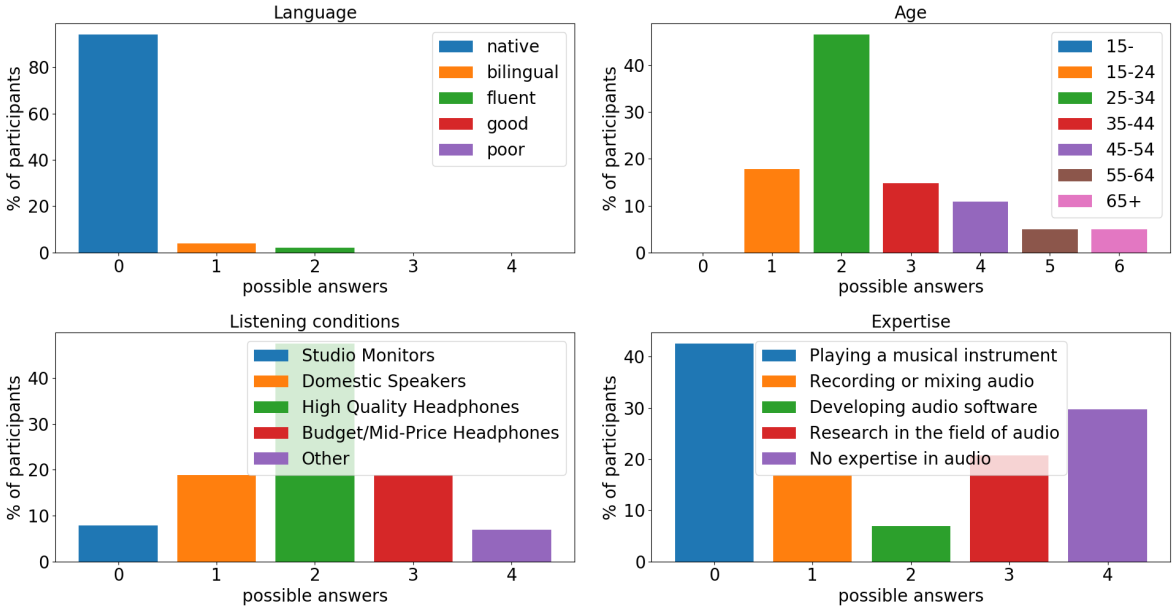


Figure A.1: Repartition of the questionnaire answers collected during the free verbalization test in percentage of the 101 participants. The different possible answers are given as labels of the graphs.

A.2 Detailed protocol explanations

- ▷ Pour ce test vous devrez écouter l'échantillon audio et remplir les cases vides avec les mots qui vous semblent le mieux décrire le son écouté. Veuillez vous concentrer autant que possible sur des termes descriptifs et éviter les termes relatifs à des jugements de valeurs comme "beau" ou "déplaisant". Par exemple, si vous deviez décrire une couleur donnée, vous pourriez dire "chaude", "vive" ou encore "cuivrée".
- ▷ Vous pourrez écouter l'échantillon autant de fois que vous le souhaitez sans aucune limitation mais il vous faudra le jouer intégralement au moins une fois.
- ▷ Il est obligatoire pour ce test de remplir au moins le premier mot mais nous vous encourageons à en mettre le plus possible sur les 5 champs proposés.
- ▷ Une fois que vous êtes satisfait de vos réponses, vous pouvez cliquer sur le bouton "Suivant" et réitérer l'expérience avec le son suivant. Durant le test, 20 sons à annoter vous seront présentés.
- ▷ Merci beaucoup pour votre participation ! Bon test !

A.3 Manual terms grouping

Table A.1: Groups of terms sharing the same lexical root for the free verbalization test.

Groups of terms
metallique/metal/metallique inox
doux/douceur
resonnant/resonne/qui resonne/resonance
vibrant/vibre/vibration
echo/echos
gresillant/gresillements/gresillement
cuivre/cuivree
souffle/souffler/soufflant
rond/ronde/arrondi
bruite/bruit/bruit blanc/bruit electrique
aigu/aigue
bourdonnant/bourdon/bourdonnement/bourdonner/bourdonne
spatial/espace
oscillant/oscillations
sirene/sirene bateau/sirene d'usine
distordu/distortion
reverberation/reverbere/reverbed/reveberation
robotique/robot
rebondissant/rebond/rebondit
variation/variable/variant/varie
continu/son continu
flute/flute de pan
vrombissant/vrombissement
abeille/abeilles
ondulant/ondulation/ondulatoire/ondule
boise/bois
ouvrant/qui s'ouvre/s'ouvre
saw/scie/dent de scie
didgeridoo/didjeridoo
insecte/insectes
retro futur/retro-futuriste
tube/tubes

Continued on next page

Table A.1 – *Continued from previous page*

Groups of terms
apaisant/apaisante
aspirateur/aspirant
bond/bondissant
bulle/bulles
cinema/cinematographique
explosif/explosion
frottement/frotte
grand/grandeur
insistance/insistant
moine/moines
plainte/plaintif
relaxant/relaxant/relaxation
saccade/saccadee
science-fiction/sf/science-fictionnel
tremblant/tremblotant

A.4 Final perceptual dimensions

Table A.2: Table of perceptual classes for the free verbalization test.

Free Verbalization Test		
Grouped Terms	Occurrences	Transversality
['doux', 'resonnant', 'sourd', 'rond', 'etouffe']	143	49
['metallique', 'froid', 'aigu']	114	44
['agressif', 'electrique', 'desagreable', 'strident']	97	40
['chaud', 'grave', 'profond', 'sombre']	83	37
['vibrant', 'cuivre', 'bourdonnant', 'vrombissant', 'abeilles', 'essaim', 'insectes']	82	47
['synthetique', 'simple', 'distordu', 'guitare']	81	35
['evolutif', 'double', 'percutant', 'voix', 'changeant', 'cosmique', 'didgeridoo', 'sifflement']	59	29
['klaxon', 'plat', 'continu', 'industriel', 'pauvre', 'vieux', 'dur', 'cor', 'stop']	58	30
['percussif', 'cloche', 'corde', 'rebondissant']	47	26
['long', 'orgue', 'eglise', 'tremblant']	46	21
['gresillant', 'bruite', 'brouille']	43	21
['bleu', 'aere', 'ouvert', 'repetitif', 'annonce', 'ovni', 'teleportation', 'tube', 'vaisseau spatial', '90s', 'aspirant', 'disco', 'double son', 'fun', 'gai', 'multicouche', 'realise', 'science-fiction', 'tonique']	41	26
['nasillard', 'pince', 'criard', 'clavecin', 'haut', 'comique', 'enigmatique', 'flangery', 'fronde', 'glacial', 'martelle', 'ouverture porte', 'serre', 'tombant']	41	24
['futuriste', 'spatial', 'laser', 'descendant']	40	22
['electronique', 'riche', 'harmonique', 'vivant']	38	13
['echo', 'reverberation', 'delay']	38	22
['chaleureux', 'organique', 'liquide', 'aquatique', 'clavier', 'sobre', 'flou', 'coeurs', 'contraste', 'discontinu', 'marin', 'mini', 'nape', 'sourdine', 'sous-marin', 'synthe', 'vieillot']	37	23

Continued on next page

Table A.2 – *Continued from previous page*

Free Verbalization Test		
Grouped Terms	Occurrences	Transversality
['neutre', 'piquant', 'tiede', 'buzz', 'pad', 'court circuit', 'douleur', 'inhabituel', 'langage', 'plastique', 'point', 'ponctuel']	37	24
['progressif', 'variation', 'aigre', 'incomplet', 'tuba', 'constipation', 'demi tour', 'maritime', 'ohm', 'pwm', 'quelconque', 'trompe de bateau']	36	23
['sirene', 'bateau', 'artificiel', 'enferme']	34	18
['alarme', 'lourd', 'alarmant', 'fin', 'bouche', 'lineaire', 'adsl', 'alarme sous marin', 'bryant', 'buzzer', 'electricite', 'insistance', 'pesant', 'poussif', 'preventif']	34	24
['avertissement', 'irritant', 'sature', 'saw', 'ample', 'paquebot', 'rauque', 'granulaire', 'grognement', "bruit d'une machine dans une usine ou atelier", 'guimbarde', 'gutural', 'mecanique', 'repos', 'saccade']	31	22
['instable', 'complexe', 'erreur', 'claquant', 'corde cassee', 'relachement', 'attaquant', 'detonant', 'eclair', 'impertinent', 'precipite', 'rate', 'structure', 'trebuche']	31	20
['souffle', 'flute']	31	20
['monotone', 'intense', 'crescendo', 'oppressant', 'anche', 'appareil', 'bruit electrique', 'inharmonique', 'intrigue', 'larsen', 'mesure', 'pose', 'scanner', 'sifflant', 'stridant', 'transporteur']	30	25
['brillant', 'grincant', 'stress', 'suspense', 'crissant', 'casse', 'cristallin', 'detone', 'emiette', 'incivif', 'insupportable', 'mortel', 'presse', 'torture']	30	18
['battement', 'venteux', 'battant', 'bruit', 'enveloppant', 'hache', 'medical', 'multiple', 'sable', 'apaisant', 'baton de pluie', 'blanc', 'clarinette', 'fuyant', 'hybride', 'japonais', 'montee', 'tuyau']	29	31
['calme', 'feutre', 'pur', 'assourdi', 'goutte', 'zen', 'beau', 'bref', 'descriptif', 'detente', 'discret', 'experience musicale', 'meditatif', 'relaxant', 'rythme', 'yoga']	29	21
['sale', 'angoissant', 'dissonant', 'abeille', 'module', 'amateur', 'detune', 'disharmonique', 'foisonnant', 'militaire', 'vibes']	28	21

Continued on next page

Table A.2 – *Continued from previous page*

Free Verbalization Test		
Grouped Terms	Occurrences	Transversality
['nasal', 'frappe', 'percant', 'brusque', 'desaccorde', 'mince', 'xylophone', 'bong', 'corde grincante', 'corde pincee', 'en cours', 'interferences', 'pinche', 'reconfortant', 'ringing', 'test']	27	20
['oscillant', 'robotique', 'alternatif', 'crispant', 'charge', 'chewing gum', 'pompier', 'retro-game', 'transe']	27	17
['inquietant', 'puissant', 'granuleux']	24	16
['rugueux', 'stressant', 'suspens', 'detroit', 'malsain', 'moto', 'paranoia', 'passagere', 'peur', 'seche', 'terreur']	23	15
['canard', 'acid', 'corne', 'elastique', 'apparition', 'atypique', 'bondissant', 'bulle', 'chute', 'epique', 'exotic', 'explosif', 'hipster', 'ouinnn', 'surprenant']	23	18
['caverneux', 'humain', 'choeur', 'acapella', 'accompagnement', 'cathedrale', 'chorale', 'creuse', 'formantique', 'harmonie', 'moine', 'oooooo', 'synthe-cheap', 'vocale']	23	12
['fort', 'sonnette', 'usine', 'accentue', 'alarme rassemblement', 'bleu clair', 'bzzzzzzz', 'eraille', 'etire', 'fondu', 'terne', 'timbre']	22	16
['sec', 'piano', 'electro', 'appel']	22	14
['effrayant', 'trompette', 'allonge', 'animal', 'biton', 'chevre', 'debut', 'deconcertant', 'etrange', 'gemissement', 'harpe', 'mourant', 'plainte', 'quasi-humain', 'son experimental', 'transition', 'triste', 'vagabond']	22	14
['gras', 'alerte', 'machine', 'aiguise', 'beaucommeunklaxon', 'dark', 'klaxon lol', 'mal aux oreilles', 'negatif', 'probleme']	21	16
['agacant', 'amusant', 'bond', 'cowboy', 'dansant', 'enrobe', 'expressif', 'insecte', 'mobile', 'mouvant', 'pneumatique', 'rapide', 'replique', 'retourne', 'rigolo', 'sinusoide', 'tramplin', 'valve (coeur)', 'virevoltant', 'volume', 'wahwah']	21	16
['vif', 'naturel', 'propre', 'attenuant', 'corde piano', 'deprimant', 'essai', 'fizzy', 'moilleux', 'nylon', 'pas fini', 'pincant', 'uniforme']	21	15
['filtre', 'corne de brume', 'mou', 'ambiant', 'lent']	20	11

Continued on next page

Table A.2 – *Continued from previous page*

Free Verbalization Test		
Grouped Terms	Occurrences	Transversality
['diffus', 'analogique', 'plein', 'sursaut', 'bienveillant', 'bruit sourd', 'depart', 'direct', 'new wave', 'pimpan', 'poing', 'referme']	20	14
['acide', 'extraterrestre', 'irregulier', 'ressort']	19	16
['vocal', 'chante', 'leger', 'chantant']	19	12
['synthetiseur', 'ouvrant', 'angoisse', 'compose', 'prolonge']	19	11
['vague', 'croissant', 'accroche', 'caracteriel', 'cinema', 'decollage', 'deplacement', 'fantastique', 'harmonieux', 'intro', 'jingle', 'normal', 'prenant', 'scintillant']	19	14
['ondulant', 'phase', 'amplifiant', 'bouillant', 'bouillonant', 'bweeeeengggggg', 'desastreux', 'flux', 'galaxique', 'metalo-feuille', 'pateux', 'sweep', 'tempete', 'vomi']	18	13
['avion', 'radio', 'aerien', 'arrivee', 'bi-sonore', 'biphase', 'deux phases', 'eau synthetique', 'espiegle', 'noir', 'panne', 'rocailleux', 'souvenirs', 'spirituel', 'tracteur', 'wooooong']	18	13
['diffusant', 'feu', 'accord', 'action', 'coherent', 'complet', 'credible', 'depassement', 'diffusion', 'dance', 'edm', 'festif', 'impressionnant', 'orange mecanique', 'pechue', 'route']	18	15
['imposant', 'carre', 'accord cordes', 'apaisante', 'austere', 'bas', 'instrument a vent', 'magnetique', 'moelleux', 'serieux', 'tesla', 'tondeuse']	17	12
['chuintant', 'ordinateur', 'achevement', 'agerable', 'allumage', 'bravo', 'delai', 'etoile', 'fusee', 'futuristique', 'introductif', 'melodique', 'qui se ferme', 'starwars....', 'tioooon-piouuu']	17	11
['interrompu', 'rassurant', 'cornemuse', 'acordeon', 'analogue', 'brass', 'cargo', 'dent-sieux', 'melodieux', 'urgence']	16	10
['ethere', 'religieux', 'voyelle', 'complexe', 'envoutant', 'euuuuh', 'expire', 'femme synthetique', 'granule', 'ooh']	16	14
['classique', 'basique', 'interphone', 'signal', 'ebauche', 'interface', 'moustique', 'perdu', 'peteux']	16	12

Continued on next page

Table A.2 – *Continued from previous page*

Free Verbalization Test		
Grouped Terms	Occurrences	Transversality
['consistant', 'blurred-noisy', 'cigale', 'experience de son', 'grand', 'large', 'massif', 'mystique', 'proche voix tibet', 'propage', 'reflechissant', 'reve profond', 'sonar']	16	13
['brut', 'statique', 'rude', 'perceuse']	15	5
['gong', 'boise', 'majestueux']	15	11
['court', 'basse']	15	10
['net', 'ambiance', 'attente', 'base', 'claire', 'corde pince', 'fff', 'montagne', 'repetition', 'tintement', 'violet']	14	11
['experience', 'expiration longue', 'fragile', 'frelons', 'fremissant', 'frottement', 'las', 'mouche', 'tenu', 'unitone']	14	10
['air', 'aspirateur', 'avancer', 'drone', 'eau', 'friction', 'glissant', 'grattant', 'guitare electrique', 'karcher', 'machine tournante', 'mitigeur', 'neige', 'trajet']	14	10
['epais', 'organe', 'affrique', 'gravite', 'tension']	14	6
['stable', 'constant', 'bruyant', 'penible', 'saxophone', 'son trainant', 'sonnerie bateau', 'sonnerie machine']	14	10
['vibrato', 'jaune', 'amen;)', 'blues', 'cheap', 'etrique', 'lisse', 'solennel', "test d'ecoute", 'tintillement']	14	8
['decroissant', 'numerique', 'impactant']	13	3
['vent', 'clair', 'mysterieux']	13	13
['cor de chasse', 'dephase', 'etreint', 'fade', 'interpellant', 'lointain', 'lumineux', 'non-maitrise', 'plaintif', 'pompeux', 'rouge', 'sursautant', 'tintant']	13	10
['camion', 'rapeux', 'bus', 'danger', 'fluctuant', 'insistant', 'machine irm', 'retro', 'tremolo']	12	10
['vintage', 'ascendant', 'eletronique', 'entrainant', 'errone', 'gain de niveau dans un jeu', 'in progress', 'monochorde', 'pret', 'saturation', 'soudain']	12	10
['intensifie', 'menacant', 'annonciateur', 'bloque', 'cymbale', 'en boucle', 'guerre', 'missile', 'tournant', 'vigoureux']	12	7
['digital', 'voise']	11	4

Continued on next page

Table A.2 – *Continued from previous page*

Free Verbalization Test		
Grouped Terms	Occurrences	Transversality
['etonnant', 'monstre', 'abyssale', 'comtemporain', 'ronronnant', 'telurique', 'tenebreux', 'tonnerre', 'visceral']	11	10
['deplaisant', 'immediat', 'porte', 'arret brutal']	11	4
['agreable', 'onde', 'chorus', 'vooonng']	10	6
['bruit blanc', 'bug', 'extrapolant', 'faible', 'insonore', 'mao', 'mauvais', 'moche', 'perturbation', 'television']	10	17
['baillant', 'banal', 'cinematographique', 'englobe', 'florissant', 'jean-michel jarre', 'planant', 'rfzrqfrzf', 'voluptueux']	9	7
['bizarre', 'creux', 'alien']	9	7
['fiction', 'nebuleux', 'orange', 'atmospherique']	8	2
['deformation', 'frisson', 'stationnaire']	8	3
['impulsif', 'tranchant']	7	4
['4x4 synthetique', 'bouche-vibrant', 'destabilisant=vibrant', 'flatulent', 'fqfgqgdb', 'muet', 'trombonoscope']	7	6
['choc', 'deraillement', 'faux', 'heure', 'longueur', 'priere', 'sinistre']	7	6
['monocorde', 'mono note']	7	3
['delicat', 'sinus', 'standard']	6	3
['confiance', 'radiant']	6	2
['attaque', 'pique']	6	2
['balancant', 'note', 'tymbale']	6	3
['celeste', 'cri', 'decouverte', 'etendu', 'train', 'wobbly']	6	6
['fluide', 'retro futur']	6	3
['regulier', 'guttural']	5	2
['dynamique', 'rayonnant']	5	2
['stereo', 'intergalactique']	5	2

Continued on next page

Table A.2 – *Continued from previous page*

Free Verbalization Test		
Grouped Terms	Occurrences	Transversality
['brouillard', 'inutile']	4	2
['depart de bateau', 'disgracieux']	4	2
['gros', 'techno']	4	2
['charnu', 'guitarcheesy', 'ponton', 'vert']	4	4

Appendix B

Semantic scale study additional material

B.1 Participants

B.1.1 Pre-test questionnaire

- Comment évalueriez-vous votre niveau en français ? *Une seule réponse possible.*
  - Langue maternelle
  - Bilingue
  - Courant
  - Bon niveau
  - Niveau élémentaire

- Veuillez sélectionner les champs qui vous concernent. *Plusieurs choix possibles.*
  - Pratique d'un instrument de musique
  - Enregistrement et mixage audio
  - Développement de logiciels audio
  - Recherches dans le domaine de l'audio

- A quelle catégorie d'âge appartenez-vous ? *Une seule réponse possible.*
  - Moins de 15
  - 15-24
  - 25-34
  - 35-44
  - 45-54
  - 55-64
  - 65+

- Pour ce test, il est vivement conseillé de se mettre dans de bonnes conditions d'écoute (environnement calme et matériel audio correct - ne pas utiliser les enceintes d'un ordinateur portable). Durant le test pour pourrez régler le volume du son en utilisant le curseur "Master Volume Control" en haut à gauche de la page. Veuillez sélectionner le

champs ci-dessous se rapprochant le plus de vos conditions d'écoute. *Une seule réponse possible.*

- Enceintes de studio
- Enceintes domestiques
- Casque de bonne qualité
- Casque de qualité moyenne/basse
- Autres

- Avez-vous déjà passé le premier test perceptif de verbalisation libre ?
  - Oui
  - Non

B.1.2 Collected information

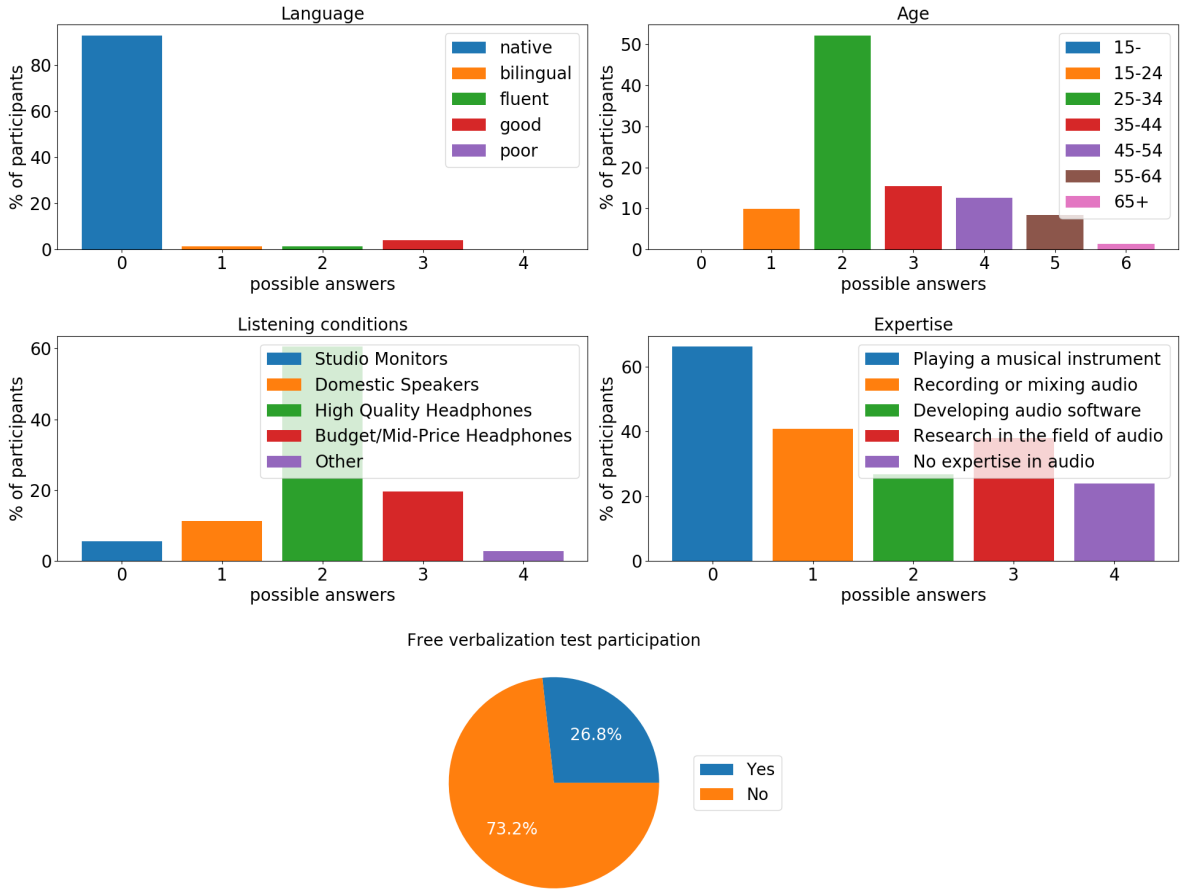


Figure B.1: Repartition of the questionnaire answers collected during the test with semantic scales in percentage of the 71 participants. The different possible answers are given as labels of the graphs.

B.2 Test protocol

- ▷ Pour ce test vous devrez écouter l'échantillon audio et l'évaluer selon les différentes échelles perceptives données.
- ▷ Vous pourrez écouter l'échantillon autant de fois que vous le souhaitez sans aucune limitation mais il vous faudra le jouer intégralement au moins une fois.
- ▷ Une fois que vous êtes satisfait de vos réponses, vous pouvez cliquer sur le bouton "Suivant" et réitérer l'expérience avec le son suivant. Durant le test, 40 sons à évaluer vous seront présentés.
- ▷ Merci beaucoup pour votre participation ! Bon test !

B.3 Results analysis

B.3.1 Intra-subject consensus analysis

The subjects that are kept for the rest of the analysis are the subjects where the intra-subject correlation coefficients are above the chosen threshold of 0.5 (in dotted green line in the figure). The histograms of the consistent subjects are displayed in orange.

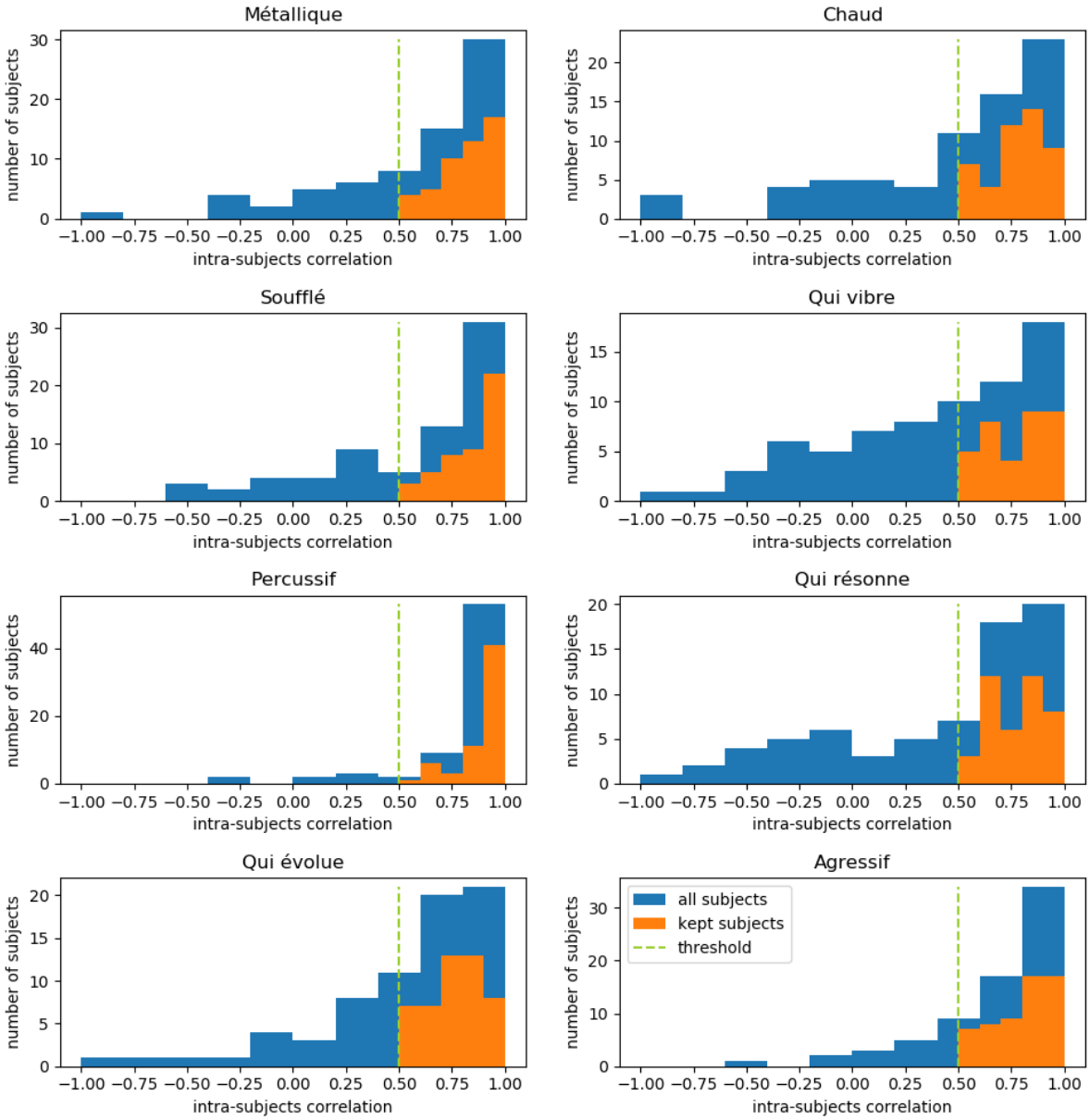


Figure B.2: Thresholding to select reliable subjects from the histogram of the intra-subject correlation coefficients for each semantic scale.

B.3.2 Inter-subject consensus clustering analysis

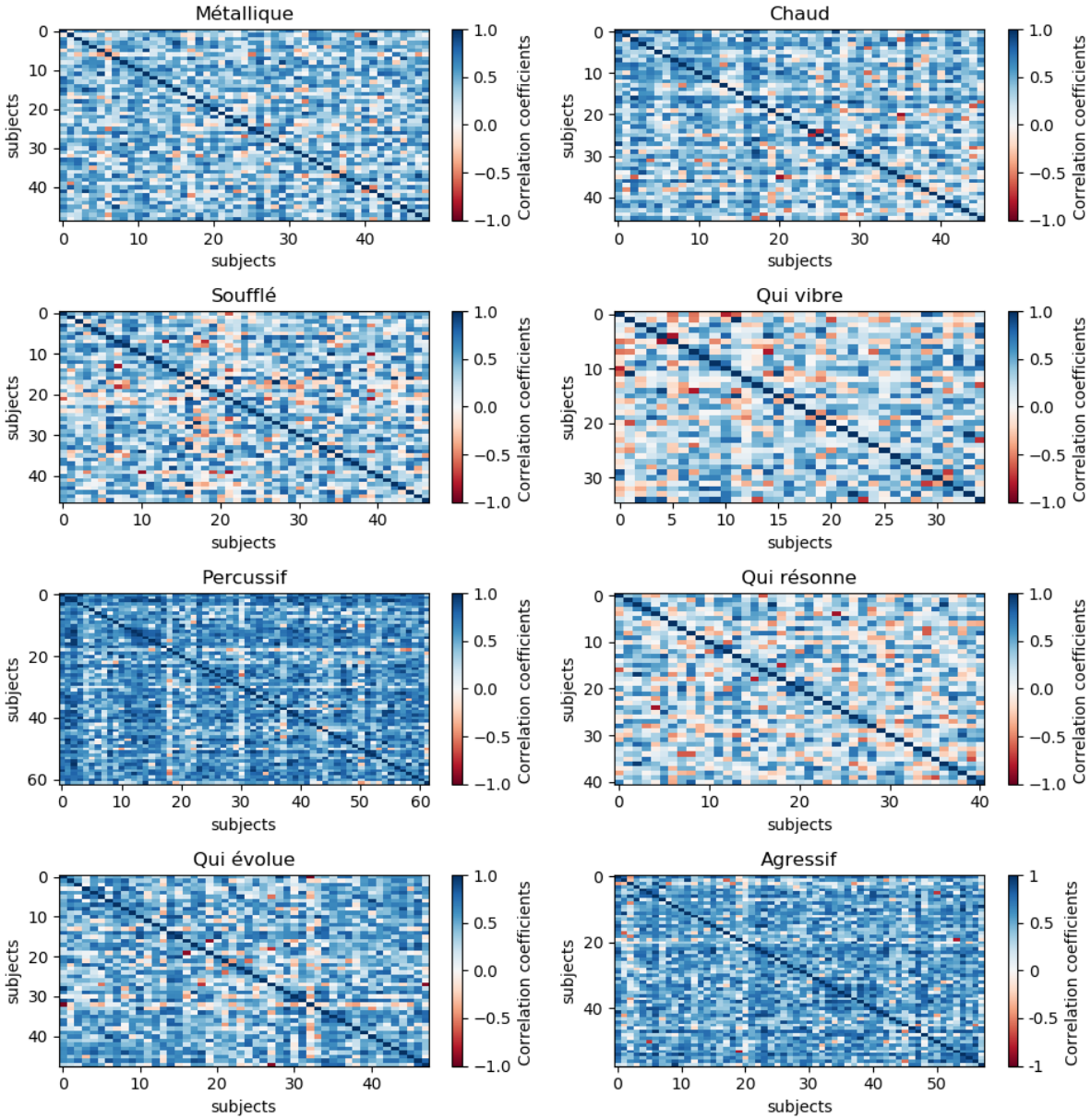


Figure B.3: Inter-subject correlation coefficients matrices given the semantic scales.

Appendix C

Mathematical additional material

C.1 Probability distributions

C.1.1 Gaussian distributions

Let $\mathcal{N}(x; \mu, \sigma^2)$ denote the Gaussian distribution for a random variable $x \in \mathbb{R}$ with mean $\mu \in \mathbb{R}$ and variance $\sigma^2 \in \mathbb{R}_+$. Its probability density function (PDF) is defined by:

$$\mathcal{N}(x; \mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right).$$

Note that for simplicity we use the same notation to denote a probability distribution and its PDF.

Let $\mathcal{N}(\mathbf{x}; \boldsymbol{\mu}, \boldsymbol{\sigma}^2)$ denote the multivariate Gaussian distribution for a real-valued random vector $\mathbf{x} \in \mathbb{R}^F$ of mean vector $\boldsymbol{\mu} \in \mathbb{R}^F$, and with statistically independent entries such that $\boldsymbol{\sigma}^2 \in \mathbb{R}_+^F$ is the vector of variances (covariance terms are zero and thus omitted in the parametrization for simplicity). Its PDF is therefore equal to the product of univariate Gaussian PDFs:

$$\mathcal{N}(\mathbf{x}; \boldsymbol{\mu}, \boldsymbol{\sigma}^2) = \prod_{f=0}^{F-1} \mathcal{N}(x_f; \mu_f, \sigma_f^2),$$

where v_f denotes the f^{th} entry of a vector $\mathbf{v}$.

Let $\mathcal{N}_c(x; \mu, \sigma^2)$ denote the proper complex Gaussian distribution for a random variable $x \in \mathbb{C}$ with mean $\mu \in \mathbb{C}$ and variance $\sigma^2 \in \mathbb{R}_+$. Its PDF is defined by:

$$\mathcal{N}_c(x; \mu, \sigma^2) = \frac{1}{\pi\sigma^2} \exp\left(-\frac{|x-\mu|^2}{\sigma^2}\right).$$

This distribution is circularly symmetric (i.e. invariant to a phase shift for x) if $\mu = 0$

C.1.2 Gamma distribution

Let $\mathcal{G}(x; a, b)$ denote the Gamma distribution for a random variable $x \in \mathbb{R}_+$ with shape and rate parameters $a > 0$ and $b > 0$ respectively. Its PDF is defined by:

$$\mathcal{G}(x; a, b) = \frac{b^a}{\Gamma(a)} x^{a-1} \exp(-bx),$$

where $\Gamma(\cdot)$ is the Gamma function.

C.1.3 Poisson distribution

Let $\mathcal{P}(x;\lambda)$ denote the Poisson distribution for a random variable $x \in \mathbb{N}$ with rate parameter $\lambda > 0$. Its PDF is defined by:

$$\mathcal{P}(x;\lambda) = \exp(-\lambda)\frac{\lambda^x}{x!}.$$

C.2 Mathematical developments for the variational inference

C.2.1 Variational Lower Bound (VLB)

$$\begin{aligned}
\mathrm{D}_{\mathrm{KL}}\Big(q_\phi(\mathbf{z}|\mathbf{x})|p_\theta(\mathbf{z}|\mathbf{x})\Big) &= \sum_{\mathbf{z}} q_\phi(\mathbf{z}|\mathbf{x}) \log \frac{q_\phi(\mathbf{z}|\mathbf{x})}{p_\theta(\mathbf{z}|\mathbf{x})} \\
&= \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}\Big[\log q_\phi(\mathbf{z}|\mathbf{x}) - \log \frac{p_\theta(\mathbf{x}|\mathbf{z})p_\theta(\mathbf{z})}{p_\theta(\mathbf{x})}\Big] \\
&= \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}\big[\log q_\phi(\mathbf{z}|\mathbf{x}) - \log p_\theta(\mathbf{x}|\mathbf{z})p_\theta(\mathbf{z})\big] \;+\; \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}\big[\log p_\theta(\mathbf{x})\big]
\end{aligned}$$

$$\Rightarrow \log p_\theta(\mathbf{x}) = \underbrace{\mathrm{D}_{\mathrm{KL}}\Big(q_\phi(\mathbf{z}|\mathbf{x})|p_\theta(\mathbf{z}|\mathbf{x})\Big)}_{\geqslant 0} + \underbrace{\mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}\big[\log p_\theta(\mathbf{x}|\mathbf{z})p_\theta(\mathbf{z}) - \log q_\phi(\mathbf{z}|\mathbf{x})\big]}_{\mathcal{L}(\phi,\theta,\mathbf{x})}$$

Then we can write:

$$\begin{aligned}
\mathcal{L}(\phi,\theta,\mathbf{x}) &= \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}\Big[\log \frac{p_\theta(\mathbf{x}|\mathbf{z})\; p_\theta(\mathbf{z})}{q_\phi(\mathbf{z}|\mathbf{x})}\Big] \\
&= \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}\big[\log p_\theta(\mathbf{x}|\mathbf{z})\big] \;-\; \sum_{\mathbf{z}} q_\phi(\mathbf{z}|\mathbf{x}) \log \frac{q_\phi(\mathbf{z}|\mathbf{x})}{p_\theta(\mathbf{z})} \\
&= \underbrace{\mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}\big[\log p_\theta(\mathbf{x}|\mathbf{z})\big]}_{\text{reconstruction accuracy}} \;-\; \underbrace{\mathrm{D}_{\mathrm{KL}}\Big(q_\phi(\mathbf{z}|\mathbf{x})|p_\theta(\mathbf{z})\Big)}_{\text{regularization}}
\end{aligned}$$

C.2.2 Differentiability of regularization term

As we have:

$$q_\phi(\mathbf{z}|\mathbf{x}) = \frac{1}{(2\pi)^{L/2}|\tilde{\mathbf{\Sigma}}_\phi(\mathbf{x})|^{1/2}}\; e^{-\frac{(\mathbf{z}-\tilde{\boldsymbol{\mu}}_\phi(\mathbf{x}))^T\tilde{\mathbf{\Sigma}}_\phi(\mathbf{x})^{-1}(\mathbf{z}-\tilde{\boldsymbol{\mu}}_\phi(\mathbf{x}))}{2}}$$

with L the dimension of the latent space, $\tilde{\mathbf{\Sigma}}_\phi(\mathbf{x}) = \mathrm{diag}(\tilde{\boldsymbol{\sigma}}^2_\phi(\mathbf{x}))$ and:

$$p_\theta(\mathbf{z}) = \frac{1}{(2\pi)^{L/2}}\; e^{-\frac{\mathbf{z}^2}{2}}$$

then we can write:

$$
\begin{aligned}
\mathrm{D_{KL}}\Big(q_\phi(\mathbf{z}|\mathbf{x})|p_\theta(\mathbf{z})\Big) &= \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}\left[\log \frac{\frac{1}{(2\pi)^{L/2}|\tilde{\mathbf{\Sigma}}_\phi(\mathbf{x})|^{1/2}}\; e^{-\frac{(\mathbf{z}-\tilde{\boldsymbol{\mu}}_\phi(\mathbf{x}))^T\tilde{\mathbf{\Sigma}}_\phi^{-1}(\mathbf{x})(\mathbf{z}-\tilde{\boldsymbol{\mu}}_\phi(\mathbf{x}))}{2}}}{\frac{1}{(2\pi)^{L/2}}\; e^{-\frac{\mathbf{z}^2}{2}}}\right] \\
&= -\frac{1}{2}\,\mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}\Big[\log|\tilde{\mathbf{\Sigma}}_\phi(\mathbf{x})| \;+\; (\mathbf{z}-\tilde{\boldsymbol{\mu}}_\phi(\mathbf{x}))^T\tilde{\mathbf{\Sigma}}_\phi^{-1}(\mathbf{x})(\mathbf{z}-\tilde{\boldsymbol{\mu}}_\phi(\mathbf{x})) \;-\; \mathbf{z}^2\Big] \\
&= -\frac{1}{2}\,\Bigg(\log|\tilde{\mathbf{\Sigma}}_\phi(\mathbf{x})| \;+\; \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}\Big[\mathrm{tr}\Big((\mathbf{z}-\tilde{\boldsymbol{\mu}}_\phi(\mathbf{x}))^T\tilde{\mathbf{\Sigma}}_\phi^{-1}(\mathbf{x})(\mathbf{z}-\tilde{\boldsymbol{\mu}}_\phi(\mathbf{x}))\Big)\Big] \\
&\qquad\qquad -\; \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}[\mathbf{z}^2]\Bigg) \\
&= -\frac{1}{2}\,\Bigg(\log|\tilde{\mathbf{\Sigma}}_\phi(\mathbf{x})| \;+\; \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}\Big[\mathrm{tr}\Big(\tilde{\mathbf{\Sigma}}_\phi^{-1}(\mathbf{x})(\mathbf{z}-\tilde{\boldsymbol{\mu}}_\phi(\mathbf{x}))(\mathbf{z}-\tilde{\boldsymbol{\mu}}_\phi(\mathbf{x}))^T\Big)\Big] \\
&\qquad\qquad -\; \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}\Big[\sum_{l=1}^{L} z_l^2\Big]\Bigg) \\
&= -\frac{1}{2}\,\Bigg(\log|\tilde{\mathbf{\Sigma}}_\phi(\mathbf{x})| \;+\; \mathrm{tr}\Big(\tilde{\mathbf{\Sigma}}_\phi^{-1}(\mathbf{x})\;\underbrace{\mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}\Big[(\mathbf{z}-\tilde{\boldsymbol{\mu}}_\phi(\mathbf{x}))(\mathbf{z}-\tilde{\boldsymbol{\mu}}_\phi(\mathbf{x}))^T\Big]}_{\tilde{\mathbf{\Sigma}}_\phi(\mathbf{x})}\Big) \\
&\qquad\qquad -\; \sum_{l=1}^{L}\mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}\Big[(z_l-\tilde{\mu}_{\phi,l}(\mathbf{x})+\tilde{\mu}_{\phi,l}(\mathbf{x}))^2\Big]\Bigg) \\
&= -\frac{1}{2}\,\Bigg(\log\prod_{l=1}^{L}\tilde{\sigma}_{\phi,l}(\mathbf{x}) \;+\; \mathrm{tr}(\mathbf{I}_L) \;-\; \sum_{l=1}^{L}\Bigg(\underbrace{\mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}\Big[(z_l-\tilde{\mu}_{\phi,l}(\mathbf{x}))^2\Big]}_{\tilde{\sigma}^2_{\phi,l}(\mathbf{x})} \\
&\qquad\qquad +\; 2\,\tilde{\mu}_{\phi,l}(\mathbf{x})\;\underbrace{\mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}[z_l]}_{\tilde{\mu}_{\phi,l}(\mathbf{x})} \;-\; \tilde{\mu}^2_{\phi,l}(\mathbf{x})\Bigg)\Bigg) \\
&= -\frac{1}{2}\,\Bigg(\sum_{l=1}^{L}\log\,\tilde{\sigma}_{\phi,l}(\mathbf{x}) \;+\; L \;-\; \sum_{l=1}^{L}\big(\tilde{\sigma}^2_{\phi,l}(\mathbf{x})+\tilde{\mu}^2_{\phi,l}(\mathbf{x})\big)\Bigg) \\
&= -\frac{1}{2}\sum_{l=1}^{L}\big(1+\;\log\,\tilde{\sigma}_{\phi,l}(\mathbf{x})-\tilde{\mu}^2_{\phi,l}(\mathbf{x})-\tilde{\sigma}^2_{\phi,l}(\mathbf{x})\big)
\end{aligned}
$$

Appendix D

AE-based models experiments additional material

D.1 Analysis-synthesis experiments on Arturia dataset

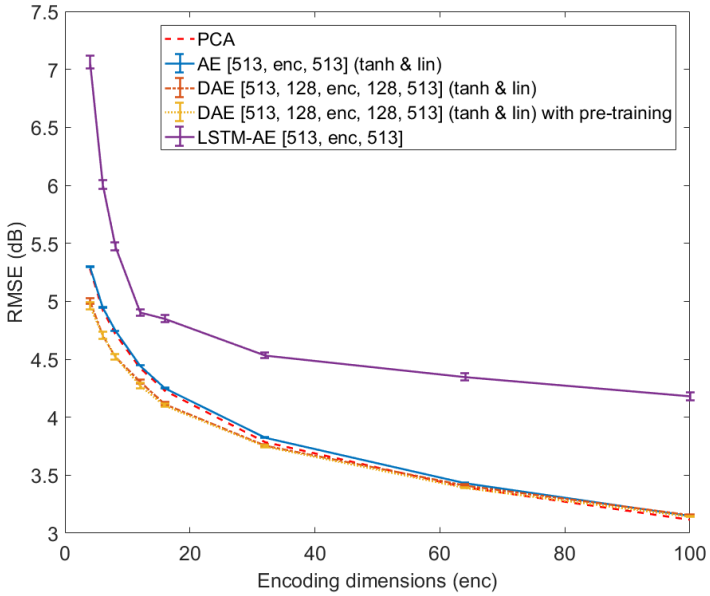


Figure D.1: Reconstruction error (RMSE in dB) obtained with PCA, AE, DAEs (with and without layer-wise training) and LSTM-AE as a function of latent space dimension (trained on the Arturia dataset computed with smaller temporal windows - 1024-point STFT).

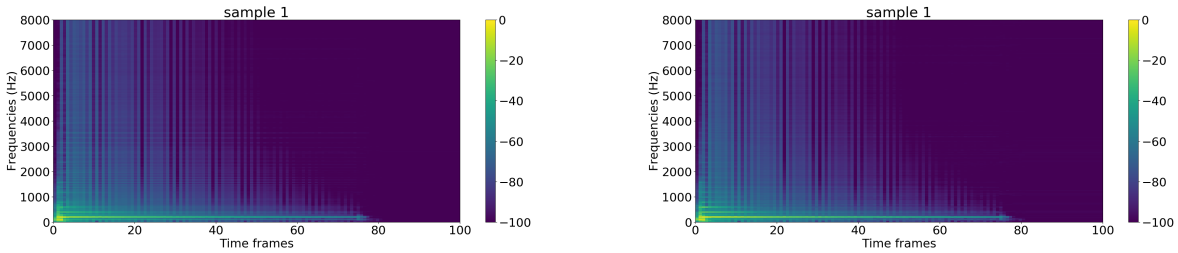
D.2 AE-based sound morphing

D.2.1 NSynth dataset

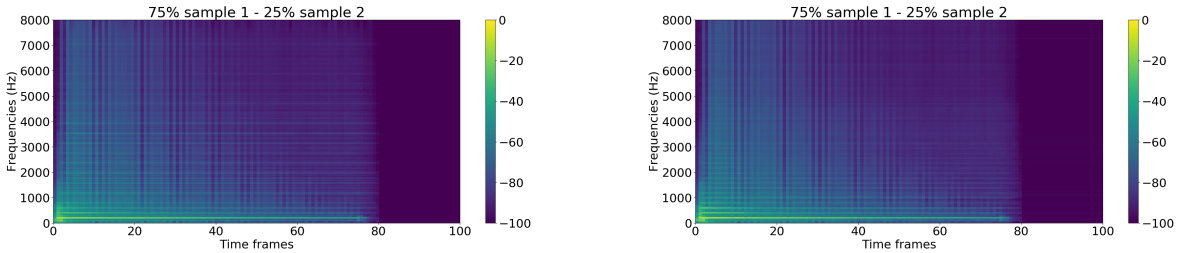
Figure D.2.
Figure D.3.

D.2.2 Arturia dataset

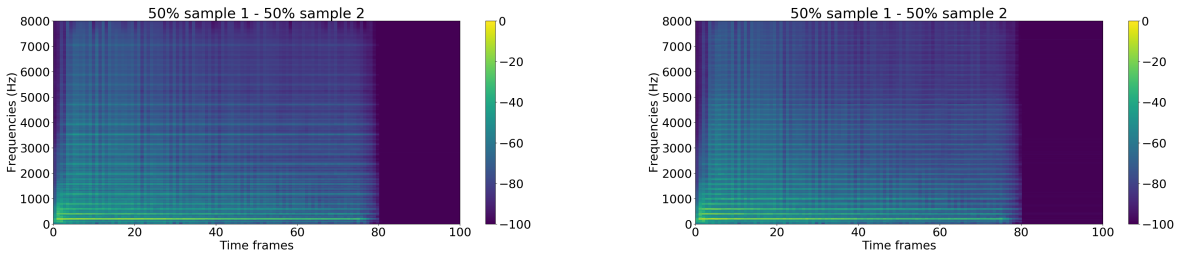
Figure D.4.
Figure D.5.



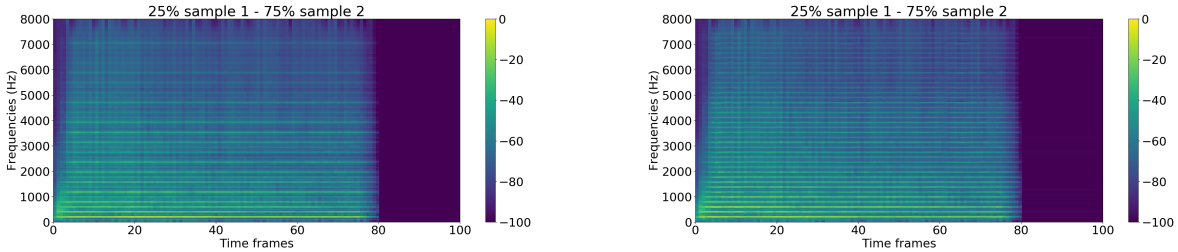
(a) Sample 1



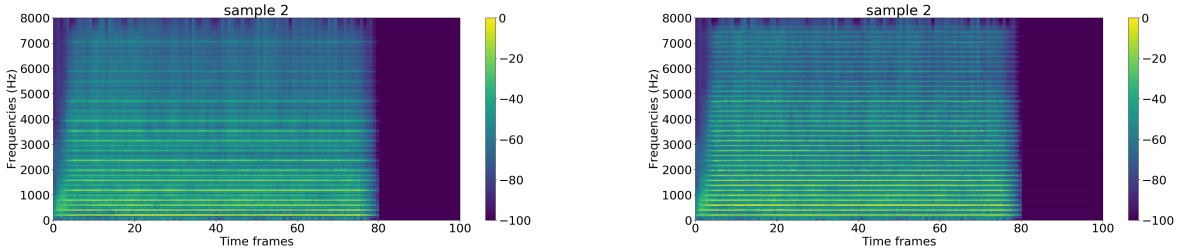
(b) 75% sample 1 – 25% sample 2



(c) 50% sample 1 – 50% sample 2

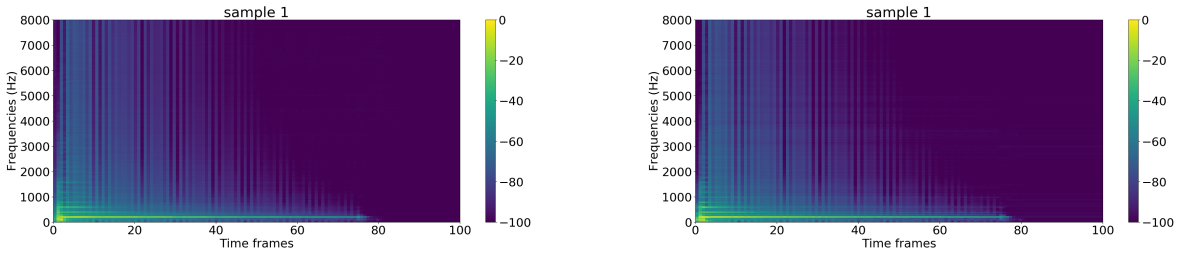


(d) 25% sample 1 – 75% sample 2

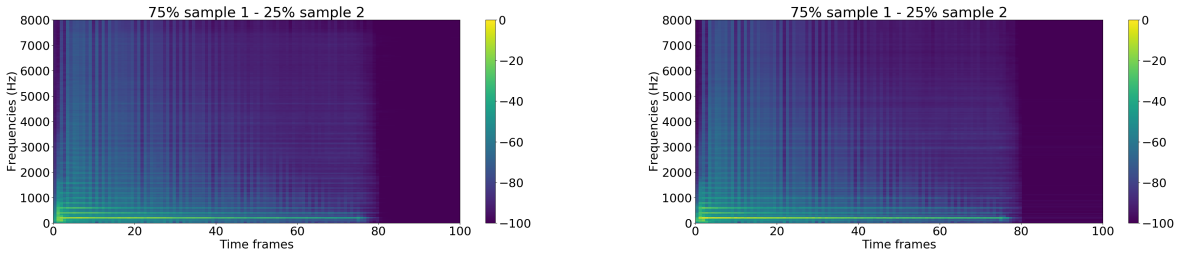


(e) Sample 2

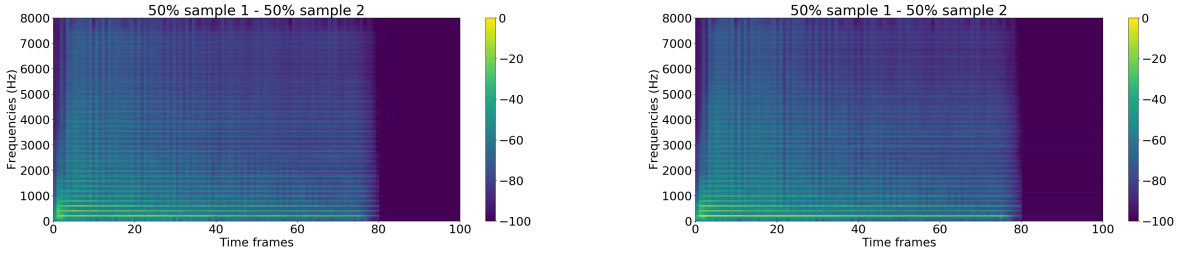
Figure D.2: Examples of decoded magnitude spectrograms after sound interpolation of 2 NSynth samples in the latent space using respectively PCA (left) and DAE (right)



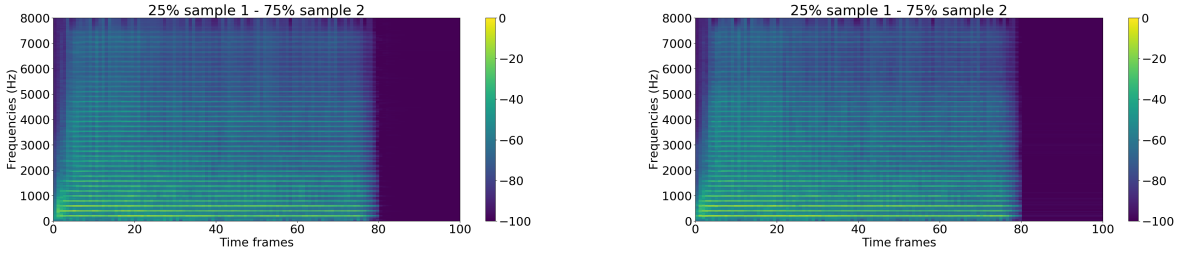
(a) Sample 1



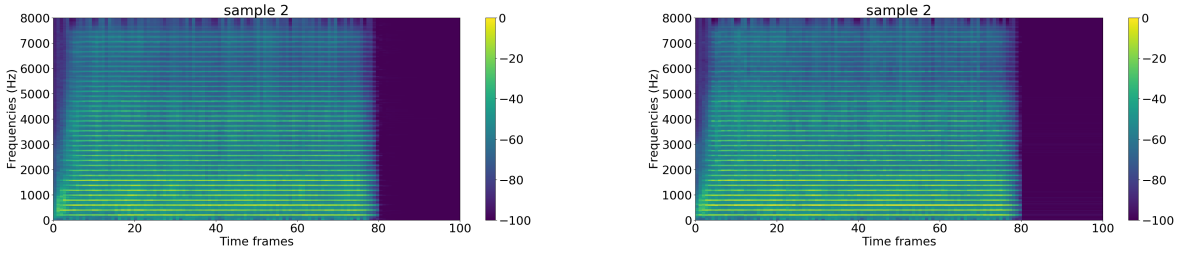
(b) 75% sample 1 – 25% sample 2



(c) 50% sample 1 – 50% sample 2

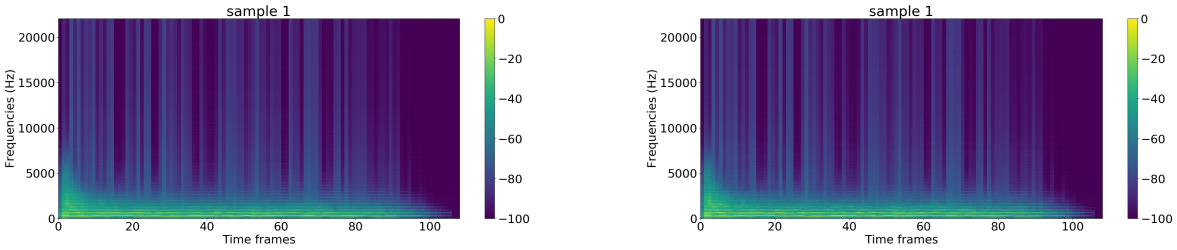


(d) 25% sample 1 – 75% sample 2

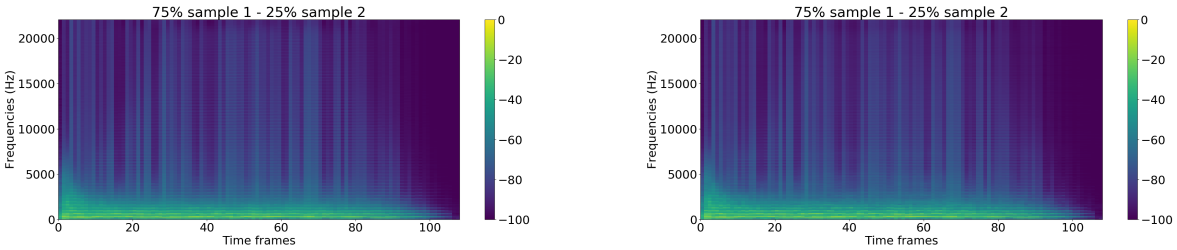


(e) Sample 2

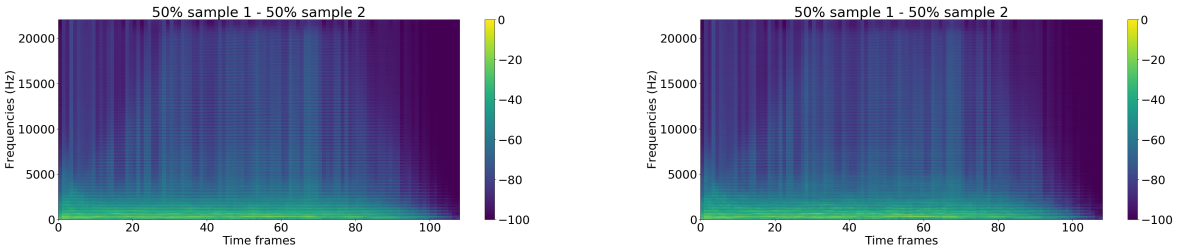
Figure D.3: Examples of decoded magnitude spectrograms after sound interpolation of 2 NSynth samples in the latent space using respectively LSTM-AE (left) and VAE (right).



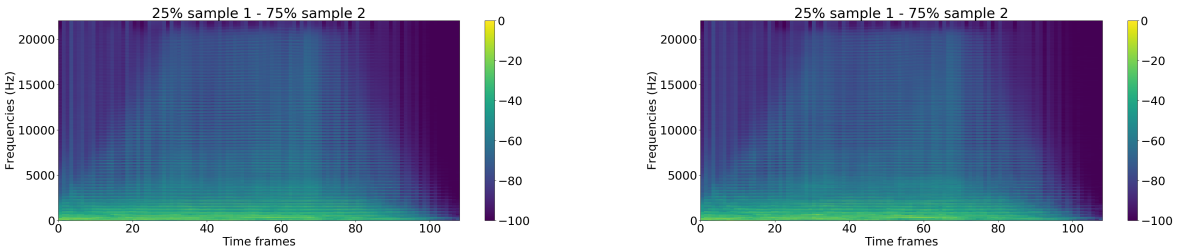
(a) Sample 1



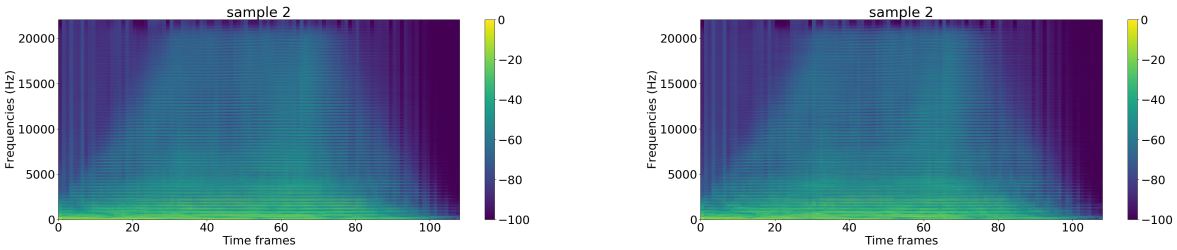
(b) 75% sample 1 – 25% sample 2



(c) 50% sample 1 – 50% sample 2

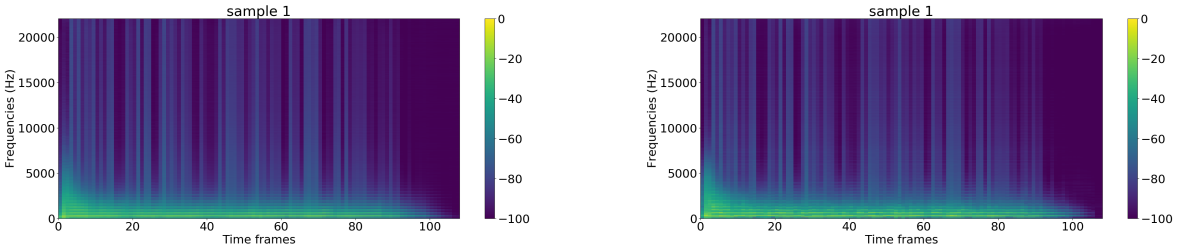


(d) 25% sample 1 – 75% sample 2

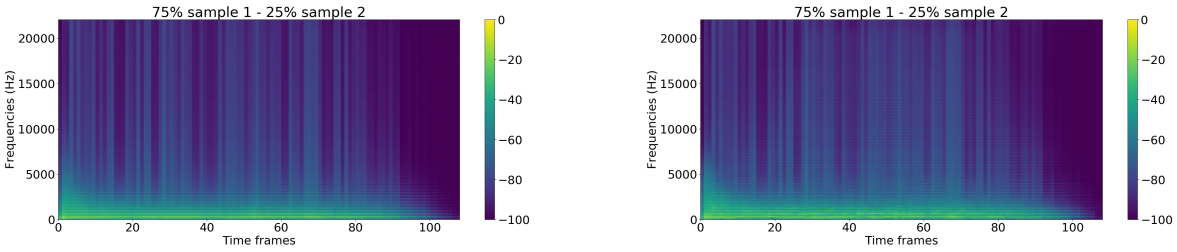


(e) Sample 2

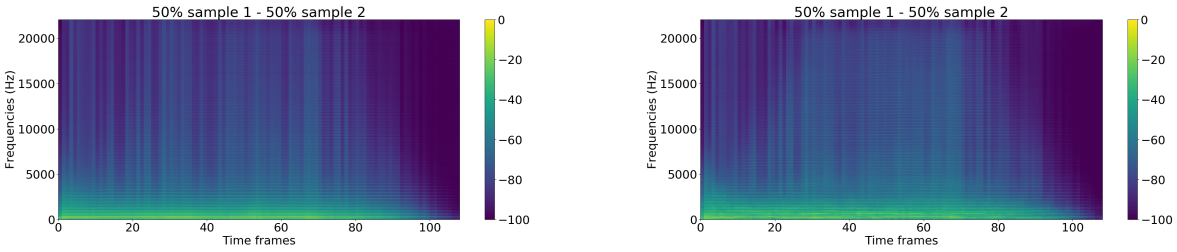
Figure D.4: Examples of decoded magnitude spectrograms after sound interpolation of 2 Arturia samples in the latent space using respectively PCA (left) and DAE (right)



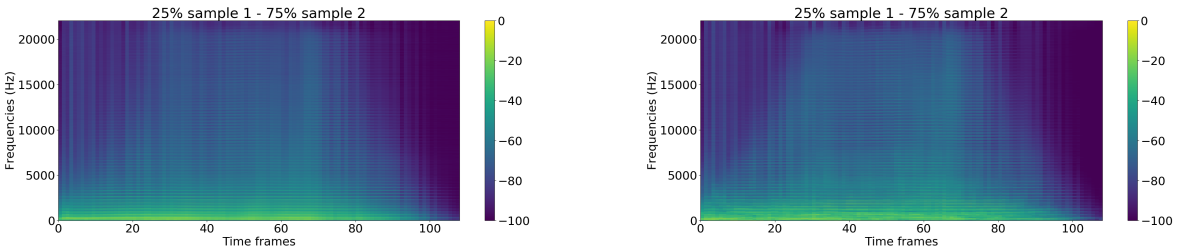
(a) Sample 1



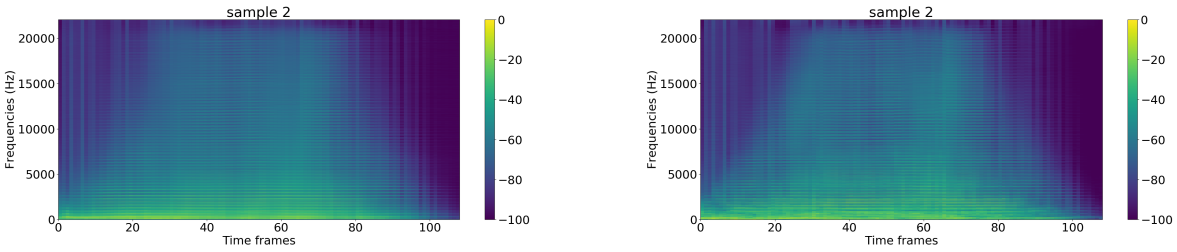
(b) 75% sample 1 – 25% sample 2



(c) 50% sample 1 – 50% sample 2



(d) 25% sample 1 – 75% sample 2



(e) Sample 2

Figure D.5: Examples of decoded magnitude spectrograms after sound interpolation of 2 Arturia samples in the latent space using respectively LSTM-AE (left) and VAE (right).

Appendix E

Weak perceptual supervision of the model additional material

E.1 A/B testing detailed protocol explanations

- ▷ Deux échantillons sonores A et B vous seront présentés. Vous pourrez les écouter autant de fois que vous le voulez en cliquant sur le bouton "Lecture".
- ▷ A n'importe quel moment vous pourrez régler le son à l'aide du curseur en haut à gauche de la page.
- ▷ Une fois les deux sons écoutés, vous pourrez sélectionner le son qui selon vous répond à la question posée juste au-dessus en cliquant sur la case contenant la lettre associée.
- ▷ Vous pourrez alors passer au son suivant en cliquant sur "Valider". En tout vous aurez 60 paires de sons à évaluer.
- ▷ Merci beaucoup pour votre participation ! Bon test !

Appendix F

Author publications

International conferences

2019 – Autoencoders for music sound modeling: a comparison of linear, shallow, deep, recurrent and variational models.

Fanny Roche, Thomas Hueber, Samuel Limier and Laurent Girin. In: *Proceedings of the Sound and Music Computing Conference (SMC)*. Málaga, Spain, 2019.

2019 – Notes on the use of variational autoencoders for speech and audio spectrogram modeling.

Laurent Girin, Thomas Hueber, Fanny Roche and Simon Leglaive. In: *Proceedings of the International Conference on Digital Audio Effects (DAFx)*. Birmingham, UK, 2019.

Journal article

2020 – Submitted article

Fanny Roche, Thomas Hueber, Maëva Garnier, Samuel Limier and Laurent Girin. *Article currently in a blind review process for publication in an international journal.*

Autoencoders for music sound modeling: a comparison of linear, shallow, deep, recurrent and variational models

Fanny Roche[1,3] **Thomas Hueber**[3] **Samuel Limier**[1] **Laurent Girin**[2,3]
[1]Arturia, Meylan, France [2]Inria Grenoble Rhône-Alpes, France
[3]Univ. Grenoble Alpes, CNRS, Grenoble INP, GIPSA-lab, Grenoble, France
fanny.roche@gipsa-lab.fr

ABSTRACT

This study investigates the use of non-linear unsupervised dimensionality reduction techniques to compress a music dataset into a low-dimensional representation which can be used in turn for the synthesis of new sounds. We systematically compare (shallow) autoencoders (AEs), deep autoencoders (DAEs), recurrent autoencoders (with Long Short-Term Memory cells – LSTM-AEs) and variational autoencoders (VAEs) with principal component analysis (PCA) for representing the high-resolution short-term magnitude spectrum of a large and dense dataset of music notes into a lower-dimensional vector (and then convert it back to a magnitude spectrum used for sound resynthesis). Our experiments were conducted on the publicly available multi-instrument and multi-pitch database NSynth. Interestingly and contrary to the recent literature on image processing, we can show that PCA systematically outperforms shallow AE. Only deep and recurrent architectures (DAEs and LSTM-AEs) lead to a lower reconstruction error. The optimization criterion in VAEs being the sum of the reconstruction error and a regularization term, it naturally leads to a lower reconstruction accuracy than DAEs but we show that VAEs are still able to outperform PCA while providing a low-dimensional latent space with nice "usability" properties. We also provide corresponding objective measures of perceptual audio quality (PEMO-Q scores), which generally correlate well with the reconstruction error.

1. INTRODUCTION

Deep neural networks, and in particular those trained in an unsupervised (or self-supervised) way such as autoencoders [1] or GANs [2], have shown nice properties to extract latent representations from large and complex datasets. Such latent representations can be sampled to generate new data. These types of models are currently widely used for image and video generation [3–5]. In the context of a project aiming at designing a music sound synthesizer driven by high-level control parameters and propelled by data-driven machine learning, we investigate the use of such techniques for music sound generation as an alternative to classical music sound synthesis techniques like additive synthesis, subtractive synthesis, frequency modulation, wavetable synthesis or physical modeling [6].

So far, only a few studies in audio processing have been proposed in this line, with a general principle that is similar to image synthesis/transformation: projection of the signal space into a low-dimensional latent space (encoding or embedding), modification of the latent coefficients, and inverse transformation of the modified latent coefficients into the original signal space (decoding).

In [7, 8], the authors implemented this principle with autoencoders to process normalized magnitude spectra. An autoencoder (AE) is a specific type of artificial neural network (ANN) architecture which is trained to reconstruct the input at the output layer, after passing through the latent space. Evaluation was made by computing the mean squared error (MSE) between the original and the reconstructed magnitude spectra.

In [9], NSynth, an audio synthesis method based on a time-domain autoencoder inspired from the WaveNet speech synthesizer [10] was proposed. The authors investigated the use of this model to find a high-level latent space well-suited for interpolation between instruments. Their autoencoder is conditioned on pitch and is fed with raw audio from their large-scale multi-instrument and multi-pitch database (the NSynth dataset). This approach led to promising results but has a high computational cost.

Another technique to synthesize data using deep learning is the so-called variational autoencoder (VAE) originally proposed in [11], which is now popular for image generation. A VAE can be seen as a probabilistic/generative version of an AE. Importantly, in a VAE, a prior can be placed on the distribution of the latent variables, so that they are well suited for the control of the generation of new data. This has been recently exploited for the modeling and transformation of speech signals [12, 13] and also for music sounds synthesis [14], incorporating some fitting of the latent space with a perceptual timbre space. VAEs have also been recently used for speech enhancement [15–17].

In line with the above-presented studies, the goal of the present paper is i) to provide an extensive comparison of several autoencoder architectures including shallow, deep, recurrent and variational autoencoders, with a systematic comparison to a linear dimensionality reduction technique, in the present case Principal Component Analysis (PCA) (to the best of our knowledge, such comparison of non-linear approaches with a linear one has never been done in previous studies). This is done using both an objec-

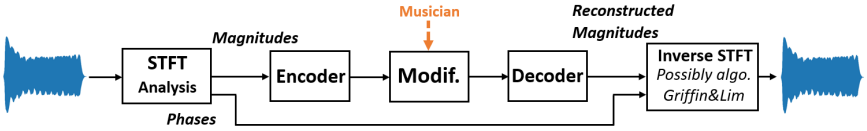


Figure 1: Global diagram of the sound analysis-transformation-synthesis process.

tive physical measure (root mean squared error – RMSE) and an objective perceptual measure (PEMO-Q [18]); ii) to compare the properties of the latent space in terms of correlation between the extracted dimensions; and iii) to illustrate how interpolation in the latent space can be performed to create interesting hybrid sounds.

2. METHODOLOGY

The global methodology applied for (V)AE-based analysis-transformation-synthesis of audio signals in this study is in line with previous works [7, 8, 12, 13]. It is illustrated in Fig. 1 and is described in the next subsections.

2.1 Analysis-Synthesis

First, a Short-Term Fourier Transform (STFT) analysis is performed on the input audio signal. The magnitude spectra are sent to the model (encoder input) on a frame-by-frame basis, and the phase spectra are stored for the synthesis stage. After possible modifications of the extracted latent variables (at the bottleneck layer output, see next subsection), the output magnitude spectra is provided by the decoder. The output audio signal is synthesized by combining the decoded magnitude spectra with the phase spectra, and by applying inverse STFT with overlap-add. If the latent coefficients are not modified in between encoding and decoding, the decoded magnitude spectra are close to the original ones and the original phase spectra can be directly used for good quality waveform reconstruction. If the latent coefficients are modified so that the decoded magnitude spectra become different from the original one, then the Griffin & Lim algorithm [19] is used to estimate/refine the phase spectra (the original phase spectra are used for initialization) and finally reconstruct the time-domain signal. A few more technical details regarding data pre-processing are given in Section 3.2.

2.2 Dimensionality Reduction Techniques

Principal Component Analysis: As a baseline, we investigated the use of PCA to reduce the dimensionality of the input vector $\mathbf{x}$. PCA is the optimal linear orthogonal transformation that provides a new coordinate system (i.e. the latent space) in which basis vectors follow modes of greatest variance in the original data [20].

Autoencoder: An AE is a specific kind of ANN traditionally used for dimensionality reduction thanks to its diabolo shape [21], see Fig. 2. It is composed of an encoder and a decoder. The encoder maps a high-dimensional low-level input vector $\mathbf{x}$ into a low-dimensional higher-level latent vector $\mathbf{z}$, which is assumed to nicely encode properties or attributes of $\mathbf{x}$. Similarly, the decoder reconstructs an estimate $\hat{\mathbf{x}}$ of the input vector $\mathbf{x}$ from the latent vector $\mathbf{z}$. The model is written as:

$$\mathbf{z} = f_{\text{enc}}(\mathbf{W}_{\text{enc}}\mathbf{x}+\mathbf{b}_{\text{enc}}) \quad \text{and} \quad \hat{\mathbf{x}} = f_{\text{dec}}(\mathbf{W}_{\text{dec}}\mathbf{z}+\mathbf{b}_{\text{dec}}),$$

where f_{enc} and f_{dec} are (entry-wise) non-linear activation functions, $\mathbf{W}_{\text{enc}}$ and $\mathbf{W}_{\text{dec}}$ are weight matrices and $\mathbf{b}_{\text{enc}}$ and $\mathbf{b}_{\text{dec}}$ are bias vectors. For regression tasks (such as the one considered in this study), a linear activation function is generally used for the output layer.

At training time, the weight matrices and the bias vectors are learned by minimizing some cost function over a training dataset. Here we consider the mean squared error (MSE) between the input $\mathbf{x}$ and the output $\hat{\mathbf{x}}$.

The model can be extended by adding hidden layers in both the encoder and decoder to create a so-called deep autoencoder (DAE), as illustrated in Fig. 2. This kind of architecture can be trained globally (end-to-end) or layer-by-layer by considering the DAE as a stack of shallow AEs [1, 22].

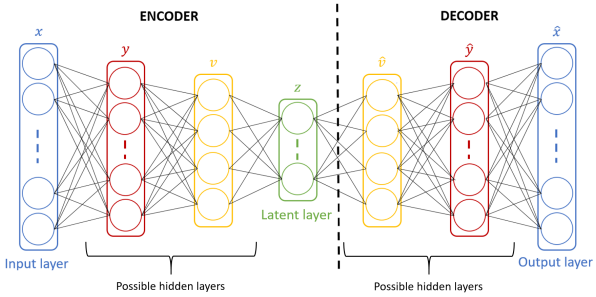


Figure 2: General architecture of a (deep) autoencoder.

LSTM Autoencoder: In a general manner, a recurrent neural network (RNN) is an ANN where the output of a given hidden layer does not depend only on the output of the previous layer (as in a feedforward architecture) but also on the internal state of the network. Such internal state can be defined as the output of each hidden neuron when processing the previous input observations. They are thus well-suited to process time series of data and capture their time dependencies. Such networks are here expected to extract latent representations that encode some aspects of the sound dynamics. Among different existing RNN architectures, in this study we used the Long Short-Term Memory (LSTM) network [23], which is known to tackle correctly the so-called vanishing gradient problem in RNNs [24]. The structure of the model depicted in Fig. 2 still holds while replacing the classical neuronal cells by LSTM cells, leading to a LSTM-AE. The cost function to optimize remains the same, i.e. the MSE between the input $\mathbf{x}$ and the

ISBN 978-84-09-08518-7

output $\hat{\mathbf{x}}$. However, the model is much more complex and has more parameters to train [23].

Variational Autoencoder: A VAE can be seen as a probabilistic AE which delivers a parametric model of the data distribution, such as:

$$p_\theta(\mathbf{x}, \mathbf{z}) = p_\theta(\mathbf{x}|\mathbf{z})p_\theta(\mathbf{z}),$$

where θ denotes the set of distribution parameters. In the present context, the likelihood function $p_\theta(\mathbf{x}|\mathbf{z})$ plays the role of a probabilistic decoder which models how the generation of observed data $\mathbf{x}$ is conditioned on the latent data $\mathbf{z}$. The prior distribution $p_\theta(\mathbf{z})$ is used to structure (or regularize) the latent space. Typically a standard Gaussian distribution $p_\theta(\mathbf{z}) = \mathcal{N}(\mathbf{z}; \mathbf{0}, \mathbf{I})$ is used, where $\mathbf{I}$ is the identity matrix [11]. This encourages the latent coefficients to be mutually orthogonal and lie on a similar range. Such properties may be of potential interest for using the extracted latent coefficients as control parameters of a music sound generator. The likelihood $p_\theta(\mathbf{x}|\mathbf{z})$ is defined as a Gaussian density:

$$p_\theta(\mathbf{x}|\mathbf{z}) = \mathcal{N}(\mathbf{x}; \boldsymbol{\mu}_\theta(\mathbf{z}), \boldsymbol{\sigma}^2_\theta(\mathbf{z})),$$

where $\boldsymbol{\mu}_\theta(\mathbf{z})$ and $\boldsymbol{\sigma}^2_\theta(\mathbf{z})$ are the outputs of the decoder network (hence $\theta = \{\mathbf{W}_{\text{dec}}, \mathbf{b}_{\text{dec}}\}$). Note that $\boldsymbol{\sigma}^2_\theta(\mathbf{z})$ indifferently denotes the covariance matrix of the distribution, which is assumed diagonal, or the vector of its diagonal entries.

The exact posterior distribution $p_\theta(\mathbf{z}|\mathbf{x})$ corresponding to the above model is intractable. It is approximated with a tractable parametric model $q_\phi(\mathbf{z}|\mathbf{x})$ that will play the role of the corresponding probabilistic encoder. This model generally has a form similar to the decoder:

$$q_\phi(\mathbf{z}|\mathbf{x}) = \mathcal{N}(\mathbf{z}; \tilde{\boldsymbol{\mu}}_\phi(\mathbf{x}), \tilde{\boldsymbol{\sigma}}^2_\phi(\mathbf{x})),$$

where $\tilde{\boldsymbol{\mu}}_\phi(\mathbf{x})$ and $\tilde{\boldsymbol{\sigma}}^2_\phi(\mathbf{x})$ are the outputs of the encoder ANN (the parameter set ϕ is composed of $\mathbf{W}_{enc}$ and $\mathbf{b}_{enc}$; $\tilde{\boldsymbol{\sigma}}^2_\phi(\mathbf{x})$ is a diagonal covariance matrix or is the vector of its diagonal entries).

Training of the VAE model, i.e. estimation of θ and ϕ, is done by maximizing the marginal log-likelihood $\log p_\theta(\mathbf{x})$ over a large training dataset of vectors $\mathbf{x}$. It can be shown that the marginal log-likelihood can be written as [11]:

$$\log p_\theta(\mathbf{x}) = \mathrm{D}_{\mathrm{KL}}(q_\phi(\mathbf{z}|\mathbf{x})|p_\theta(\mathbf{z}|\mathbf{x})) + \mathcal{L}(\phi, \theta, \mathbf{x}),$$

where $\mathrm{D}_{\mathrm{KL}} \geq 0$ denotes the Kullback-Leibler divergence (KLD) and $\mathcal{L}(\phi, \theta, \mathbf{x})$ is the variational lower bound (VLB) given by:

$$\mathcal{L}(\phi, \theta, \mathbf{x}) = \underbrace{-\mathrm{D}_{\mathrm{KL}}(q_\phi(\mathbf{z}|\mathbf{x})|p_\theta(\mathbf{z}))}_{\text{regularization}} + \underbrace{\mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}[\log p_\theta(\mathbf{x}|\mathbf{z})]}_{\text{reconstruction accuracy}}. \quad (1)$$

In practice, the model is trained by maximizing $\mathcal{L}(\phi, \theta, \mathbf{x})$ over the training dataset with respect to parameters ϕ and θ. We can see that the VLB is the sum of two terms. The first term acts as a regularizer encouraging the approximate posterior $q_\phi(\mathbf{z}|\mathbf{x})$ to be close to the prior $p_\theta(\mathbf{z})$. The second term represents the average reconstruction accuracy. Since the expectation w.r.t. $q_\phi(\mathbf{z}|\mathbf{x})$ is difficult to compute analytically, it is approximated using a Monte Carlo estimate and samples drawn from $q_\phi(\mathbf{z}|\mathbf{x})$. For other technical details that are not relevant here, the reader is referred to [11].

As discussed in [12] and [25], a weighting factor, denoted β, can be introduced in (1) to balance the regularization and reconstruction terms:

$$\mathcal{L}(\phi, \theta, \beta, \mathbf{x}) = -\beta\, \mathrm{D}_{\mathrm{KL}}(q_\phi(\mathbf{z}|\mathbf{x})|p_\theta(\mathbf{z})) + \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}[\log p_\theta(\mathbf{x}|\mathbf{z})], \quad (2)$$

This enables the user to better control the trade-off between output signal quality and compactness/orthogonality of the latent coefficients $\mathbf{z}$. Indeed, if the reconstruction term is too strong relatively to the regularization term, then the distribution of the latent space will be poorly constrained by the prior $p_\theta(\mathbf{z})$, turning the VAE into an AE. Conversely, if it is too weak, then the model may focus too much on constraining the latent coefficients to follow the prior distribution while providing poor signal reconstruction [25]. In the present work we used this type of β-VAE and we present the results obtained with different values of β. These latter were selected manually after pilot experiments to ensure that the values of the regularization and the reconstruction accuracy terms in (2) are in the same range.

3. EXPERIMENTS

3.1 Dataset

In this study, we used the NSynth dataset introduced in [9]. This is a large database (more than 30 GB) of 4s long monophonic music sounds sampled at 16 kHz. They represent 1,006 different instruments generating notes with different pitches (from MIDI 21 to 108) and different velocities (5 different levels from 25 to 127). To generate these samples different methods were used: Some acoustic and electronic instruments were recorded and some others were synthesized. The dataset is labeled with: i) instrument family (e.g., keyboard, guitar, synth_lead, reed), ii) source (acoustic, electronic or synthetic), iii) instrument index within the instrument family, iv) pitch value, and v) velocity value. Some other labels qualitatively describe the samples, e.g. brightness or distortion, but they were not used in our work.

To train our models, we used a subset of 10,000 different sounds randomly chosen from this NSynth database, representing all families of instruments, different pitches and different velocities. We split this dataset into a training set (80%) and testing set (20%). During the training phase, 20% of the training set was kept for validation. In order to have a statistically robust evaluation, a k-fold cross-validation procedure with $k = 5$ was used to train and test all different models (we divided the dataset into 5 folds, used 4 of them for training and the remaining one for test, and repeated this procedure 5 times so that each sound of the initial dataset was used once for testing).

3.2 Data Pre-Processing

For magnitude and phase short-term spectra extraction, we applied a 1,024-point STFT to the input signal using a sliding Hamming window with 50% overlap. Frames corre-

ISBN 978-84-09-08518-7

sponding to silence segments were removed. The corresponding 513-point positive-frequency magnitude spectra were then converted to log-scale and normalized in energy: We fixed the maximum of each log-spectrum input vector to 0 dB (the energy coefficient was stored to be used for signal reconstruction). Then, the log-spectra were thresholded, i.e. every log-magnitude below a fixed threshold was set to the threshold value. Finally they were normalized between -1 and 1, which is a usual procedure for ANN inputs. Three threshold values were tested: -80 dB, -90 dB and -100 dB. Corresponding denormalization, log-to-linear conversion and energy equalization were applied after the decoder, before signal reconstruction with transmitted phases and inverse STFT with overlap-add.

3.3 Autoencoder Implementations

We tried different types of autoencoders: AE, DAE, LSTM-AE and VAE. For all the models we investigated several values for the encoding dimension, i.e. the size of the bottleneck layer / latent variable vector, from $enc = 4$ to 100 (with a fine-grained sampling for $enc \leq 16$). Different architectures were tested for the DAEs: [513, 128, enc, 128, 513], [513, 256, enc, 256, 513] and [513, 256, 128, enc, 128, 256, 513]. Concerning the LSTM-AE, our implementation used two vanilla forward LSTM layers (one for the encoder and one for the decoder) with non-linear activation functions giving the following architecture: [513, enc, 513]. Both LSTM layers were designed for many-to-many sequence learning, meaning that a sequence of inputs, i.e. of spectral magnitude vectors, is encoded into a sequence of latent vectors of same temporal size and then decoded back to a sequence of reconstructed spectral magnitude vectors. The architecture we used for the VAE was [513, 128, enc, 128, 513] and we tested different values of the weight factor β. For all the neural models, we tested different pairs of activation functions for the hidden layers and output layer, respectively: (tanh, linear), (sigmoid, linear) and (tanh, sigmoid).

AE, DAE, LSTM-AE and VAE models were implemented using the *Keras* toolkit [26] (we used the *scikit-learn* [27] toolkit for the PCA). Training was performed using the Adam optimizer [28] with a learning rate of 10^{-3} over 600 epochs with early stopping criterion (with a patience of 30 epochs) and a batch size of 512. The DAEs were trained in two different ways, with and without layer-wise training.

3.4 Experimental Results for Analysis-Resynthesis

Fig. 3 shows the reconstruction error (RMSE in dB) obtained with PCA, AE, DAE and LSTM-AE models on the test set (averaged over the 5 folds of the cross-validation procedure), as a function of the dimension of the latent space. The results obtained with the VAE (using the same protocol, and for different β values) are shown in Fig. 4. For the sake of clarity, we present here only the results obtained for i) a threshold of -100 dB applied on the log-spectra, and ii) a restricted set of the tested AE, DAE and VAE architectures (listed in the legends of the figures). Similar trends were observed for other thresholds and other tested architectures. For each considered dimension of the latent space, a 95% confidence interval of each reconstruction error was obtained by conducting paired t-test, considering each sound (i.e. each audio file) of the test set as an independent sample.

RMSE provides a global measure of magnitude spectra reconstruction but can be insufficiently correlated to perception depending on which spectral components are correctly or poorly reconstructed. To address this classical issue in audio processing, we also calculated objective measures of perceptual audio quality, namely PEMO-Q scores [18]. The results are reported in Fig. 5 and Fig. 6.

As expected, the RMSE decreases with the dimension of the latent space for all methods. Interestingly, PCA systematically outperforms (or at worst equals) shallow AE. This somehow contradicts recent studies on image compression for which a better reconstruction is obtained with AE compared to PCA [1]. To confirm this unexpected result, we replicated our PCA vs. AE experiment on the MNIST image dataset [29], using the same AE implementation and a standard image preprocessing (i.e. vectorization of each 28×28 pixels gray-scale image into a 784-dimensional feature vector). In accordance with the literature, the best performance was systematically obtained with AE (for any considered dimension of the latent space). This difference of AE's behavior when considering audio and image data was unexpected and, to our knowledge, it has never been reported in the literature.

Then, contrary to (shallow) AE, DAEs systematically outperform PCA (and thus AE), with up to almost 20% improvement (for $enc = 12$ and $enc = 16$). Our experiments did not reveal notable benefit of layer-by-layer DAE training over end-to-end training. Importantly, for small dimensions of the latent space (e.g. smaller than 16), RMSE obtained with DAE decreases much faster than with PCA and AE. This is even more the case for LSTM-AE which shows an improvement of the reconstruction error of more than 23% over PCA (for $enc = 12$ and $enc = 16$). These results confirm the benefits of using a more complex architecture than shallow AE, here deep or recurrent, to efficiently extract high-level abstractions and compress the audio space. This is of great interest for sound synthesis for which the latent space has to be kept as low-dimensional as possible (while maintaining a good reconstruction accuracy) in order to be "controlled" by a musician.

Fig. 4 shows that the overall performance of VAEs is in between the performance of DAEs (even equals DAEs for lower encoding dimensions, say smaller than 12) and the performances of PCA and AE. Let us recall that minimizing the reconstruction accuracy is not the only goal of VAE which also aims at constraining the distribution of the latent space. As shown in Fig. 4, the parameter β, which balances regularization and reconstruction accuracy in (2), plays a major role. As expected, high β values foster regularization at the expense of reconstruction accuracy. However, with $\beta \leqslant 2.10^{-6}$ the VAE clearly outperforms PCA, e.g. up to 20% for $enc = 12$.

It can be noticed that when the encoding dimension is high ($enc = 100$), PCA seems to outperform all the other models. Hence, in that case, the simpler (linear model)

ISBN 978-84-09-08518-7

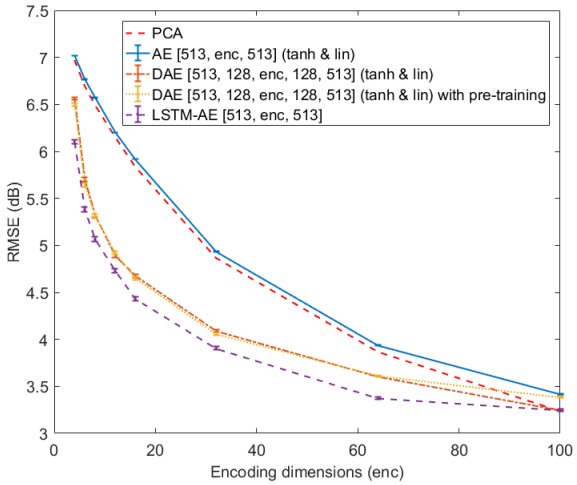


Figure 3: Reconstruction error (RMSE in dB) obtained with PCA, AE, DAE (with and without layer-wise training) and LSTM-AE, as a function of latent space dimension.

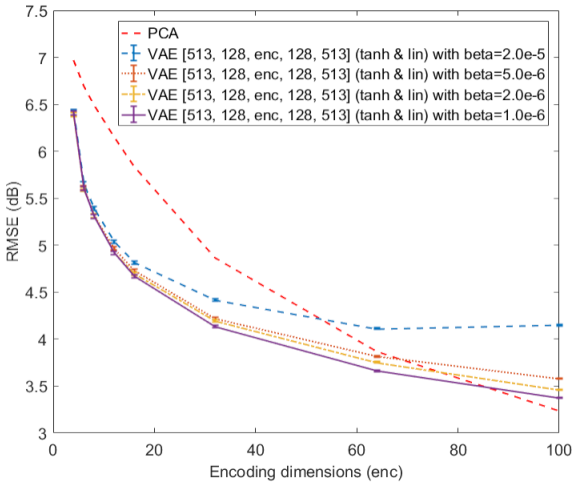


Figure 4: Reconstruction error (RMSE in dB) obtained with VAEs as a function of latent space dimension (RMSE obtained with PCA is also recalled).

seems to be the best (we can conjecture that achieving the same level of performance with autoencoders would require more training data, since the number of free parameters of these model increases drastically). However, using such high-dimensional latent space as control parameters of a music sound generator is impractical.

Similar conclusions can be drawn from Fig. 5 and Fig. 6 in terms of audio quality. Indeed, in a general manner, the PEMO-Q scores are well correlated with RMSE measures in our experiments. PEMO-Q measures for PCA and AE are very close, but PCA still slightly outperforms the shallow AE. The DAEs and the VAEs both outperform the PCA (up to about 11% for $enc = 12$ and $enc = 16$) with the audio quality provided by the DAEs being a little better than for the VAEs. Surprisingly, and contrary to RMSE scores, the LSTM-AE led to a (slightly) lower PEMO-Q scores, for all considered latent dimensions. Further investigations will be done to assess the relevance of such differences at the perceptual level.

3.5 Decorrelation of the Latent Dimensions

Now we report further analyses aiming at investigating how the extracted latent dimensions may be used as *control* parameters by the musician. In the present sound synthesis framework, such control parameters are expected to respect (at least) the following two constraints i) to be as decorrelated as possible in order to limit the redundancy in the spectrum encoding, ii) to have a clear and easy-to-understand perceptual meaning. In the present study, we focus on the first constraint by comparing PCA, DAEs, LSTM-AE and VAEs in terms of correlation of the latent dimensions. More specifically, the absolute values of the correlation coefficient matrices of the latent vector $\mathbf{z}$ were computed on each sound from the test dataset and Fig. 7 reports the mean values averaged over all the sounds of the test dataset. For the sake of clarity, we present here these results only for a latent space of dimension 16 for one model of DAE ([513, 128, 16, 128, 513] (tanh & lin) with end-to-end training) and for VAEs with the same architecture ([513, 128, 16, 128, 513] (tanh & lin)) and different values of β (from 1.10^{-6} to 2.10^{-5}).

As could be expected from the complexity of its structure, we can see that the LSTM-AE extracts a latent space where the dimensions are significantly correlated with each other. Such additional correlations may come from the sound dynamics which provide redundancy in the prediction. We can also see that PCA and VAEs present similar behaviors with much less correlation of the latent dimensions, which is an implicit property of these models. Interestingly, and in accordance with (2), we can notice that the higher the β, the more regularized the VAE and hence the more decorrelated the latent dimensions. Importantly, Fig. 7 clearly shows that for a well-chosen β value, the VAE can both extract latent dimensions that are much less correlated than for corresponding DAEs, which makes it a better candidate for extracting good control parameters, while allowing fair to good reconstruction accuracy (see Fig. 4). The β value has thus to be chosen wisely in order to find the optimal trade-off between decorrelation of the latent dimensions and reconstruction accuracy.

3.6 Examples of Sound Interpolation

As a first step towards the practical use of the extracted latent space for navigating through the sound space and creating new sounds, we illustrate how it can be used to interpolate between sounds, in the spirit of what was done for instrument hybridization in [9]. We selected a series of pairs of sounds from the NSynth dataset with the two sounds in a pair having different characteristics. For each pair, we proceeded to separate encoding, entry-wise linear interpolation of the two resulting latent vectors, decoding, and finally individual signal reconstruction with inverse STFT and the Griffin and Lim algorithm to reconstruct the phase spectrogram [19]. We experimented dif-

ISBN 978-84-09-08518-7

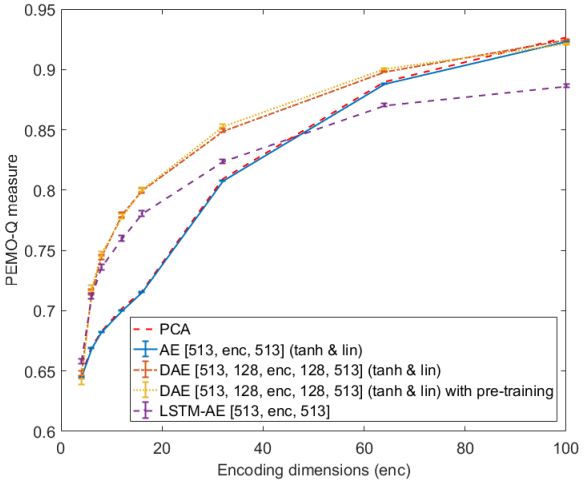


Figure 5: PEMO-Q measures obtained with PCA, AE, DAEs (with and without layer-wise training) and LSTM-AE, as a function of latent space dimension.

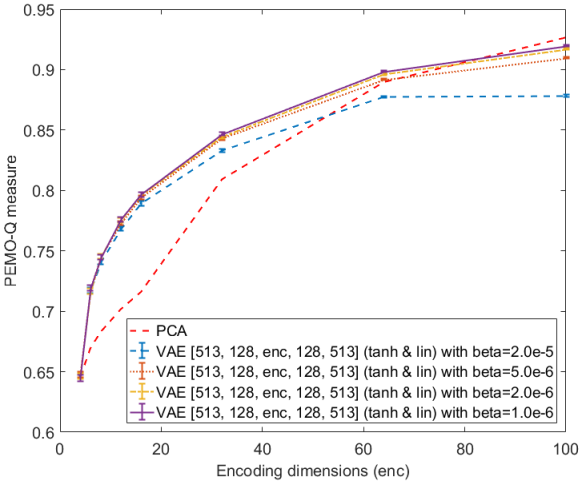


Figure 6: PEMO-Q measures obtained with VAEs as a function of latent space dimension (measures obtained with PCA are also recalled).

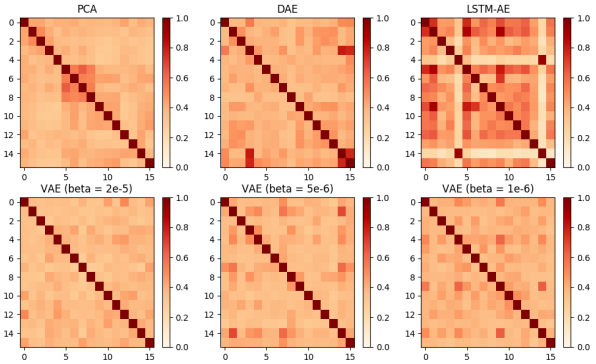


Figure 7: Correlation matrices of the latent dimensions (average absolute correlation coefficients) for PCA, DAE, LSTM-AE and VAEs.

ferent degrees of interpolation between the two sounds: $\hat{\mathbf{z}} = \alpha\ \mathbf{z}_1 + (1-\alpha)\ \mathbf{z}_2$, with $\mathbf{z}_i$ the latent vector of sound i, $\hat{\mathbf{z}}$ the new interpolated latent vector, and $\alpha \in [0, 0.25, 0.5, 0.75, 1]$ (this interpolation is processed independently on each pair of vectors of the time sequence). The same process was applied using the different AE models we introduced earlier.

Fig. 8 displays one example of results obtained with PCA, with the LSTM-AE and with the VAE (with $\beta = 1.10^{-6}$), with an encoding dimension of 32. Qualitatively, we note that interpolations in the latent space lead to a smooth transition between source and target sound. By increasing sequentially the degree of interpolation, we can clearly go from one sound to another in a consistent manner, and create interesting hybrid sounds. The results obtained using PCA interpolation are (again qualitatively) below the quality of the other models. The example spectrogram obtained with interpolated PCA coefficients is blurrier around the harmonics and some audible artifacts appear. On the opposite, the LSTM-AE seems to outperform the other models by better preserving the note attacks (see comparison with VAE in Fig. 8). More interpolation examples along with corresponding audio samples can be found at `https://goo.gl/Tvvb9e`.

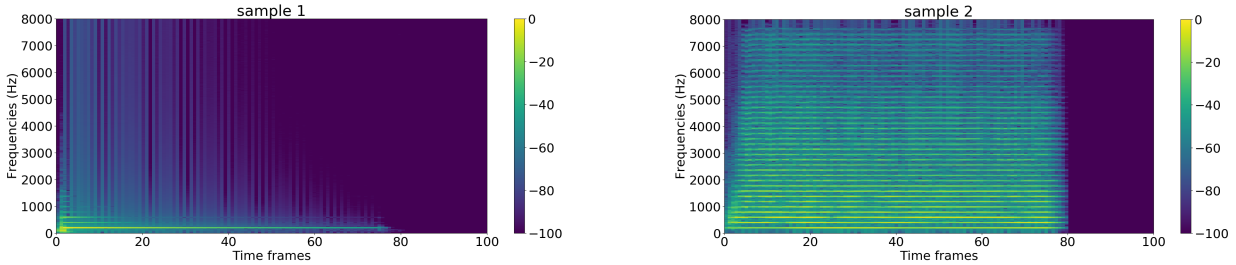
4. CONCLUSIONS AND PERSPECTIVES

In this study, we investigated dimensionality reduction based on autoencoders to extract latent dimensions from a large music sound dataset. Our goal is to provide a musician with a new way to generate sound textures by exploring a low-dimensional space. From the experiments conducted on a subset of the publicly available database NSynth, we can draw the following conclusions: i) Contrary to the literature on image processing, shallow autoencoders (AEs) do not here outperform principal component analysis (in terms of reconstruction accuracy); ii) The best performance in terms of signal reconstruction is always obtained with deep or recurrent autoencoders (DAEs or LSTM-AE); iii) Variational autoencoders (VAEs) lead to a fair-to-good reconstruction accuracy while constraining the statistical properties of the latent space, ensuring some amount of decorrelation across latent coefficients and limiting their range. These latter properties make the VAEs good candidates for our targeted sound synthesis application.

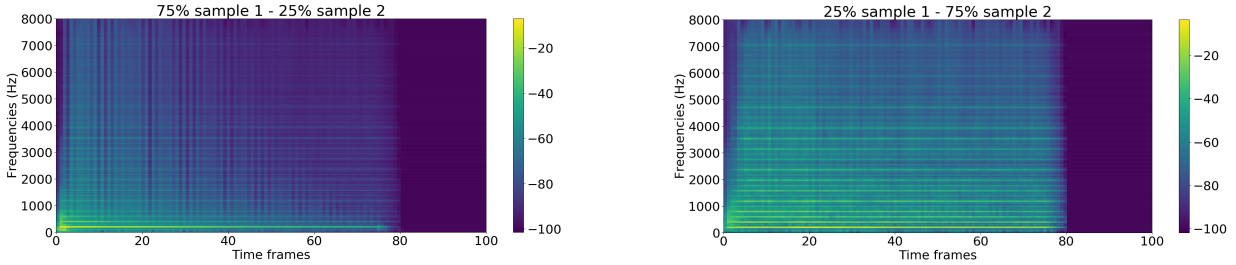
In line with the last conclusion, future works will mainly focus on VAEs. First, we will investigate recurrent architecture for VAE such as the one proposed in [30]. Such approach may lead to latent dimensions encoding separately the sound texture and its dynamics, which may be of potential interest for the musician.

Then, we will address the crucial question of the perceptual meaning/relevance of the latent dimensions. Indeed using a non-informative prior distribution of $\mathbf{z}$ such as a standard normal distribution does not ensure that each dimension of $\mathbf{z}$ represents an interesting perceptual dimension of the sound space, although this is a desirable objec-

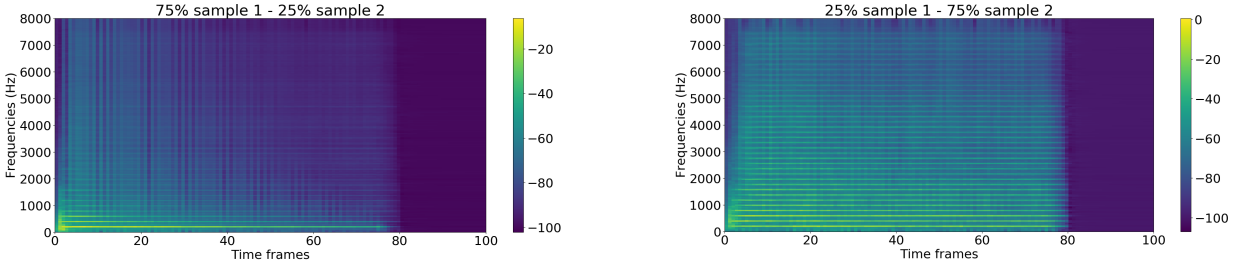
ISBN 978-84-09-08518-7



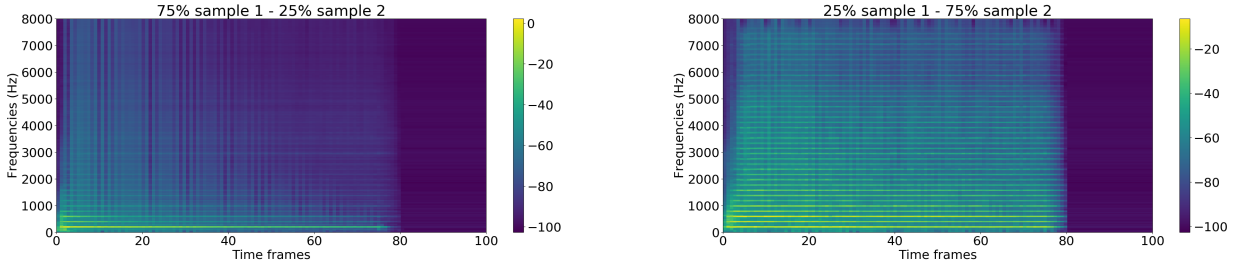
(a) Original samples - Left : bass_electronic_010-055-100, Right : brass_acoustic_050-055-100



(b) PCA



(c) LSTM-AE



(d) VAE

Figure 8: Examples of decoded magnitude spectrograms after sound interpolation of 2 samples (top) in the latent space using respectively PCA (2nd row), LSTM-AE (3rd row) and VAE (bottom). A more detailed version of the figure can be found at `https://goo.gl/Tvvb9e`.

tive. In [14], the authors recently proposed a first solution to this issue in the context of a restricted set of acoustic instruments. They introduced in the variational lower bound (2) of the VAE loss an additional regularization term encouraging the latent space to respect the structure of the instrument timbre. In the same spirit, our future works will investigate different strategies to model the complex relationships between sound textures and their perception, and introduce these models at the VAE latent space level.

5. ACKNOWLEDGMENT

The authors would like to thank Simon Leglaive for our fruitful discussions. This work was supported by ANRT in the framework of the PhD program CIFRE 2016/0942.

6. REFERENCES

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

[2] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in Neural Information Process. Systems*, Montreal, Canada, 2014.

[3] S. Reed, Z. Akata, X. Yan, L. Logeswaran, B. Schiele, and H. Lee, "Generative adversarial text to image synthesis," in *Proc. of the Int. Conf. on Machine Learning*, New York, NY, 2016.

[4] S. Mehri, K. Kumar, I. Gulrajani, R. Kumar, S. Jain, J. Sotelo, A. Courville, and Y. Bengio, "SampleRNN: An unconditional end-to-end neural audio generation model." in *Int. Conf. on Learning Representations*, 2017.

[5] S. Tulyakov, M.-Y. Liu, X. Yang, and J. Kautz, "Mocogan: Decomposing motion and content for video generation," in *Proc. of the IEEE conf. on computer vision and pattern recognition*, 2018.

[6] E. Miranda, *Computer Sound Design: Synthesis Techniques and Programming*, ser. Music Technology series. Focal Press, 2002.

[7] A. Sarroff and M. Casey, "Musical audio synthesis using autoencoding neural nets," in *Joint Int. Computer Music Conf. and Sound and Music Computing Conf.*, Athens, Greece, 2014.

[8] J. Colonel, C. Curro, and S. Keene, "Improving neural net auto encoders for music synthesis," in *Audio Engineering Society Convention*, New-York, NY, 2017.

[9] J. Engel, C. Resnick, A. Roberts, S. Dieleman, D. Eck, K. Simonyan, and M. Norouzi, "Neural audio synthesis of musical notes with wavenet autoencoders," *arXiv preprint arXiv:1704.01279*, 2017.

[10] A. Van Den Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. Senior, and K. Kavukcuoglu, "Wavenet: A generative model for raw audio," *arXiv preprint arXiv:1609.03499*, 2016.

[11] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," *arXiv preprint arXiv:1312.6114*, 2013.

[12] M. Blaauw and J. Bonada, "Modeling and transforming speech using variational autoencoders." in *Conf. of the Int. Speech Comm. Association (Interspeech)*, San Francisco, CA, 2016.

[13] W.-N. Hsu, Y. Zhang, and J. Glass, "Learning latent representations for speech generation and transformation," *arXiv preprint arXiv:1704.04222*, 2017.

[14] P. Esling, A. Chemla-Romeu-Santos, and A. Bitton, "Generative timbre spaces with variational audio synthesis," in *Proc. of the Int. Conf. on Digital Audio Effects 2018*, Aveiro, Portugal, 2018.

[15] S. Leglaive, L. Girin, and R. Horaud, "A variance modeling framework based on variational autoencoders for speech enhancement," in *IEEE Int. Workshop on Machine Learning for Signal Process.*, Aalborg, Denmark, 2018.

[16] ——, "Semi-supervised multichannel speech enhancement with variational autoencoders and non-negative matrix factorization," in *IEEE Int. Conf. on Acoustics, Speech and Signal Process.*, Brighton, UK, 2019.

[17] L. Li, H. Kameoka, and S. Makino, "Fast MVAE: Joint separation and classification of mixed sources based on multichannel variational autoencoder with auxiliary classifier," in *IEEE Int. Conf. on Acoustics, Speech and Signal Process.*, Brighton, UK, 2019.

[18] R. Huber and B. Kollmeier, "PEMO-Q: A new method for objective audio quality assessment using a model of auditory perception," *IEEE Transactions on Audio, Speech, and Language Process.*, vol. 14, no. 6, pp. 1902–1911, 2006.

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

[20] C. M. Bishop, *Pattern recognition and machine learning*. Springer, 2006.

[21] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016.

[22] Y. Bengio, P. Lamblin, D. Popovici, and H. Larochelle, "Greedy layer-wise training of deep networks," in *Advances in Neural Information Process. Systems*, Vancouver, Canada, 2007.

[23] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[24] Y. Bengio, P. Simard, and P. Frasconi, "Learning long-term dependencies with gradient descent is difficult," *IEEE transactions on neural networks*, vol. 5, no. 2, pp. 157–166, 1994.

[25] I. Higgins, L. Matthey, A. Pal, C. Burgess, X. Glorot, M. Botvinick, S. Mohamed, and A. Lerchner, "β-VAE: learning basic visual concepts with a constrained variational framework." in *Int. Conf. on Learning Representations*, 2017.

[26] F. Chollet *et al.*, "Keras," https://keras.io, 2015.

[27] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of machine learning research*, vol. 12, no. Oct, pp. 2825–2830, 2011.

[28] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[29] Y. LeCun, C. Cortes, and C. Burges, "The MNIST database of handwritten digits," http://yann.lecun.com/exdb/mnist/, 1998.

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

NOTES ON THE USE OF VARIATIONAL AUTOENCODERS FOR SPEECH AND AUDIO SPECTROGRAM MODELING

Laurent Girin, Thomas Hueber
Univ. Grenoble Alpes, CNRS,
Grenoble INP, GIPSA-lab
Grenoble, France
laurent.girin@grenoble-inp.fr
thomas.hueber@grenoble-inp.fr

Fanny Roche *
Arturia
Meylan, France
fanny.roche@arturia.com

Simon Leglaive
Inria Grenoble Rhône-Alpes
Grenoble, France
simon.leglaive@inria.fr

ABSTRACT

Variational autoencoders (VAEs) are powerful (deep) generative artificial neural networks. They have been recently used in several papers for speech and audio processing, in particular for the modeling of speech/audio spectrograms. In these papers, very poor theoretical support is given to justify the chosen data representation and decoder likelihood function or the corresponding cost function used for training the VAE. Yet, a nice theoretical statistical framework exists and has been extensively presented and discussed in papers dealing with nonnegative matrix factorization (NMF) of audio spectrograms and its application to audio source separation. In the present paper, we show how this statistical framework applies to VAE-based speech/audio spectrogram modeling. This provides the latter insights on the choice and interpretability of data representation and model parameterization.

1. INTRODUCTION

Autoencoders (AEs) are a specific type of deep neural networks (DNNs) that can learn from data a non-linear projection of the signal space into a low-dimensional latent space (encoding step), followed by inverse non-linear transformation of the latent coefficients into the original signal space (decoding step) [1]. AEs have been essentially used as an unsupervised technique for data dimension reduction. More recently, variational autoencoders (VAEs) were proposed as a probabilistic/generative extension of AEs [2]: Instead of deterministically mapping the input vector $\mathbf{x}$ into a unique vector of latent coefficients $\mathbf{z}$, as done in AEs, the VAE *encoder network* maps $\mathbf{x}$ into the parameters of a conditional distribution $q_\phi(\mathbf{z}|\mathbf{x})$ of $\mathbf{z}$. Similarly, the *decoder network* maps a vector of latent coefficient $\mathbf{z}$ into the parameters of a conditional distribution $p_\theta(\mathbf{x}|\mathbf{z})$ of $\mathbf{x}$. A VAE decoder is thus intrinsically a (non-linear and deep) generative model of $\mathbf{x}$, conditioned on the latent variable $\mathbf{z}$ (which is itself conditioned on the input when decoding follows encoding). VAEs thus combine the modeling power of DNNs with the flexibility of generative models.

VAEs have recently received a strong interest for speech and audio processing, more specifically for modeling, transformation and synthesis of speech signals [3, 4, 5, 6], for music sound synthesis [7, 8], and for single-channel [9, 10, 11, 12] and multi-channel [13, 14, 15] speech enhancement and separation. In all those papers, VAEs are used to process a sequence of vectors encoding the short-time Fourier transform (STFT) spectrogram extracted from speech or music signals. For synthesis/transformation applications, the output audio signal is reconstructed using the decoded magnitude spectrogram, after possible modification of the latent coefficients, and either the phase of the original signal or some reconstructed phase more coherent with the decoded magnitude spectrogram. For speech enhancement application, the decoder of the VAE is used as a supervised generative model of the speech signal in the STFT domain, which is exploited in a probabilistic enhancement/separation method.

A keypoint is that in most of these papers, very few justification is given about the precise choice of the encoder and decoder conditional distributions, or the corresponding cost function used for VAE training. These distributions are generally chosen as Gaussian for convenience, but the choice for their parameters is not clearly justified. The same about the related issue of data representation: It is chosen a bit arbitrarily, without clear theoretical support, possibly more considering DNN training issues rather than fundamental signal processing ones.

Yet, this theoretical framework exists. In fact, it has been extensively presented and discussed in the seminal papers [16] and [17]. Those papers describe the statistical framework underlying the decomposition of audio magnitude/power spectrograms using Nonnegative Matrix Factorization (NMF) [18]. These developments have then been extensively used for audio source separation, see e.g. among many others [19, 20, 21, 22, 23, 24, 25]. In the present paper, we show how this theoretical statistical framework applies to the VAE model. Based on [16, 17], we describe the three main cases encountered in practice, with three modeling cost functions corresponding to three signal statistical models. We show how this provides interesting insights on the choice and interpretability of data representation and loss function for speech/audio spectrogram modeling with VAEs.

The remainder of this paper is organized as follows. Section 2 presents the VAE framework. In Section 3, we discuss the way VAEs are currently used to model speech/audio signals in the literature, and raise a set of related questions. In Section 4 we present the nonnegative representation and underlying signal statistical models as a general framework, of which NMF is a particular case, and we show how this framework also applies to VAE-based spectrogram modeling. Section 5 illustrates this discussion with some experiments on speech/audio analysis-synthesis with VAEs. Section 6 draws a series of conclusions and perspectives.

* This work is supported by a CIFRE PhD Grant funded by ANRT

2. VARIATIONAL AUTOENCODERS

As mentioned in the introduction, a VAE can be seen as a probabilistic autoencoder. In the original formulation of the seminal paper [2], a VAE delivers a parametric model of data distribution:

$$p_\theta(\mathbf{x}, \mathbf{z}) = p_\theta(\mathbf{x}|\mathbf{z})p_\theta(\mathbf{z}), \tag{1}$$

where $\mathbf{x} \in \mathbb{R}^F$ is a vector of observed data, $\mathbf{z} \in \mathbb{R}^L$ is a corresponding vector of latent data, with $L \ll F$, and θ denotes the set of distribution parameters. The likelihood function $p_\theta(\mathbf{x}|\mathbf{z})$ plays the role of a probabilistic decoder which models how the generation of observed data $\mathbf{x}$ is conditioned on the latent data $\mathbf{z}$. The prior distribution $p_\theta(\mathbf{z})$ is used to structure (or regularize) the latent space. Typically a standard Gaussian distribution is used: $p_\theta(\mathbf{z}) = \mathcal{N}(\mathbf{z}; \mathbf{0}, \mathbf{I}_L)$, where $\mathbf{I}_L$ is the identity matrix of size L. This encourages the latent coefficients to be orthogonal and with similar range. Note that this prior actually lacks parameters. The likelihood $p_\theta(\mathbf{x}|\mathbf{z})$ is usually defined as Gaussian:

$$p_\theta(\mathbf{x}|\mathbf{z}) = \mathcal{N}(\mathbf{x}; \boldsymbol{\mu}_\theta(\mathbf{z}), \boldsymbol{\sigma}^2_\theta(\mathbf{z})), \tag{2}$$

where $\mathcal{N}(\mathbf{x}; \boldsymbol{\mu}, \boldsymbol{\sigma}^2)$ denotes the probability density function (pdf) of the multivariate Gaussian distribution which is defined in the Appendix, and $\boldsymbol{\mu}_\theta(\mathbf{z}) \in \mathbb{R}^F$ and $\boldsymbol{\sigma}^2_\theta(\mathbf{z}) \in \mathbb{R}^F_+$ are the outputs of the decoder network. The parameter set θ is composed of the weights of this decoder network. Note that the entries of $\mathbf{x}$ are assumed independent as common in VAEs, so the vector $\boldsymbol{\sigma}^2_\theta(\mathbf{z})$ contains the diagonal coefficients of a diagonal covariance matrix.

The exact posterior distribution $p_\theta(\mathbf{z}|\mathbf{x})$ corresponding to the above model is intractable. It is approximated with a tractable parametric model $q_\phi(\mathbf{z}|\mathbf{x})$ that plays the role of the corresponding probabilistic encoder. This model generally has a form similar to the decoder:

$$q_\phi(\mathbf{z}|\mathbf{x}) = \mathcal{N}(\mathbf{z}; \tilde{\boldsymbol{\mu}}_\phi(\mathbf{x}), \tilde{\boldsymbol{\sigma}}^2_\phi(\mathbf{x})), \tag{3}$$

where $\tilde{\boldsymbol{\mu}}_\phi(\mathbf{x}) \in \mathbb{R}^L$ and $\tilde{\boldsymbol{\sigma}}^2_\phi(\mathbf{x}) \in \mathbb{R}^L_+$ are the outputs of the encoder network. The parameter set ϕ is composed of the weights of this encoder network. As before, $\tilde{\boldsymbol{\sigma}}^2_\phi(\mathbf{x})$ is a vector containing the diagonal entries of a diagonal covariance matrix.

Training of the VAE model, i.e. estimation of θ and ϕ, is made by optimizing a lower-bound of the marginal log-likelihood $\log p_\theta(\mathbf{x})$ computed from a large training dataset of vectors $\mathbf{x}$. It is shown in [2] that the marginal log-likelihood for an individual vector $\mathbf{x}$ writes:

$$\log p_\theta(\mathbf{x}) = d_{\text{KL}}(q_\phi(\mathbf{z}|\mathbf{x})|p_\theta(\mathbf{z}|\mathbf{x})) + \mathcal{L}(\phi, \theta, \mathbf{x}), \tag{4}$$

where $d_{\text{KL}} \geq 0$ denotes the Kullback-Leibler (KL) divergence and $\mathcal{L}(\phi, \theta, \mathbf{x})$ is the variational lower bound (VLB) given by:

$$\mathcal{L}(\phi, \theta, \mathbf{x}) = \underbrace{\mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}[\log p_\theta(\mathbf{x}|\mathbf{z})]}_{\text{reconstruction accuracy}} \underbrace{- d_{\text{KL}}(q_\phi(\mathbf{z}|\mathbf{x})|p_\theta(\mathbf{z}))}_{\text{regularization}}. \tag{5}$$

We can see that the VLB is the sum of two terms. The first term represents the average reconstruction accuracy. The second term acts as a regularizer encouraging the approximate posterior $q_\phi(\mathbf{z}|\mathbf{x})$ to be close to the prior $p_\theta(\mathbf{z})$. Since the expectation taken with respect to $q_\phi(\mathbf{z}|\mathbf{x})$ in the reconstruction accuracy term is analytically intractable, it is approximated using a Monte Carlo estimate with R samples $\mathbf{z}^{(r)}$ independently and identically drawn from $q_\phi(\mathbf{z}|\mathbf{x})$:

$$\mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}[\log p_\theta(\mathbf{x}|\mathbf{z})] \approx \frac{1}{R}\sum_{r=1}^{R} \log p_\theta(\mathbf{x}|\mathbf{z}^{(r)}). \tag{6}$$

In practice a training dataset $\mathbf{X} = \{\mathbf{x}_n\}_{n=1}^{N_{tr}}$ is used for the training of the VAE. Under the hypothesis of independent and identically distributed (i.i.d.) training vectors, the VAE training is done by maximizing the total VLB, which is the sum of individual VLBs over the training vectors. If we consider only one Monte Carlo sample per training vector (which is common practice provided that the batch size is sufficiently large [2]), or if we consider several Monte Carlo samples as additional training data, we can write the total VLB as:

$$\begin{aligned}\mathcal{L}(\phi, \theta, \mathbf{X}) = &\sum_{n=1}^{N_{tr}} \log p_\theta(\mathbf{x}_n|\mathbf{z}_n) \\ &- \sum_{n=1}^{N_{tr}} d_{\text{KL}}(q_\phi(\mathbf{z}_n|\mathbf{x}_n)|p_\theta(\mathbf{z}_n)).\end{aligned} \tag{7}$$

For the present case of Gaussian likelihood (2) and Gaussian encoding distribution (3), the VLB in (7) becomes:

$$\begin{aligned}\mathcal{L}(\phi, \theta, \mathbf{X}) = &-\sum_{n=1}^{N_{tr}}\sum_{f=0}^{F-1}\left(\log \sigma^2_{\theta,f}(\mathbf{z}_n) + \frac{(x_{fn} - \mu_{\theta,f}(\mathbf{z}_n))^2}{2\sigma^2_{\theta,f}(\mathbf{z}_n)}\right) \\ &+ \frac{1}{2}\sum_{n=1}^{N_{tr}}\sum_{l=1}^{L}\left(\log \tilde{\sigma}^2_{\phi,l}(\mathbf{x}_n) - \tilde{\mu}_{\phi,l}(\mathbf{x}_n)^2 - \tilde{\sigma}^2_{\phi,l}(\mathbf{x}_n)\right)\end{aligned} \tag{8}$$

where the subscript f or l denotes the f-th or l-th entry of a vector. Maximization of the total VLB is done by using the usual back-propagation technique and gradient-based optimization, which are not detailed in this paper. For more technical details that are not relevant here, the reader is referred to [2].

3. VAES FOR SPECTROGRAM MODELING: FACTS AND QUESTIONS

In this section, we analyze how VAEs are generally used for speech and audio spectrogram modeling in the recent literature. Although some of the points discussed below may seem trivial, they rise a series of fundamental questions that are poorly discussed in these papers and that we will address in the following.

3.1. Audio signal representation in the STFT domain

As shortly stated in the introduction, the processing is generally carried out in the STFT domain. Let $\mathbf{S} = [s_{fn}]_{f=0,n=1}^{F-1,N} \in \mathbb{C}^{F\times N}$ denote the STFT of a speech/audio signal, where f is the frequency bin index and n is the time frame index. Let $\mathbf{X} = [x_{fn}]_{f=0,n=1}^{F-1,N} \in \mathbb{R}_+^{F\times N}$ denote the corresponding real-valued and nonnegative *magnitude* or *power* spectrogram, i.e. $\mathbf{X} = |\mathbf{S}|$ or $\mathbf{X} = |\mathbf{S}|^2$, where $|.|$ and $.^2$ are to be understood as entry-wise operators. Note that we use the same notation as in the previous section on purpose, since the VAE modeling will precisely be applied on speech/audio spectrograms. Note also that $\mathbf{X} = |\mathbf{S}|^2$ is a sampled power spectrogram, aka a periodogram, i.e. an estimate of the power spectral density (PSD) $\mathbb{E}[|\mathbf{S}|^2]$ built from a single observation of the data in each time-frequency bin (and the same for the magnitude spectrogram).

3.2. Data representation, pre-processing and normalization

A VAE considers vectors as input and output. Hence an STFT spectrogram is processed as a sequence of successive spectral vectors $\mathbf{x}_n = [x_{fn}]_{f=0}^{F-1} \in \mathbb{R}_+^F$, each vector representing an STFT frame. Note that all x_{fn} are assumed independent across frequency bins and time frames, which is not to be confused with possible time-frequency structuration of the distribution parameters. An important practical question in VAEs is the choice of the audio STFT data representation. We did not observe any consensus in the literature.

For synthesis and transformation applications, e.g. [6], the observed/generated vector at time frame n generally corresponds to the short-term magnitude or power spectrum. There may be two explanations for that: (i) the original VAE formulation of [2] (i.e. the Gaussian models in (2) and (3)) considers real-valued and not complex-valued vectors, but in that case what about the non-negativity? and (ii) the magnitude or power spectrogram is the primary information used in the synthesis/transformation applications considered in the referenced papers (the phase spectrogram being processed separately).

For speech enhancement applications, the VAE speech model is generally plugged in a more general statistical framework including a noise model and a speech + noise mixture model, e.g. [9, 10]. In this framework, the original (real-valued) formulation of the VAE has been extended to model the complex-valued STFT vector $\mathbf{s}_n = [s_{fn}]_{f=0}^{F-1} \in \mathbb{C}^F$. This has been done by replacing the Gaussian distribution over real-valued vectors in (2) with the circularly symmetric complex Gaussian distribution that is widely used in speech enhancement and source separation probabilisitic methods [26, 27]. This important point is poorly commented in the referenced papers. Moreover, although $\mathbf{s}_n$ is here modeled by the VAE decoder, $\mathbf{x}_n$ as a short-term magnitude or power spectrum is still considered at the input of the encoder during VAE training.[1] The possible consequences (or absence of consequences) of this input/output mismatch are not discussed either. Note that here also, all s_{fn} are assumed independent across frequency bins and time frames, as is usually done in the speech enhancement and source separation literature.

It is important to note that in practice, the encoder input vector can contain magnitudes or squared magnitudes as discussed above, but also log-magnitudes as in [4], or actually any vector encoding a magnitude spectrum, possibly pre-processed and normalized in different manners. Normalization is a typical example of DNN-driven process, it has no theoretical justification from the signal processing point-of-view but it is known as helping a DNN training in general. So it is applied very frequently, and actually on purpose in VAEs. Also, the encoder input vector can be of different nature than the VAE decoder output vector, which is composed of probability distribution parameters; not to be confused with the output of the VAE as a generative model. Some of the output parameters may be homogeneous to the input data, e.g. mean vectors, and some others may not be, e.g. variance parameters. Moreover, data normalization can also be applied to output data, and the normalization/denormalization can be conducted in different manners at the input and at the output. Then, does data representation, pre-processing and normalization have any consequence on the theoretical foundations of the model?

[1] For speech enhancement applications, the encoder is only used for VAE training. During the speech signal inference process, only the decoder is used.

3.3. Statistical modeling and implications for VAE training

The choice of the reconstruction term of the loss function for the VAE training is often poorly discussed in papers dealing with VAE-based spectrogram modeling. A typical yet poorly justified approach could be: Let us choose a data representation that is appropriate for the considered application, for example a magnitude spectrum vector $\mathbf{x}_n$, and let us apply some normalization that is appropriate for DNNs. Then systematic application of the Gaussian model (2) is the easy way, leading to the weighted squared error form in the reconstruction term of (8). If we further set the variance parameters $\sigma^2_{\theta,f}(\mathbf{z}_n)$ to an arbitrarily fixed value σ^2 (i.e. we consider only the mean parameters $\mu_{\theta,f}(\mathbf{z}_n)$ as the free VAE outputs), then (8) becomes (up to an additive constant factor):

$$\mathcal{L}(\phi,\theta,\mathbf{X}) = -\frac{1}{\sigma^2}\sum_{n=1}^{N_{tr}}\sum_{f=0}^{F-1}\frac{1}{2}\big(x_{fn}-\mu_{\theta,f}(\mathbf{z}_n)\big)^2 + \frac{1}{2}\sum_{n=1}^{N_{tr}}\sum_{l=1}^{L}\Big(\log\tilde{\sigma}^2_{\phi,l}(\mathbf{x}_n) - \tilde{\mu}_{\phi,l}(\mathbf{x}_n)^2 - \tilde{\sigma}^2_{\phi,l}(\mathbf{x}_n)\Big) \quad (9)$$

This means that using the basic mean squared error (MSE) as the reconstruction term of the VAE loss function amounts to maximize the likelihood function under the present "fixed-variance free-mean" Gaussian model, hence providing some nice theoretical interpretation of the process. Yet this interpretation is poorly discussed in the papers. Does this approach have limitations? Does it make sense to model normalized magnitude vectors with a Gaussian distribution? Do other strategies exist? And what is the link with the problem of data representation?

As briefly mentioned in the introduction, a consistent theoretical framework exists that enables one to justify and interpret the choice of data representation, likelihood function and reconstruction term of the loss function, and how those points are related. This is what we present in the next section.

4. LINKING NMF AND VAE

In this section, we build on the existing statistical framework related to nonnegative representations, in particular Nonnegative Matrix Factorization (NMF), and its application to the modeling of speech/audio spectrograms. Most of the technical material presented here is extracted from [16] and [17]. We first shortly present the principle of NMF decomposition, then we go to the major point of this section which is to show that the underlying statistical framework directly applies to the VAE model, and can thus be used to give a solid theoretical interpretation of VAE-based modeling of speech/audio spectrograms. We finally report the three major NMF-based generative models considered in [16] and [17] and give their VAE counterparts.

4.1. The NMF model

NMF consists in modeling a matrix $\mathbf{V} = [v_{fn}]_{f,n} \in \mathbb{R}_+^{F\times N}$ of nonnegative entries as the product of two nonnegative matrices $\mathbf{W} = [w_{fk}]_{f,k} \in \mathbb{R}_+^{F\times K}$ and $\mathbf{H} = [h_{kn}]_{k,n} \in \mathbb{R}_+^{K\times N}$. In other words we have $\mathbf{V} \approx \widehat{\mathbf{V}} = \mathbf{WH}$, or equivalently $\widehat{v}_{fn} = (\mathbf{WH})_{fn} = \sum_{k=1}^{K} w_{fk}h_{kn}$. A low-rank approximation of $\mathbf{V}$, represented with a reduced number of parameters, is obtained by setting K such that $K(F+N) \ll FN$. In the speech/audio processing literature, $\widehat{\mathbf{V}}$ is typically used to model the signal ("true" or

"theoretical") PSD $\mathbb{E}[|\mathbf{S}|^2]$ based on the observed power spectrogram $\mathbf{X} = |\mathbf{S}|^2$ (or the same for the "true" magnitude spectrogram based on the observed magnitude spectrogram $\mathbf{X} = |\mathbf{S}|$). The interest of this approach is thus to provide a model of the signal PSD in each time-frequency bin with a very reasonable number of parameters (if K is chosen properly).

Calculating $\widehat{\mathbf{V}}$ from a given observed nonnegative matrix $\mathbf{X}$ is done by minimizing over $\mathbf{W}$ and $\mathbf{H}$ the following error under a non-negativity constraint:

$$D(\mathbf{X}|\widehat{\mathbf{V}}) = \sum_{n=1}^{N} \sum_{f=0}^{F-1} d(x_{fn}|\hat{v}_{fn}), \tag{10}$$

where $d(\cdot|\cdot)$ is a scalar divergence. The three most popular cost functions are the squared Euclidian distance $d_{\text{EUC}}(x|y) = 0.5(x-y)^2$, the generalized Kullback-Leibler (KL) divergence $d_{\text{KL}}(x|y) = x\log(x/y) - x + y$, and the Itakura-Saito (IS) divergence $d_{\text{IS}}(x|y) = x/y - \log(x/y) - 1$. For each of them, a set of algorithms have been proposed to solve the above minimization problem. Their presentation is out of the scope of this paper, where we focus on the link with the VAE and the underlying statistical models. For the same reason, we do not deal with the interpretation of NMF as a model of composite signals [16, 17], which is of primary importance in the source separation literature.

4.2. Linking NMF- and VAE-based spectrogram modeling

Now the major point of the present paper is the following: *The minimization of the global cost function* (10)*, the choice of the scalar cost function in* (10)*, the choice of data representation, and the interpretation in terms of underlying statistical model are problems that are all common to NMF and VAE.* In other words, *a common framework exists where* $\widehat{\mathbf{V}}$ *may as well be an NMF model* $\widehat{\mathbf{V}} = \mathbf{WH}$ *or the concatenation of successive (nonnegative) output vectors of a VAE*, e.g. $\widehat{\mathbf{V}} = [\boldsymbol{\sigma}_\theta^2(\mathbf{z}_1), \boldsymbol{\sigma}_\theta^2(\mathbf{z}_2), \cdots, \boldsymbol{\sigma}_\theta^2(\mathbf{z}_N)]$, which is the case for VAE-based spectrogram modeling. Indeed, as will be detailed below, for both NMF and VAE models, (10) is nothing but a reformulation of the negative log-likelihood function of the underlying generative model. More specifically, if $\widehat{\mathbf{V}}$ is the output of a VAE, the reconstruction accuracy in (7) and the cost function (10) are identical up to a constant multiplicative positive factor α, sign, and a constant additive factor. In short, (7) can be rewritten as:

$$\begin{aligned}\mathcal{L}(\phi, \theta, \mathbf{X}) = -\alpha &\sum_{n=1}^{N_{tr}} \sum_{f=0}^{F-1} d(x_{fn}|\hat{v}_{fn}) \\ &- \sum_{n=1}^{N_{tr}} d_{\text{KL}}(q_\phi(\mathbf{z}_n|\mathbf{x}_n)|p_\theta(\mathbf{z}_n)).\end{aligned} \tag{11}$$

In the VAE model framework, minimization of (10) *thus amounts to optimal estimation of the VAE parameters in the maximum-likelihood (ML) sense*. Let us temper a bit: (10) only concerns the VAE decoder, and the complete VAE is actually optimized by maximizing (7) (or (11)), i.e. the combination of (10) with the VLB regularization term. This latter is important to differentiate a VAE from a deterministic AE. Let us note that in the VAE framework, ML estimation of $\widehat{\mathbf{V}}$ is to be understood as a shortcut for ML estimation of θ, the decoder parameters, which requires the joint estimation of the encoder parameters ϕ during the VAE training. Finally, let us also note that α plays the role of balancing factor between reconstruction and regularization, and quite interestingly, it is very similar to the β factor of the β-VAE model proposed in [28] in an ad-hoc manner, for the same aim (though β is applied to the regularization term instead of the reconstruction term).

Although all these points may sound trivial to readers familiar with the statistical interpretation of NMF spectrogram modeling, to our knowledge they have never been pointed out in the literature on VAE-based speech/audio processing. One reasonable explanation for this may be that NMF studies often start with the cost function formulated as (10), and the interpretation in terms of underlying generative model comes in second (when it comes), whereas VAE studies start with a generative model then go to the cost function formulated as (7).

4.3. Practical cases

We now apply the above considerations to the three major cases considered in [16] and [17], which correspond to different divergences $d(\cdot|\cdot)$ in (10) and (11).

Euclidian distance case In the NMF context, it has been shown in [16, 17] that choosing and minimizing the squared Euclidian distance between $\mathbf{X}$ and $\widehat{\mathbf{V}} = \mathbf{WH}$ corresponds to ML estimation of $\mathbf{W}$ and $\mathbf{H}$ under the assumption of the Gaussian model

$$x_{fn} \sim \mathcal{N}(x_{fn}; \widehat{v}_{fn}, \sigma^2), \tag{12}$$

with $\widehat{v}_{fn} = (\mathbf{WH})_{fn} = \sum_{k=1}^{K} w_{fk}h_{kn}$. Similarly, in the VAE case, choosing and minimizing the squared Euclidian distance between x_{fn} and $\widehat{v}_{fn}$ in (11), with $\widehat{v}_{fn} = \mu_{\theta,f}(\mathbf{z}_n)$, corresponds to ML estimation of $\widehat{v}_{fn}$ under the assumption of the Gaussian model (2), with a fixed variance $\sigma^2_{\theta,f}(\mathbf{z}_n) = \sigma^2, \forall(f,n)$. Actually this is what we have already done at the end of Section 3, and formalized in (9). In both NMF and VAE cases, we have the following underlying model:

$$x_{fn} = \widehat{v}_{fn} + e_{fn}, \tag{13}$$

where e_{fn} is an i.i.d. additive white Gaussian noise, i.e. $e_{fn} \overset{i.i.d.}{\sim} \mathcal{N}(0, \sigma^2)$. Moreover, identifying (11) and (9) leads to $\alpha = 1/\sigma^2$, hence σ^2 plays the role of balancing factor between reconstruction and regularization.

A Gaussian model is often favored because of its generality and its nice features in mathematical derivations. For instance, it has been used for VAE-based speech spectrogram modeling in [4, 6, 11, 29]. However, although this approach could work quite well in many settings, it suffers from what is referred to as an interpretation ambiguity in [16]: Although x_{fn} represents a magnitude or power spectrum, $\mathcal{N}(x_{fn}; \mu_{\theta,f}(\mathbf{z}_n), \sigma^2)$ may produce negative data (even if we somehow enforce $\mu_{\theta,f}(\mathbf{z}_n) \geq 0$). This problem may be partly fixed by appropriate data normalization (e.g. min-max rescaling within $[-1, 1]$) and/or with log-scaling. However, it is subject to discussion if the distribution of log-magnitude spectra of real-world speech and audio signals has a Gaussian shape or not.

Itakura-Saito divergence case Alternately, it was shown and largely discussed in [17] that using the IS divergence in (10) corresponds to maximizing the log-likelihood function under the assumption of a Gamma distribution for x_{fn}. More precisely, the statistical model is:

$$x_{fn} \sim \mathcal{G}(x_{fn}; \alpha, \alpha/\widehat{v}_{fn}), \tag{14}$$

where $\mathcal{G}(.;a,b)$ is the Gamma distribution with shape parameter $a>0$ and rate parameter $b>0$, and whose pdf is defined in the Appendix. In the NMF framework we have $\widehat{v}_{fn}=(\mathbf{WH})_{fn}=\sum_{k=1}^{K}w_{fk}h_{kn}$, but this result is still valid in the VAE framework where we now have $\widehat{v}_{fn}=\sigma^2_{\theta,f}(\mathbf{z}_n)$. In both NMF and VAE cases, we have the following underlying model:

$$x_{fn}=\widehat{v}_{fn}e_{fn}, \tag{15}$$

where e_{fn} is an i.i.d. multiplicative Gamma noise, i.e. $e_{fn}\overset{i.i.d.}{\sim}\mathcal{G}(e_{fn};\alpha,\alpha)$.

Importantly, it was also shown in [17] that if x_{fn} corresponds to a linear-scale squared magnitude, minimizing the IS divergence corresponds to ML estimation of $\widehat{v}_{fn}$ under a circularly symmetric complex Gaussian model for the STFT coefficients $s_{fn}\in\mathbb{C}$ corresponding to $x_{fn}=|s_{fn}|^2\in\mathbb{R}_+$, with a variance $\mathbb{E}[|s_{fn}|^2]$ equal to $\widehat{v}_{fn}$. In short, $s_{fn}\sim\mathcal{N}_c(s_{fn};0,\widehat{v}_{fn})$, where the pdf of the complex Gaussian distribution $\mathcal{N}_c$ is defined in the Appendix. This interpretation is quite important since this model and associated ML fitting procedure have been used extensively in speech enhancement and speech/audio source separation, in combination with NMF, e.g. [19, 21, 23], or not, e.g. [26, 20, 30]. Indeed, in such applications, we are interested in inferring the complex-valued source STFT coefficients s_{fn} from corrupted observations. Again, this result is valid for both NMF and VAE frameworks: In IS-based NMF, we have $\mathbb{E}[|s_{fn}|^2]=\mathbb{E}[x_{fn}]=\widehat{v}_{fn}=\sum_{k=1}^{K}w_{fk}h_{kn}$. In IS-based VAE, we have $\mathbb{E}[|s_{fn}|^2]=\mathbb{E}[x_{fn}]=\widehat{v}_{fn}=\sigma^2_{\theta,f}(\mathbf{z}_n)$ and the mean parameters $\mu_{\theta,f}(\mathbf{z}_n)$ are simply disregarded since (2) is implicitly replaced with the above Gamma model of x_{fn}. Note that IS-VAE was shown to outperform IS-NMF for speech enhancement in [10].

Generalized Kullback-Leibler divergence case Finally, minimizing the KL divergence between x_{fn} and $\widehat{v}_{fn}$ corresponds to ML estimation of $\widehat{v}_{fn}$ under the assumption of a Poisson distribution for x_{fn}:

$$x_{fn}\sim\mathcal{P}(x_{fn};\widehat{v}_{fn}), \tag{16}$$

where $\mathcal{P}(.;\lambda)$ is the Poisson distribution with scale parameter $\lambda>0$ and whose pdf is defined in the Appendix. Note that there is here no equivalent model in terms of additive or multiplicative noise. In theory, the Poisson distribution is defined for nonnegative integer-valued random variables, but this issue can be fixed by considering high-resolution fixed-point quantization of the spectrograms. As above, this result is valid for both NMF and VAE models. Here, $\widehat{v}_{fn}$ plays the role of a scale parameter, hence in principle the output of a KL-based VAE is a vector of scale parameters $\widehat{v}_{fn}=\sigma_{\theta,f}(\mathbf{z}_n)$ for $f=0,...,F-1$. Although, as stated above, arbitrary normalization and corresponding denormalization can be applied. Historically, KL-based NMF has been applied on (linear-scale) magnitude spectra instead of power spectra, see the seminal papers [31, 32], but in fact there is no underlying model on the complex-valued STFT coefficients s_{fn} to support this principle. In other words, in most papers on KL-based NMF, $\widehat{v}_{fn}$ is a scale parameter over magnitude spectra, because x_{fn} is a magnitude spectra, but it could as well be a scale parameter over a different representation. Of course, the same remark applies to a KL-based VAE.

In summary, in the speech/audio spectrogram NMF modeling framework, we had:

- EUC-NMF: $p_\theta(\mathbf{X}|\mathbf{Z})=\prod_{f,n}\mathcal{N}(x_{fn};(\mathbf{WH})_{fn},\sigma^2)$;
- IS-NMF: $p_\theta(\mathbf{X}|\mathbf{Z})=\prod_{f,n}\mathcal{G}(x_{fn};\alpha,\alpha/(\mathbf{WH})_{fn})$ and $p_\theta(\mathbf{S}|\mathbf{Z})=\prod_{f,n}\mathcal{N}_c(s_{fn};0,(\mathbf{WH})_{fn})$ with $x_{fn}=|s_{fn}|^2$;
- KL-NMF: $p_\theta(\mathbf{X}|\mathbf{Z})=\prod_{f,n}\mathcal{P}(x_{fn};(\mathbf{WH})_{fn})$.

In the VAE framework we have:

- EUC-VAE: $p_\theta(\mathbf{X}|\mathbf{Z})=\prod_{f,n}\mathcal{N}(x_{fn};\mu_{\theta,f}(\mathbf{z}_n),\sigma^2)$;
- IS-VAE: $p_\theta(\mathbf{X}|\mathbf{Z})=\prod_{f,n}\mathcal{G}(x_{fn};\alpha,\alpha/\sigma^2_{\theta,f}(\mathbf{z}_n))$ and $p_\theta(\mathbf{S}|\mathbf{Z})=\prod_{f,n}\mathcal{N}_c(s_{fn};0,\sigma^2_{\theta,f}(\mathbf{z}_n))$ with $x_{fn}=|s_{fn}|^2$;
- KL-VAE: $p_\theta(\mathbf{X}|\mathbf{Z})=\prod_{f,n}\mathcal{P}(x_{fn};\sigma_{\theta,f}(\mathbf{z}_n))$.

4.4. A practical note on the implementation of the VAE loss function

The above considerations have a practical consequence in the coding of the loss function when implementing a VAE with a deep learning library. Indeed, in practice, as stated above, input/output data are often pre-processed (e.g. log-scaled) and/or normalized to facilitate the VAE training. For the statistical interpretation considered in this paper to hold, the reconstruction term of the VAE loss function, as implemented in a deep learning toolkit, must have the form of the log-likelihood function $\log p_\theta(\mathbf{x}|\mathbf{z})$, and the data used in this loss function must be consistent with the model, i.e. if they have been previously normalized, then *they must be denormalized*. Using the normalized data would break the consistency of the underlying statistical model.

Let us give an example, by considering the Gamma model in (14) for the squared STFT magnitudes $x_{fn}=|s_{fn}|^2$. This model *implies that we have to* use the IS divergence in the reconstruction term of the loss function in (11). At training time, the VAE is fed with pre-processed/normalized data $x_{fn}^{\text{norm}}=g(x_{fn})$ and it provides pre-processed/normalized scale parameters $\widehat{v}_{fn}^{\text{norm}}=\tilde{g}(\widehat{v}_{fn})$. Note that the pre-processing/normalization of data and parameters may be different, as denoted by the different $g(\cdot)$ and $\tilde{g}(\cdot)$ functions. Then the implementation of the reconstruction term of the loss function based on the IS divergence and "applied to" x_{fn}^{norm} and $\widehat{v}_{fn}^{\text{norm}}$ should be of the form:

$$\frac{g^{-1}(x_{fn}^{\text{norm}})}{\tilde{g}^{-1}(\widehat{v}_{fn}^{\text{norm}})}-\log\frac{g^{-1}(x_{fn}^{\text{norm}})}{\tilde{g}^{-1}(\widehat{v}_{fn}^{\text{norm}})}-1=d_{\text{IS}}(x_{fn}|\widehat{v}_{fn}). \tag{17}$$

The denormalized outputs $\widehat{v}_{fn}=\tilde{g}^{-1}(\widehat{v}_{fn}^{\text{norm}})$ are then "automatically" homogeneous to scale parameters. In contrast, using directly the normalized values in the above reconstruction term (i.e. calculating $d_{\text{IS}}(x_{fn}^{\text{norm}}|\widehat{v}_{fn}^{\text{norm}})$) or using another distance (e.g. the MSE) on either the normalized or denormalized data would not be consistent with the Gamma model considered in this example.

5. EXPERIMENTS

In this section, we briefly present the results of experiments that were conducted to illustrate our discussion. We processed VAE-based analysis-synthesis of sound spectrograms for the three cases described in Section 4. Waveform resynthesis was done by combining the output magnitude spectrogram with the phase spectrogram of the original signal. We applied this on speech signals (TIMIT dataset [33], 10 utterances $\times$ 462 speakers in the training set, for a total of about 4h, and 10 different utterances $\times$ 168 different speakers in the test set, for a total of about 1.5h) and music

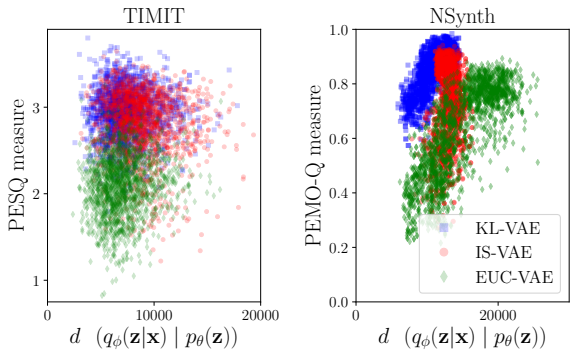


Figure 1: Audio quality as a function of the regularization term of (5).

signals (a subset of the large NSynth dataset [34], 88 notes with 4 different velocities from 17 instruments for the training set and 3 instruments for the test set all from the acoustic keyboards family, for a total of 9h of signals) at a 16 kHz sampling rate. The STFT was computed using a 64-ms sine window ($F = 513$) and a 75% overlap.

The VAE decoder network contains three layers of size [64, 128, 513] and the encoder network is the symmetric. Both networks use $\tanh$ and identity activation functions for the hidden and output layers respectively. The output of the encoder and decoder networks are thus real-valued, and as proposed in the original paper on VAEs [2], we output the logarithm of variance/scale parameters for the IS-VAE and KL-VAE cases. At the input of the encoder, we provide either magnitude spectrograms (KL-VAE and EUC-VAE) or power spectrograms (IS-VAE).

The results are plotted in Fig. 1. In order to measure the quality of the reconstructed signal independently of the nature of the cost function, PESQ scores [35] (for speech) and PEMO-Q scores [36] (for music) were calculated on the resynthesized signals in the test set. These scores are plotted in Fig. 1 as a function of the regularization term of (5). Each point represents either a utterance (left) or a music note (right) from the dataset. We set $\alpha = 0.1$ in (7) for the IS-VAE, and $\alpha = 1$ for both EUC-VAE and KL-VAE. This was to ensure (i) to keep a sufficiently small regularization term in the loss function so that VAEs are not turning into a deterministic autoencoders, and (ii) to obtain the same range of regularization term values for the 3 cost functions, so that the performance can be fairly compared in terms of reconstruction quality. We can see in Fig. 1 that for music signals (PEMO-Q scores) KL-VAE globally performs the best, followed by IS-VAE (with an overlapping zone of equal performance). For speech signals (PESQ scores), KL-VAE and IS-VAE are providing similar results. EUC-VAE generally provides lower scores.

6. CONCLUSION

We can now draw the following conclusions:

- The three presented cost functions usable for NMF or VAE modeling all correspond to an underlying statistical model of processed spectrogram $\mathbf{X} = [\mathbf{x}_n]_{n=1}^N$. For all three cases, training the VAE with data $\mathbf{X}$ corresponds to ML estimation of VAE decoder parameters under the corresponding statistical model of $\mathbf{X}$.
- Among these three cases, only one (IS-case) has an underlying statistical model of the speech/audio signal STFT coefficient s_{fn} (circularly symmetric complex Gaussian), which has proven to be of great interest for speech enhancement and source separation applications.
- The reconstruction accuracy and regularization of the VAE can be weighted using the α factor in (11). For EUC-VAE and IS-VAE this factor is naturally emerging as a parameter of the underlying statistical model, which provides a nice alternative (or interpretation) to the ad-hoc definition of the similar β factor introduced in [28]. This is not the case for KL-VAE, where $\alpha = 1$. For the interpretation of IS-VAE in terms of complex Gaussian model on s_{fn} to hold, we must also have $\alpha = 1$.
- In our experiments, KL-VAE and IS-VAE perform better than EUC-VAE according to perceptually-motivated objective measures.
- Although we necessarily presented this extension in the context of nonnegative representations, VAEs are not limited to nonnegative data. They can be applied to any real-valued data. This is what is done when processing log-scale spectrograms such as in [4]. The IS and KL divergences and associated Gamma and Poisson models are limited to nonnegative data, but the Euclidian distance and associated Gaussian model are not.
- In practice, input/output data are often pre-processed and/or normalized. If the pre-processed/normalized data are used in the VAE practical implementation, then the loss function should include denormalization and inverse pre-processing.
- All the points considered in this paper are valid for recurrent VAEs [37], which are likely to become popular in speech/audio processing as well. Also, generalization of NMF to more general divergences and corresponding statistical interpretation exist, e.g. [38, 39]. It is likely to be relevant for VAEs.

We spent time and effort to understand the correct form that a VAE loss function should have in a deep learning library to be consistent with a sounded signal statistical model. We believe that sharing the content of this paper (and code if the paper is accepted) with the speech/audio processing community can help colleagues to take VAEs into hand faster and in a principled manner. Also, we believe that the bridge we built in this paper can benefit to both the speech enhancement / source separation community and the musical sound processing community.

A. PROBABILITY DISTRIBUTIONS

A.1. Gaussian distributions

Let $\mathcal{N}(x; \mu, \sigma^2)$ denote the Gaussian distribution for a random variable $x \in \mathbb{R}$ with mean $\mu \in \mathbb{R}$ and variance $\sigma^2 \in \mathbb{R}_+$. Its probability density function (pdf) is defined by:

$$\mathcal{N}(x; \mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right). \qquad (18)$$

Note that for simplicity we use the same notation to denote a probability distribution and its pdf.

Let $\mathcal{N}(\mathbf{x}; \boldsymbol{\mu}, \boldsymbol{\sigma}^2)$ denote the multivariate Gaussian distribution for a real-valued random vector $\mathbf{x} \in \mathbb{R}^F$ of mean vector $\boldsymbol{\mu} \in \mathbb{R}^F$,

and with statistically independent entries such that $\boldsymbol{\sigma}^2 \in \mathbb{R}_+^F$ is the vector of variances (covariance terms are zero and thus omitted in the parametrization for simplicity). Its pdf is therefore equal to the product of univariate Gaussian pdfs:

$$\mathcal{N}(\mathbf{x}; \boldsymbol{\mu}, \boldsymbol{\sigma}^2) = \prod_{f=0}^{F-1} \mathcal{N}(x_f; \mu_f, \sigma_f^2), \tag{19}$$

where v_f denotes the f-th entry of a vector $\mathbf{v}$.

Let $\mathcal{N}_c(x; \mu, \sigma^2)$ denote the proper complex Gaussian distribution for a random variable $x \in \mathbb{C}$ with mean $\mu \in \mathbb{C}$ and variance $\sigma^2 \in \mathbb{R}_+$. Its pdf is defined by:

$$\mathcal{N}_c(x; \mu, \sigma^2) = \frac{1}{\pi\sigma^2} \exp\left(-\frac{|x-\mu|^2}{\sigma^2}\right). \tag{20}$$

This distribution is circularly symmetric (i.e. invariant to a phase shift for x) if $\mu = 0$

A.2. Gamma distribution

Let $\mathcal{G}(x; a, b)$ denote the Gamma distribution for a random variable $x \in \mathbb{R}_+$ with shape and rate parameters $a > 0$ and $b > 0$ respectively. Its pdf is defined by:

$$\mathcal{G}(x; a, b) = \frac{b^a}{\Gamma(a)} x^{a-1} \exp(-bx), \tag{21}$$

where $\Gamma(\cdot)$ is the Gamma function.

A.3. Poisson distribution

Let $\mathcal{P}(x; \lambda)$ denote the Poisson distribution for a random variable $x \in \mathbb{N}$ with rate parameter $\lambda > 0$. Its pdf is defined by:

$$\mathcal{P}(x; \lambda) = \exp(-\lambda)\frac{\lambda^x}{x!}. \tag{22}$$

7. REFERENCES

[1] P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, and P.-A. Manzagol, "Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion," *Journal of Machine Learning Research*, vol. 11, no. Dec, pp. 3371–3408, 2010.

[2] D.P. Kingma and M. Welling, "Auto-encoding variational Bayes," in *Int. Conf. Learning Representations (ICLR)*, 2014.

[3] M. Blaauw and J. Bonada, "Modeling and transforming speech using variational autoencoders.," in *Conf. of the Int. Speech Comm. Association (Interspeech)*, San Francisco, CA, 2016.

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

[5] W.-N. Hsu, Y. Zhang, and J. Glass, "Learning latent representations for speech generation and transformation," in *Conf. of the Int. Speech Comm. Association (Interspeech)*, Stockholm, Sweden, 2017.

[6] K. Akuzawa, Y. Iwasawa, and Y. Matsuo, "Expressive speech synthesis via modeling expressions with variational autoencoder," in *Conf. of the Int. Speech Comm. Association (Interspeech)*, Hyderabad, India, 2018.

[7] F. Roche, T. Hueber, S. Limier, and L. Girin, "Autoencoders for music sound modeling: A comparison of linear, shallow, deep, recurrent and variational models," in *Sound and Music Computing Conference (SMC)*, Málaga, Spain, 2019.

[8] P. Esling, A. Chemla-Romeu-Santos, and A. Bitton, "Bridging audio analysis, perception and synthesis with perceptually-regularized variational timbre spaces," in *Int. Society for Music Information Retrieval Conf. (ISMIR)*, Paris, France, 2018.

[9] Y. Bando, M. Mimura, K. Itoyama, K. Yoshii, and T. Kawahara, "Statistical speech enhancement based on probabilistic integration of variational autoencoder and non-negative matrix factorization," in *IEEE Int. Conf. on Acoustics, Speech and Signal Process. (ICASSP)*, Calgary, Canada, 2018.

[10] S. Leglaive, L. Girin, and R. Horaud, "A variance modeling framework based on variational autoencoders for speech enhancement," in *IEEE Int. Workshop on Machine Learning for Signal Processing (MLSP)*, Aalborg, Denmark, 2018.

[11] L. Pandey, A. Kumar, and V. Namboodiri, "Monaural audio source separation using variational autoencoders," in *Conf. of the Int. Speech Comm. Association (Interspeech)*, Hyderabad, India, 2018.

[12] S. Leglaive, U. Şimşekli, A. Liutkus, L. Girin, and R. Horaud, "Speech enhancement with variational autoencoders and alpha-stable distributions," in *IEEE Int. Conf. on Acoustics, Speech and Signal Process. (ICASSP)*, 2019.

[13] K. Sekiguchi, Y. Bando, K. Yoshii, and T. Kawahara, "Bayesian multichannel speech enhancement with a deep speech prior," in *Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA)*, 2018, pp. 1233–1239.

[14] L. Li, H. Kameoka, and S. Makino, "Fast MVAE: Joint separation and classification of mixed sources based on multichannel variational autoencoder with auxiliary classifier," in *IEEE Int. Conf. on Acoustics, Speech and Signal Process. (ICASSP)*, Brighton, UK, 2019.

[15] S. Leglaive, L. Girin, and R. Horaud, "Semi-supervised multichannel speech enhancement with variational autoencoders and non-negative matrix factorization," in *IEEE Int. Conf. on Acoustics, Speech and Signal Process. (ICASSP)*, Brighton, UK, 2019.

[16] C. Févotte and A.T. Cemgil, "Nonnegative matrix factorizations as probabilistic inference in composite models," in *European Signal Processing Conference (EUSIPCO)*, Glasgow, Scotland, 2009.

[17] C. Févotte, N. Bertin, and J.-L. Durrieu, "Nonnegative matrix factorization with the itakura-saito divergence: With application to music analysis," *Neural computation*, vol. 21, no. 3, pp. 793–830, 2009.

[18] D.D. Lee and H.S. Seung, "Learning the parts of objects by non-negative matrix factorization," *Nature*, vol. 401, pp. 788–791, 1999.

[19] A. Ozerov and C. Févotte, "Multichannel nonnegative matrix factorization in convolutive mixtures for audio source separation," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 18, no. 3, pp. 550–563, 2010.

[20] N.Q. Duong, E. Vincent, and R. Gribonval, "Under-determined reverberant audio source separation using a full-rank spatial covariance model," *IEEE Trans. Audio, Speech, Language Process.*, vol. 18, no. 7, pp. 1830–1840, 2010.

[21] T. Gerber, M. Dutasta, L. Girin, and C. Févotte, "Professionally-produced music separation guided by covers," in *Int. Society for Music Information Retrieval Conf. (ISMIR)*, Porto, Portugal, 2012.

[22] P. Smaragdis, C. Fevotte, G. Mysore, N. Mohammadiha, and M. Hoffman, "Static and dynamic source separation using nonnegative factorizations: A unified view," *IEEE Signal Processing Magazine*, vol. 31, no. 3, pp. 66–75, 2014.

[23] D. Kounades-Bastian, L. Girin, X. Alameda-Pineda, S. Gannot, and R. Horaud, "A variational EM algorithm for the separation of moving sound sources," in *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, New Paltz, NJ, USA, 2015.

[24] S. Leglaive, R. Badeau, and G. Richard, "Multichannel audio source separation with probabilistic reverberation priors," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 24, no. 12, pp. 2453–2465, 2016.

[25] S. Leglaive, R. Badeau, and G. Richard, "Student's t Source and Mixing Models for Multichannel Audio Source Separation," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 26, no. 6, pp. 1150–1164, 2018.

[26] Y. Ephraim and D. Malah, "Speech enhancement using a minimum-mean square error short-time spectral amplitude estimator," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 32, no. 6, pp. 1109–1121, 1984.

[27] A. Liutkus, R. Badeau, and G. Richard, "Gaussian processes for underdetermined source separation," *IEEE Transactions on Signal Processing*, vol. 59, no. 7, pp. 3155–3167, 2011.

[28] I. Higgins, L. Matthey, A. Pal, C. Burgess, X. Glorot, M. Botvinick, S. Mohamed, and A. Lerchner, "β-vae: learning basic visual concepts with a constrained variational framework.," in *Int. Conf. on Learning Representations (ICLR)*, 2017.

[29] Y. Jung, Y. Kim, Y. Choi, and H. Kim, "Joint learning using denoising variational autoencoders for voice activity detection," in *Conf. of the Int. Speech Comm. Association (Interspeech)*, Hyderabad, India, 2018.

[30] X. Li, L. Girin, and R. Horaud, "An EM algorithm for audio source separation based on the convolutive transfer function," in *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, New Paltz, NJ, USA, 2017.

[31] P. Smaragdis and J.C. Brown, "Non-negative matrix factorization for polyphonic music transcription," in *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, New Paltz, NJ, 2003.

[32] T. Virtanen, "Monaural sound source separation by nonnegative matrix factorization with temporal continuity and sparseness criteria," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 3, pp. 1066–1074, 2007.

[33] J.S. Garofolo, L.F. Lamel, W.M. Fisher, J.G. Fiscus, D.S. Pallett, N.L. Dahlgren, and V. Zue, "TIMIT acoustic phonetic continuous speech corpus," in *Linguistic data consortium*, 1993.

[34] J. Engel, C. Resnick, A. Roberts, S. Dieleman, D. Eck, K. Simonyan, and M. Norouzi, "Neural audio synthesis of musical notes with wavenet autoencoders," *arXiv preprint arXiv:1704.01279*, 2017.

[35] A.W. Rix, J.G. Beerends, M.P. Hollier, and A.P. Hekstra, "Perceptual evaluation of speech quality (PESQ): A new method for speech quality assessment of telephone networks and codecs," in *IEEE Int. Conf. on Acoustics, Speech and Signal Process. (ICASSP)*, 2001, pp. 749–752.

[36] R. Huber and B. Kollmeier, "PEMO-Q: A new method for objective audio quality assessment using a model of auditory perception," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 14, no. 6, pp. 1902–1911, 2006.

[37] J. Chung, K. Kastner, L. Dinh, K. Goel, A. C. Courville, and Y. Bengio, "A recurrent latent variable model for sequential data," in *Advances in Neural Information Processing Systems*, 2015, pp. 2980–2988.

[38] S. Sra and I.S. Dhillon, "Generalized nonnegative matrix approximations with bregman divergences," in *Advances in Neural Information Processing Systems*, 2006, pp. 283–290.

[39] A. Cichocki, S. Cruces, and S. Amari, "Generalized alpha-beta divergences and their application to robust nonnegative matrix factorization," *Entropy*, vol. 13, no. 1, pp. 134–170, 2011.

Bibliography

Adrien, J.-M. (1991). "Representations of musical signals". In: ed. by Giovanni De Poli, Aldo Piccialli, and Curtis Roads. Cambridge, MA, USA: MIT Press. Chap. The missing link: Modal synthesis, pp. 269–298 (cit. on p. 17).

Akuzawa, K., Y. Iwasawa, and Y. Matsuo (2018). "Expressive speech synthesis via modeling expressions with variational autoencoder". In: *Proceedings of the Conference of the International Speech Communication Association (Interspeech)*. Hyderabad, India, pp. 3067–3071 (cit. on pp. 4, 24, 68).

Allen, J.B. (1977). "Short-term spectral analysis, synthesis, and modification by discrete Fourier transform". In: *IEEE Transactions on Acoustics, Speech, and Signal Processing* 25.3, pp. 235–238 (cit. on p. 71).

American National Standards Institute (1973). *American National Standard psychoacoustical terminology*. New York, NY, USA: American National Standards Institute (cit. on p. 6).

Aramaki, M., M. Besson, R. Kronland-Martinet, and S. Ystad (2010). "Controlling the perceived material in an impact sound synthesizer". In: *IEEE/ACM Transactions on Audio, Speech, and Language Processing (TASLP)* 19.2, pp. 301–314 (cit. on p. 2).

Arthur, D. and S. Vassilvitskii (2007). "k-means++: The advantages of careful seeding". In: *Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms*. Society for Industrial and Applied Mathematics. New Orleans, LA, USA, pp. 1027–1035 (cit. on p. 31).

Audio Engineering Society (AES) (2017). "Loudness guidelines for OTT and OVD Content". In: *Technical Document AESTD1006.1.17-10* (cit. on p. 106).

Balazs, P., M. Dörfler, F. Jaillet, N. Holighaus, and G. Velasco (2011). "Theory, implementation and applications of nonstationary Gabor frames". In: *Journal of computational and applied mathematics* 236.6, pp. 1481–1496 (cit. on p. 68).

Bando, Y., M. Mimura, K. Itoyama, K. Yoshii, and T. Kawahara (2018). "Statistical speech enhancement based on probabilistic integration of variational autoencoder and non-negative matrix factorization". In: *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. Calgary, Canada (cit. on pp. 24, 69).

Basaran, D., S. Essid, and G. Peeters (2018). "Main melody extraction with source-filter NMF and CRNN". In: *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*. Paris, France (cit. on p. 23).

Bellemare, M. and C. Traube (2005). "Verbal description of piano timbre according to advanced performers". In: *Proceedings of the European Society for the Cognitive Sciences of Music (ESCOM2005)* (cit. on p. 40).

Bengio, Y., P. Simard, and P. Frasconi (1994). "Learning long-term dependencies with gradient descent is difficult". In: *IEEE Transactions on Neural Networks* 5.2, pp. 157–166 (cit. on p. 63).

Bengio, Y., P. Lamblin, D. Popovici, and H. Larochelle (2007). "Greedy layer-wise training of deep networks". In: *Advances in Neural Information Processing Systems (NIPS)*. Vancouver, Canada (cit. on pp. 63, 97).

Bensa, J., D. Dubois, R. Kronland-Martinet, and S. Ystad (2004). "Perceptive and cognitive evaluation of a piano synthesis model". In: *Proceedings of the International Symposium on Computer Music Modeling and Retrieval (CMMR)*. Springer. Esbjerg, Denmark, pp. 232–245 (cit. on pp. 10, 12).

Bishop, C. (2006). *Pattern recognition and machine learning*. Springer (cit. on pp. 18, 19, 62).

Blaauw, M. and J. Bonada (2016). "Modeling and transforming speech using variational autoencoders". In: *Proceedings of the Conference of the International Speech Communication Association (Interspeech)*. San Francisco, CA (cit. on pp. 24, 61, 66, 68).

Caclin, A., S. McAdams, B. Smith, and S. Winsberg (2005). "Acoustic correlates of timbre space dimensions: A confirmatory study using synthetic tones". In: *The Journal of the Acoustical Society of America* 118.1, pp. 471–482 (cit. on p. 6).

Cance, C. and D. Dubois (2015). "Dire notre expérience du sonore : nomination et référenciation". In: *Langue Française* 4, pp. 15–32 (cit. on pp. 10, 12).

Castellengo, M. (2015). *Ecoute musicale et acoustique : avec 420 sons et leurs sonagrammes décryptés*. Eyrolles (cit. on pp. 6, 10, 29, 40, 42).

Castellengo, M. and D. Dubois (2005). "Timbre or timbres? Property of the signal, the instrument, or cognitive construction?" In: *Proceedings of the Conference on Interdisciplinary Musicology (CIM05)*. Montréal, Canada (cit. on pp. 6, 28).

Chandna, P., M. Blaauw, J. Bonada, and E. Gómez (2019). "A vocoder based method for singing voice extraction". In: *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE. Brighton, UK, pp. 990–994 (cit. on p. 23).

Chapelle, O. and A. Zien (2005). "Semi-supervised classification by low density separation". In: *AISTATS*. Vol. 2005. Citeseer, pp. 57–64 (cit. on p. 96).

Chapelle, O., B. Schölkopf, and A. Zien (2006). *Semi-supervised learning*. Cambridge, MA, USA: MIT Press (cit. on pp. 96, 97).

Cheminée, P., C. Gherghinoiu, and C. Besnainou (2005). "Analyses des verbalisations libres sur le son du piano versus analyses acoustiques". In: *Proceedings of the Conference on Interdisciplinary Musicology (CIM05)*. Montréal, Canada (cit. on pp. 12, 36, 40, 57, 58).

Chen, K., C. Choy, M. Savva, A. Chang, T. Funkhouser, and S. Savarese (2018). "Text2shape: Generating shapes from natural language by learning joint embeddings". In: *Proceedings of the Asian Conference on Computer Vision (ACCV)*. Springer. Perth, Australia, pp. 100–116 (cit. on p. 2).

Chollet, F. et al. (2015). *Keras*. https://keras.io (cit. on pp. 75, 99).

Chowning, J. (1973). "The synthesis of complex audio spectra by means of frequency modulation". In: *Journal of the Audio Engineering Society* 21.7, pp. 526–534 (cit. on p. 13).

Chung, J., K. Kastner, L. Dinh, K. Goel, A. C. Courville, and Y. Bengio (2015). "A recurrent latent variable model for sequential data". In: *Advances in Neural Information Processing Systems (NIPS)*. Montréal, Canada, pp. 2980–2988 (cit. on p. 117).

Colonel, J., C. Curro, and S. Keene (2017). "Improving neural net auto encoders for music synthesis". In: *Audio Engineering Society Convention*. New York, NY, USA (cit. on pp. 23, 60, 67).

Conan, S., E. Thoret, M. Aramaki, O. Derrien, C. Gondre, S. Ystad, and R. Kronland-Martinet (2014). "An intuitive synthesizer of continuous-interaction sounds: Rubbing, scratching, and rolling". In: *Computer Music Journal* 38.4, pp. 24–37 (cit. on p. 2).

d'Alessandro, C. and B. Doval (2003). "Voice quality modification for emotional speech synthesis". In: *European Conference on Speech Communication and Technology (EUROSPEECH)*. Geneva, Switzerland (cit. on p. 4).

d'Alessandro, C., N. d'Alessandro, S. Le Beux, and B. Doval (2006a). "Comparing time-domain and spectral-domain voice source models for gesture controlled vocal instruments". In: *Proceedings of the International Conference on Voice Physiology and Biomechanics (ICVPB)*. Vol. 34. Tokyo, Japan, p. 37 (cit. on p. 4).

d'Alessandro, N., C. d'Alessandro, S. Le Beux, and B. Doval (2006b). "Real-time CALM synthesizer new approaches in hands-controlled voice synthesis". In: *Proceedings of the Conference on New Interfaces for Musical Expression (NIME)*. Paris, France, pp. 266–271 (cit. on p. 4).

Day, W. and H. Edelsbrunner (1984). "Efficient algorithms for agglomerative hierarchical clustering methods". In: *Journal of Classification* 1.1, pp. 7–24 (cit. on p. 38).

Diniz, P., E. Da Silva, and S. Netto (2010). *Digital signal processing: System analysis and design*. Cambridge University Press (cit. on p. 68).

Dolson, M.B. (1986). "The phase vocoder: A tutorial". In: *Computer Music Journal* 10.4, pp. 14–27 (cit. on p. 71).

Donahue, C., J. McAuley, and M. Puckette (2019). "Adversarial audio synthesis". In: *Proceedings of the International Conference on Learning Representations (ICLR)*. New Orleans, LA, USA (cit. on pp. 24, 117).

Donnadieu, S. (2007). "Mental representation of the timbre of complex sounds". In: *Analysis, Synthesis, and Perception of Musical Sounds*. Springer, pp. 272–319 (cit. on p. 6).

Dubois, D. (1997). *Catégorisation et cognition : de la perception au discours*. Editions Kimé (cit. on p. 35).

— (2008). "Sens communs et sens commun : expériences sensibles, connaissance(s) ou doxa ?" In: *Langages* 2, pp. 41–53 (cit. on p. 35).

Duong, N., E. Vincent, and R. Gribonval (2010). "Under-determined reverberant audio source separation using a full-rank spatial covariance model". In: *IEEE Transactions on Audio, Speech, and Language Processing (TASLP)* 18.7, pp. 1830–1840 (cit. on p. 69).

Ehrette, T. (2004). "Les voix des services vocaux, de la perception à la modélisation". PhD thesis. Paris 11 (cit. on pp. 10–12, 28, 43, 54, 116).

Engel, J., C. Resnick, A. Roberts, S. Dieleman, M. Norouzi, D. Eck, and K. Simonyan (2017). "Neural audio synthesis of musical notes with wavenet autoencoders". In: *Proceedings of the International Conference on Machine Learning (ICML)*. Sydney, Australia, pp. 1068–1077 (cit. on pp. 23–25, 67, 72, 73).

Engel, J., K. K. Agrawal, S. Chen, I. Gulrajani, C. Donahue, and A. Roberts (2019). "GANSynth: Adversarial neural audio synthesis". In: *Proceedings of the International Conference on Learning Representations (ICLR)*. New Orleans, LA, USA (cit. on pp. 24, 25, 67, 117).

Esling, P., A. Chemla-Romeu-Santos, and A. Bitton (2018a). "Bridging audio analysis, perception and synthesis with perceptually-regularized variational timbre spaces". In: *Pro-*

ceedings of the International Society for Music Information Retrieval Conference (ISMIR). Paris, France, pp. 175–181 (cit. on pp. 95, 97, 99).

Esling, P., A. Chemla-Romeu-Santos, and A. Bitton (2018b). "Generative timbre spaces with variational audio synthesis". In: *Proceedings of the International Conference on Digital Audio Effects (DAFx)*. Aveiro, Portugal (cit. on pp. 11, 24, 61, 68, 94, 95, 99).

European Broadcasting Union (EBU) (2016). "Loudness range: A measure to normalization supplement EBU R 128 loudness". In: *EBU R 128 TECH 3342* Supplementary information for R 128 Version 3.0 (cit. on p. 106).

Faure, A. (2000). "Des sons aux mots, comment parle-t-on du timbre musical ?" PhD thesis. Ecole des Hautes Etudes en Sciences Sociales (EHESS) (cit. on pp. 10–12, 28, 36, 43, 57, 58).

Feugère, L., C. d'Alessandro, B. Doval, and O. Perrotin (2017). "Cantor Digitalis: Chironomic parametric synthesis of singing". In: *EURASIP Journal on Audio, Speech, and Music Processing* 2017.1, p. 2 (cit. on p. 4).

Févotte, C. and A.T. Cemgil (2009). "Nonnegative matrix factorizations as probabilistic inference in composite models". In: *European Signal Processing Conference (EUSIPCO)*. IEEE. Glasgow, Scotland (cit. on p. 68).

Févotte, C., N. Bertin, and J.-L. Durrieu (2009). "Nonnegative matrix factorization with the Itakura-Saito divergence: With application to music analysis". In: *Neural Computation* 21.3, pp. 793–830 (cit. on p. 68).

Foroughmand, H. and G. Peeters (2019). "Deep-rhythm for global tempo estimation in music". In: *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*. Delft, The Netherlands (cit. on p. 23).

Fritz, C., A. Blackwell, I. Cross, J. Woodhouse, and B. Moore (2012). "Exploring violin sound quality: Investigating English timbre descriptors and correlating resynthesized acoustical modifications with perceptual properties". In: *The Journal of the Acoustical Society of America* 131.1, pp. 783–794 (cit. on p. 40).

Gaillard, P. (2000). "Etude de la perception des transitoires d'attaque des sons de steeldrum : particularités acoustiques, transformation par synthèse et catégorisation". PhD thesis. Toulouse 2 (cit. on pp. 7, 8, 10–12).

Garnier, M. (2003). "Approche de la qualité vocale dans le chant lyrique : perception, verbalisation et corrélats acoustiques". MA thesis. Université Pierre et Marie Curie, Paris (cit. on p. 40).

Garnier, M., N. Henrich, M. Castellengo, D. Sotiropoulos, and D. Dubois (2007). "Characterisation of voice quality in Western lyrical singing: From teachers' judgements to acoustic descriptions". In: *Journal of Interdisciplinary Music Studies* 1.2, pp. 62–91 (cit. on pp. 10–12, 36, 58, 116).

Garofolo, J.S., L.F. Lamel, W.M. Fisher, J.G. Fiscus, D.S. Pallett, N.L. Dahlgren, and V. Zue (1993). "TIMIT acoustic phonetic continuous speech corpus". In: *Linguistic Data Consortium* (cit. on p. 24).

George, E.B. and M.J.T. Smith (1997). "Speech analysis/synthesis and modification using an analysis-by-synthesis/overlap-add sinusoidal model". In: *IEEE Transactions on Speech and Audio Processing* 5.5, pp. 389–406 (cit. on p. 71).

Girin, L., T. Hueber, and X. Alameda-Pineda (2017). "Extending the cascaded Gaussian mixture regression framework for cross-speaker acoustic-articulatory mapping". In: *IEEE/ACM Transactions on Audio, Speech, and Language Processing (TASLP)* 25.3, pp. 662–673 (cit. on p. 97).

Girin, L., T. Hueber, F. Roche, and S. Leglaive (2019). "Notes on the use of variational autoencoders for speech and audio spectrogram modeling". In: *Proceedings of the International Conference on Digital Audio Effects (DAFx)*. Birmingham, UK (cit. on p. 68).

Goodfellow, I., J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio (2014). "Generative adversarial nets". In: *Advances in Neural Information Processing Systems (NIPS)*. Montreal, Canada, pp. 2672–2680 (cit. on pp. 20–22).

Goodfellow, I., Y. Bengio, and A. Courville (2016). *Deep learning*. http://www.deeplearningbook.org. MIT Press (cit. on pp. 18, 21, 62).

Gounaropoulos, A. and C. Johnson (2006). "Synthesising timbres and timbre-changes from adjectives/adverbs". In: *Workshops on Applications of Evolutionary Computation*. EuroGP'06. Springer. Berlin, Germany, pp. 664–675 (cit. on p. 2).

Gregor, K., I. Danihelka, A. Graves, D. Rezende, and D. Wierstra (2015). "DRAW: A recurrent neural network for image generation". In: *Proceedings of the International Conference on Machine Learning (ICML)*. Lille, France, pp. 1462–1471 (cit. on p. 21).

Grey, J.M. (1977). "Multidimensional perceptual scaling of musical timbres". In: *the Journal of the Acoustical Society of America* 61.5, pp. 1270–1277 (cit. on pp. 8, 9, 11, 12, 24).

Griffin, D. and J. Lim (1984). "Signal estimation from modified short-time Fourier transform". In: *IEEE Transactions on Acoustics, Speech, and Signal Processing* 32.2, pp. 236–243 (cit. on pp. 24, 70, 86, 106).

Guyot, F. (1996). "Etude de la perception sonore en termes de reconnaissance et d'appréciation qualitative : une approche par la catégorisation". PhD thesis. Le Mans (cit. on pp. 10, 12).

Han, Y., J. Kim, and K. Lee (2017). "Deep convolutional neural networks for predominant instrument recognition in polyphonic music". In: *IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP)* 25.1, pp. 208–221 (cit. on p. 23).

Higgins, I., L. Matthey, A. Pal, C. Burgess, X. Glorot, M. Botvinick, S. Mohamed, and A. Lerchner (2017). "β-VAE: Learning basic visual concepts with a constrained variational framework". In: *Proceedings of the International Conference on Learning Representations (ICLR)*. Toulon, France (cit. on pp. 21, 66).

Hiller, L. and P. Ruiz (1971). "Synthesizing musical sounds by solving the wave equation for vibrating objects: Part 1". In: *Journal of the Audio Engineering Society* 19.6, pp. 462–470 (cit. on p. 17).

Hinton, G. and R. Salakhutdinov (2006). "Reducing the dimensionality of data with neural networks". In: *Science* 313.5786, pp. 504–507 (cit. on pp. 20, 63, 76).

— (2007). "Using deep belief nets to learn covariance kernels for Gaussian processes". In: *Advances in Neural Information Processing Systems (NIPS)*. Vancouver, Canada, pp. 1249–1256 (cit. on p. 97).

Hochreiter, S. and J. Schmidhuber (1997). "Long short-term memory". In: *Neural Computation* 9.8, pp. 1735–1780 (cit. on p. 63).

Hothorn, T., F. Bretz, and P. Westfall (2008). "Simultaneous inference in general parametric models". In: *Biometrical Journal: Journal of Mathematical Methods in Biosciences* 50.3, pp. 346–363 (cit. on p. 109).

Howard, D., A. Disley, and A. Hunt (2007). "Timbral adjectives for the control of a music synthesizer". In: *Proceedings of the 19th International Congress on Acoustics*. Madrid, Spain, pp. 2–7 (cit. on p. 2).

Hsu, W.-N., Y. Zhang, and J. Glass (2017). "Learning latent representations for speech generation and transformation". In: *Proceedings of the Conference of the International Speech Communication Association (Interspeech)*. Stockholm, Sweden, pp. 1273–1277 (cit. on pp. 24, 61, 68).

Huber, R. and B. Kollmeier (2006). "PEMO-Q: A new method for objective audio quality assessment using a model of auditory perception". In: *IEEE Transactions on Audio, Speech, and Language Processing (TASLP)* 14.6, pp. 1902–1911 (cit. on p. 75).

Hueber, T., L. Girin, X. Alameda-Pineda, and G. Bailly (2015). "Speaker-adaptive acoustic-articulatory inversion using cascaded Gaussian mixture regression". In: *IEEE/ACM Transactions on Audio, Speech, and Language Processing (TASLP)* 23.12, pp. 2246–2259 (cit. on p. 97).

International Telecommunication Union (ITU) (2015). "Algorithms to measure audio programme loudness and true-peak audio level". In: *Recommendation ITU-R BS.1770-4* BS Series - Broadcasting service (sound) (cit. on p. 106).

Isola, P., J.-Y. Zhu, T. Zhou, and A. Efros (2017). "Image-to-image translation with conditional adversarial networks". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Honolulu, HI, USA, pp. 1125–1134 (cit. on p. 22).

Iverson, P. and C. Krumhansl (1993). "Isolating the dynamic attributes of musical timbre". In: *The Journal of the Acoustical Society of America* 94.5, pp. 2595–2603 (cit. on pp. 9, 12, 24).

Jaccard, P. (1912). "The distribution of the flora in the alpine zone." In: *New Phytologist* 11.2, pp. 37–50 (cit. on p. 36).

Jadoul, Y., B. Thompson, and B. De Boer (2018). "Introducing Parselmouth: A Python interface to Praat". In: *Journal of Phonetics* 71, pp. 1–15 (cit. on p. 46).

Jillings, N., D. Moffat, B. De Man, and J. Reiss (2015). "Web Audio Evaluation Tool: A browser-based listening test environment". In: *Proceedings of the Sound and Music Computing Conference (SMC)*. Maynooth, Ireland (cit. on pp. 34, 48, 108).

Karras, T., T. Aila, S. Laine, and J. Lehtinen (2018). "Progressive growing of GANs for improved quality, stability, and variation". In: *Proceedings of the International Conference on Machine Learning (ICML)*. Vancouver, Canada (cit. on p. 22).

Kaufman, L. and P. Rousseeuw (1990). *Finding groups in data: An introduction to cluster analysis*. John Wiley & Sons (cit. on p. 51).

Kingma, D. and M. Welling (2014). "Auto-encoding variational Bayes". In: *Proceedings of the International Conference on Learning Representations (ICLR)*. Banff, Canada (cit. on pp. 21, 63, 64, 66).

Kingma, D. P. and J. Ba (2015). "Adam: A method for stochastic optimization". In: *Proceedings of the International Conference on Learning Representations (ICLR)*. San Diego, CA, USA (cit. on pp. 75, 99).

Kleimola, J. (2013). "Nonlinear abstract sound synthesis algorithms". PhD thesis. Aalto University (cit. on p. 13).

Kostopoulos, G., S. Karlos, S. Kotsiantis, and O. Ragos (2018). "Semi-supervised regression: A recent review". In: *Journal of Intelligent & Fuzzy Systems* 35.2, pp. 1483–1500 (cit. on p. 96).

Kreiman, J. and B. Gerratt (1998). "Validity of rating scale measures of voice quality". In: *The Journal of the Acoustical Society of America* 104.3, pp. 1598–1608 (cit. on pp. 12, 47).

Kreiman, J., B. Gerratt, G. Kempster, A. Erman, and G. Berke (1993). "Perceptual evaluation of voice quality: Review, tutorial, and a framework for future research". In: *Journal of Speech, Language, and Hearing Research* 36.1, pp. 21–40 (cit. on pp. 47, 49, 54).

Kreković, G., A. Pošćić, and D. Petrinović (2016). "An algorithm for controlling arbitrary sound synthesizers using adjectives". In: *Journal of New Music Research* 45.4, pp. 375–390 (cit. on p. 2).

Krizhevsky, A. (2009). *Learning multiple layers of features from tiny images*. Tech. rep. University of Toronto (cit. on p. 21).

Krumhansl, C. (1989). "Why is musical timbre so hard to understand?" In: *Structure and Perception of Electroacoustic Sound and Music* 9, pp. 43–53 (cit. on pp. 6, 24).

Kulkarni, T., W. Whitney, P. Kohli, and J. Tenenbaum (2015). "Deep convolutional inverse graphics network". In: *Advances in Neural Information Processing Systems (NIPS)*. Montreal, Canada, pp. 2539–2547 (cit. on p. 21).

Lakatos, S. (2000). "A common perceptual space for harmonic and percussive timbres". In: *Perception & Psychophysics* 62.7, pp. 1426–1439 (cit. on pp. 9, 12, 24).

Larsen, A. B. L., S. K. Sønderby, H. Larochelle, and O. Winther (2016). "Autoencoding beyond pixels using a learned similarity metric". In: *Proceedings of the International Conference on Machine Learning (ICML)*. New York, NY, USA, pp. 1558–1566 (cit. on p. 22).

Lartillot, O., P. Toiviainen, and T. Eerola (2008). "A Matlab toolbox for music information retrieval". In: *Data Analysis, Machine Learning and Applications*. Springer, pp. 261–268 (cit. on p. 46).

Lavoie, M. (2013). "Conceptualisation et communication des nuances de timbre à la guitare classique". PhD thesis. Université de Montréal (cit. on pp. 6, 36, 40).

LeCun, Y., C. Cortes, and C. Burges (1998). *The MNIST batabase of handwritten digits*. http://yann.lecun.com/exdb/mnist/ (cit. on pp. 20, 76).

Ledig, C., L. Theis, F. Huszár, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang, et al. (2017). "Photo-realistic single image super-resolution using a generative adversarial network". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Honolulu, HI, USA, pp. 4681–4690 (cit. on p. 22).

Leglaive, S., R. Hennequin, and R. Badeau (2015). "Singing voice detection with deep recurrent neural networks". In: *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE. Brisbane, Australia, pp. 121–125 (cit. on p. 23).

Leglaive, S., L. Girin, and R. Horaud (2018). "A variance modeling framework based on variational autoencoders for speech enhancement". In: *IEEE International Workshop on Machine Learning for Signal Processing (MLSP)*. Aalborg, Denmark (cit. on pp. 24, 69).

— (2019a). "Semi-supervised multichannel speech enhancement with variational autoencoders and non-negative matrix factorization". In: *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. Brighton, UK (cit. on pp. 24, 69).

Leglaive, S., U. Şimşekli, A. Liutkus, L. Girin, and R. Horaud (2019b). "Speech enhancement with variational autoencoders and alpha-stable distributions". In: *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. Brighton, UK (cit. on p. 24).

Lemaitre, G. and D. Rocchesso (2014). "On the effectiveness of vocal imitations and verbal descriptions of sounds". In: *The journal of the Acoustical Society of America* 135.2, pp. 862–873 (cit. on p. 10).

Lichte, W.H. (1941). "Attributes of complex tones". In: *Journal of Experimental Psychology* 28.6, p. 455 (cit. on pp. 3, 12).

Liutkus, A., R. Badeau, and G. Richard (2011). "Gaussian processes for underdetermined source separation". In: *IEEE Transactions on Signal Processing* 59.7, pp. 3155–3167 (cit. on p. 69).

Lopopolo, A. and E. van Miltenburg (2015). "Sound-based distributional models". In: *Proceedings of the International Conference on Computational Semantics (IWCS)*. London, UK, pp. 70–75 (cit. on pp. 35, 116).

Magron, P. (2016). "Reconstruction de phase par modèles de signaux : application à la séparation de sources audio". PhD thesis. Telecom ParisTech (cit. on pp. 71, 72).

Makhzani, A., J. Shlens, N. Jaitly, I. Goodfellow, and B. Frey (2015). "Adversarial autoencoders". In: *arXiv preprint arXiv:1511.05644* (cit. on p. 22).

Marozeau, J., A. de Cheveigné, S. McAdams, and S. Winsberg (2003). "The dependency of timbre on fundamental frequency". In: *The Journal of the Acoustical Society of America* 114.5, pp. 2946–2957 (cit. on p. 29).

McAdams, S. and B.L. Giordano (2009). "The perception of musical timbre". In: *The Oxford Handbook of Music Psychology*, pp. 72–80 (cit. on p. 7).

McAdams, S., S. Winsberg, S. Donnadieu, G. De Soete, and J. Krimphoff (1995). "Perceptual scaling of synthesized musical timbres: Common dimensions, specificities, and latent subject classes". In: *Psychological Research* 58.3, pp. 177–192 (cit. on pp. 9, 11, 12, 24).

McAulay, R.J. and T.F. Quatieri (1986). "Speech analysis/synthesis based on a sinusoidal representation". In: *IEEE Transactions on Acoustics, Speech, and Signal Processing* 34.4, pp. 744–754 (cit. on p. 71).

— (1995). "Sinusoidal coding". In: *Speech Coding and Synthesis*. Ed. by W.B. Kleijn and K.K. Paliwal. New York: Elsevier. Chap. 4 (cit. on p. 71).

McFee, B., C. Raffel, D. Liang, D. Ellis, M. McVicar, E. Battenberg, and O. Nieto (2015). "librosa: Audio and music signal analysis in Python". In: *Proceedings of the 14th Python in Science Conference*. Vol. 8. Austin, TX, USA (cit. on pp. 31, 46).

Mescheder, L., S. Nowozin, and A. Geiger (2017). "Adversarial variational bayes: Unifying variational autoencoders and generative adversarial networks". In: *Proceedings of the In-*

ternational Conference on Machine Learning (ICML). JMLR. org. Sydney, Australia, pp. 2391–2400 (cit. on p. 22).

Mikolov, T., I. Sutskever, K. Chen, G. Corrado, and J. Dean (2013). "Distributed representations of words and phrases and their compositionality". In: *Advances in Neural Information Processing Systems (NIPS)*. Lake Tahoe, NV, USA, pp. 3111–3119 (cit. on p. 35).

Miller, J. and E. Carterette (1975). "Perceptual space for musical structures". In: *The Journal of the Acoustical Society of America* 58.3, pp. 711–720 (cit. on p. 12).

Miranda, E. (2002). *Computer sound design: Synthesis techniques and programming*. Music Technology series. Focal Press (cit. on pp. 13–17).

Miron, M., J. Janer Mestres, and E. Gómez (2017). "Monaural score-informed source separation for classical music using convolutional neural networks". In: *Proceedings of the International Society for Music Information Retrieval (ISMIR)*. Suzhou, China (cit. on p. 23).

Nigam, K., A. K. McCallum, S. Thrun, and T. Mitchell (2000). "Text classification from labeled and unlabeled documents using EM". In: *Machine Learning* 39.2-3, pp. 103–134 (cit. on p. 96).

Oliveira, C. S., F. G. Cozman, and I. Cohen (2005). "Splitting the unsupervised and supervised components of semisupervised learning". In: *Proceedings of the Workshop Learning with Partially Classified Training Data, International Conference on Machine Learning (ICML)*. Bonn, Germany (cit. on p. 97).

Opolko, F. and J. Wapnick (1987). *MUMS: McGill University master samples* (cit. on p. 9).

Park, H. and C. Yoo (2017). "Melody extraction and detection through LSTM-RNN with harmonic sum loss". In: *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE. New Orleans, LA, USA, pp. 2766–2770 (cit. on p. 22).

Pedregosa, F. et al. (2011). "Scikit-learn: Machine learning in Python". In: *Journal of Machine Learning Research* 12, pp. 2825–2830 (cit. on p. 75).

Peeters, G. (2004). *A large set of audio features for sound description (similarity and classification) in the CUIDADO project*. Tech. rep. IRCAM (cit. on pp. 31, 46).

Peeters, G., B. Giordano, P. Susini, N. Misdariis, and S. McAdams (2011). "The timbre toolbox: Extracting audio descriptors from musical signals". In: *The Journal of the Acoustical Society of America* 130.5, pp. 2902–2916 (cit. on pp. 11, 31, 46).

Plomp, R. (1976). *Aspects of tone sensation: A psychophysical study*. Academic Press (cit. on p. 12).

Pons, J., O. Slizovskaia, R. Gong, E. Gómez, and X. Serra (2017). "Timbre analysis of music audio signals with convolutional neural networks". In: *European Signal Processing Conference (EUSIPCO)*. IEEE. Kos Island, Greece, pp. 2744–2748 (cit. on p. 23).

Portnoff, M.R. (1980). "Time-frequency representation of digital signals and systems based on short-time Fourier analysis". In: *IEEE Transactions on Acoustics, Speech, and Signal Processing* 28.1, pp. 55–69 (cit. on p. 71).

Puckette, M. (2007). *Theory and techniques of electronic music*. World Scientific Press (cit. on p. 13).

Radford, A., L. Metz, and S. Chintala (2016). "Unsupervised representation learning with deep convolutional generative adversarial networks". In: *Proceedings of the International Conference on Learning Representations (ICLR)*. San Juan, Puerto Rico (cit. on p. 24).

Ranzato, M.A. and M. Szummer (2008). "Semi-supervised learning of compact document representations with deep networks". In: *Proceedings of the International Conference on Machine learning (ICML)*. Helsinki, Finland, pp. 792–799 (cit. on p. 97).

Reed, S., Z. Akata, X. Yan, L. Logeswaran, B. Schiele, and H. Lee (2016). "Generative adversarial text to image synthesis". In: *Proceedings of the International Conference on Machine Learning (ICML)*. New York, NY, USA (cit. on pp. 2, 22).

Rezende, D., S. Mohamed, and D. Wierstra (2014). "Stochastic backpropagation and approximate inference in deep generative models". In: *Proceedings of the International Conference on Machine Learning (ICML)*. Beijing, China, pp. 1278–1286 (cit. on pp. 21, 63).

Risset, J.-C. and D. Wessel (1982). "Exploration of timbre by analysis and synthesis". In: *The Psychology of Music* 2, p. 151 (cit. on p. 6).

Roads, C. (1996). *The computer music tutorial*. MIT press (cit. on p. 13).

Roche, F., T. Hueber, S. Limier, and L. Girin (2019). "Autoencoders for music sound modeling: A comparison of linear, shallow, deep, recurrent and variational models". In: *Proceedings of the Sound and Music Computing Conference (SMC)*. Málaga, Spain (cit. on p. 73).

Rosch, E. (1999). "Principles of categorization". In: *Concepts: Core Readings* 189 (cit. on p. 10).

Rosenblatt, F. (1961). *Principles of neurodynamics: Perceptrons and the theory of brain mechanisms*. Tech. rep. Cornell Aeronautical Lab Inc Buffalo NY (cit. on p. 19).

Rumelhart, D., G. Hinton, and R. Williams (1986). "Learning representations by back-propagating errors". In: *Nature* 323.6088, p. 533 (cit. on pp. 20, 62, 66).

Russ, M. (2008). *Sound synthesis and sampling*. Focal Press (cit. on p. 13).

Salimans, T., D. Kingma, and M. Welling (2015). "Markov chain monte carlo and variational inference: Bridging the gap". In: *Proceedings of the International Conference on Machine Learning (ICML)*. Lille, France, pp. 1218–1226 (cit. on p. 21).

Salimans, T., I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen (2016). "Improved techniques for training GANs". In: *Advances in Neural Information Processing Systems (NIPS)*. Barcelona, Spain, pp. 2234–2242 (cit. on p. 25).

Samson, S., R. Zatorre, and J. Ramsay (1997). "Multidimensional scaling of synthetic musical timbre: Perception of spectral and temporal characteristics". In: *Canadian Journal of Experimental Psychology/Revue canadienne de psychologie expérimentale* 51.4, p. 307 (cit. on p. 12).

Sarroff, A. and M. Casey (2014). "Musical audio synthesis using autoencoding neural nets". In: *Proceedings of the Joint International Computer Music Conference (ICMC) and Sound & Music Computing conference (SMC)*. Athens, Greece (cit. on pp. 23, 60, 67).

Schaeffer, P. (1966). *Traité des objets musicaux*. Le Seuil (cit. on p. 11).

Schreiber, H. and M. Müller (2018). "A Single-Step approach to musical tempo estimation using a convolutional neural network". In: *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*. Paris, France (cit. on p. 23).

Schwarz, D. (2004). "Data-driven concatenative sound synthesis". PhD thesis. Université Paris 6 - Pierre et Marie Curie (cit. on p. 15).

— (2006). "Concatenative sound synthesis: The early years". In: *Journal of New Music Research* 35.1, pp. 3–22 (cit. on p. 15).

Serra, X. (1997). "Musical sound modeling with sinusoids plus noise". In: *Musical Signal Processing*. Ed. by C. Roads, S. Pope, A. Picialli, and G. De Poli. Lisse, the Netherlands: Swets & Zeitlinger (cit. on p. 71).

— (2007). "State of the art and future directions in musical sound synthesis". In: *IEEE Workshop on Multimedia Signal Processing (MMSP)*. IEEE. Chania, Crete, Greece, pp. 9–12 (cit. on pp. 13, 17).

Serra, X. and J. Smith (1990). "Spectral modeling synthesis: A sound analysis/synthesis system based on a deterministic plus stochastic decomposition". In: *Computer Music Journal* 14.4, pp. 12–24 (cit. on p. 71).

Shmelkov, K., C. Schmid, and K. Alahari (2018). "How good is my GAN?" In: *Proceedings of the European Conference on Computer Vision (ECCV)*. Munich, Germany, pp. 213–229 (cit. on p. 22).

Smaragdis, P. and J.C. Brown (2003). "Non-negative matrix factorization for polyphonic music transcription". In: *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*. New Paltz, NJ (cit. on p. 69).

Smith, J. O. (1991). "Viewpoints on the history of digital synthesis". In: *Proceedings of the International Computer Music Conference (ICMC)*. International Computer Music Association. Montreal, Canada (cit. on p. 13).

— (1992). "Physical modeling using digital waveguides". In: *Computer Music Journal* 16.4, pp. 74–91 (cit. on p. 17).

— (2004). "Virtual acoustic musical instruments: Review and update". In: *Journal of New Music Research* 33.3, pp. 283–304 (cit. on pp. 16, 17).

Steinmetz, C. (2018). *pyloudnorm*. Python package (cit. on p. 106).

Sturmel, N. and L. Daudet (2011). "Signal reconstruction from STFT magnitude: A state of the art". In: *Proceedings of the International Conference on Digital Audio Effects (DAFx)*. Paris, France, pp. 375–386 (cit. on p. 70).

Tolonen, T., V. Välimäki, and M. Karjalainen (1998). *Evaluation of modern sound synthesis methods*. Helsinki University of Technology (cit. on pp. 13, 14, 16, 17).

Tolstikhin, I., O. Bousquet, S. Gelly, and B. Schoelkopf (2018). "Wasserstein auto-encoders". In: *Proceedings of the International Conference on Learning Representations (ICLR)*. Toulon, France (cit. on p. 116).

Townsend, B. (1996). "New guitar music". In: *Guitar Review*, pp. 40–40 (cit. on p. 40).

Traube, C. (2004). "An interdisciplinary study of the timbre of the classical guitar". PhD thesis. McGill University (cit. on pp. 6, 10–12, 40, 57, 58).

van den Oord, A., N. Kalchbrenner, and K. Kavukcuoglu (2016a). "Pixel recurrent neural networks". In: *Proceedings of the International Conference on Machine Learning (ICML)*. New York, NY, USA, pp. 1747–1756 (cit. on p. 23).

van den Oord, A., S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. Senior, and K. Kavukcuoglu (2016b). "Wavenet: A generative model for raw audio". In: *arXiv preprint arXiv:1609.03499* (cit. on pp. 23, 67).

van der Maaten, L. and G. Hinton (2008). "Visualizing data using t-SNE". In: *Journal of Machine Learning Research* 9.Nov, pp. 2579–2605 (cit. on pp. 32, 94, 103).

Vijayakumar, A., R. Vedantam, and D. Parikh (2017). "Sound-Word2Vec: Learning word representations grounded in sounds". In: *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Copenhagen, Denmark, pp. 920–925 (cit. on pp. 35, 116).

Vincent, E., M. Jafari, S. Abdallah, M. Plumbley, and M. Davies (2011). "Probabilistic modeling paradigms for audio source separation". In: *Machine Audition: Principles, Algorithms and Systems*. IGI global, pp. 162–185 (cit. on p. 69).

Vincent, P., H. Larochelle, I. Lajoie, Y. Bengio, and P.-A. Manzagol (2010). "Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion". In: *Journal of Machine Learning Research* 11.Dec, pp. 3371–3408 (cit. on pp. 20, 21).

Virtanen, T. (2007). "Monaural sound source separation by nonnegative matrix factorization with temporal continuity and sparseness criteria". In: *IEEE Transactions on Audio, Speech, and Language Processing (TASLP)* 15.3, pp. 1066–1074 (cit. on p. 69).

von Bismarck, G. (1974). "Timbre of steady sounds: A factorial investigation of its verbal attributes". In: *Acta Acustica united with Acustica* 30.3, pp. 146–159 (cit. on pp. 3, 12, 29).

von Helmholtz, H.L.F. (1875). *On the sensations of tone as a physiological basis for the theory of music*. Longmans, Green (cit. on p. 6).

Walker, J., C. Doersch, A. Gupta, and M. Hebert (2016). "An uncertain future: Forecasting from static images using variational autoencoders". In: *Proceedings of the European Conference on Computer Vision (ECCV)*. Springer. Amsterdam, Netherlands, pp. 835–851 (cit. on p. 21).

Wessel, D. (1987). *Control of phrasing and articulation in synthesis*. Ann Arbor, MI: Michigan Publishing, University of Michigan Library (cit. on pp. 9, 10).

Xian, W., P. Sangkloy, V. Agrawal, A. Raj, J. Lu, C. Fang, F. Yu, and J. Hays (2018). "TextureGAN: Controlling deep image synthesis with texture patches". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Salt Lake City, UT, USA, pp. 8456–8465 (cit. on p. 2).

Yu, F., Y. Zhang, S. Song, A. Seff, and J. Xiao (2015). "LSUN: Construction of a large-scale image dataset using deep learning with humans in the loop". In: *arXiv preprint arXiv:1506.03365* (cit. on p. 22).

Zacharakis, A. (2013). "Musical timbre: Bridging perception with semantics". PhD thesis. Queen Mary University of London (cit. on pp. 3, 12).

Zacharakis, A., K. Pastiadis, and J.D. Reiss (2014). "An interlanguage study of musical timbre semantic dimensions and their acoustic correlates". In: *Music Perception: An Interdisciplinary Journal* 31.4, pp. 339–358 (cit. on pp. 11, 12, 28, 57, 58, 116).

Zhang, H., T. Xu, H. Li, S. Zhang, X. Wang, X. Huang, and D. Metaxas (2017). "StackGAN: Text to photo-realistic image synthesis with stacked generative adversarial networks". In: *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*. Venice, Italy, pp. 5907–5915 (cit. on p. 22).

Zhang, Z., Y. Xie, and L. Yang (2018). "Photographic text-to-image synthesis with a hierarchically-nested adversarial network". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Salt Lake City, UT, USA, pp. 6199–6208 (cit. on p. 2).

Zhou, Z.-H. (2017). "A brief introduction to weakly supervised learning". In: *National Science Review* 5.1, pp. 44–53 (cit. on p. 96).

Zhou, Z.-H. and M. Li (2005). "Semi-supervised regression with co-training". In: *IJCAI*. Vol. 5, pp. 908–913 (cit. on p. 96).

Zhu, J.-Y., T. Park, P. Isola, and A. Efros (2017). "Unpaired image-to-image translation using cycle-consistent adversarial networks". In: *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*. Venice, Italy, pp. 2223–2232 (cit. on p. 22).

Zweig, M. and G. Campbell (1993). "Receiver-operating characteristic (ROC) plots: A fundamental evaluation tool in clinical medicine." In: *Clinical Chemistry* 39.4, pp. 561–577 (cit. on p. 109).

Résumé en français

Introduction

Un des enjeux majeurs du marché des synthétiseurs et de la recherche en synthèse sonore aujourd'hui est de proposer une nouvelle forme de synthèse permettant de générer des sonorités inédites de qualité tout en offrant aux utilisateurs de nouveaux contrôles plus intuitifs afin de les aider dans leur processus créatif. En effet, les synthétiseurs sont actuellement des outils très puissants offrant aux musiciens une large palette de possibilités pour la création de sons, mais souvent très complexes avec des paramètres de contrôle dont la manipulation nécessite des connaissances expertes. C'est dans ce contexte que s'inscrivent ces travaux de thèse qui se sont déroulés en partenariat[1] entre le GIPSA-Lab[2] et Arturia[3] dont une des préoccupations majeures est d'offrir aux musiciens des produits innovants, intuitifs et faciles d'utilisation.

Inspirés par les récentes avancées en génération d'images proposant un contrôle intuitif de la synthèse comme décrire textuellement l'image voulue ou en faire un croquis par exemple, nous nous sommes intéressés à la recherche de méthodes permettant de contrôler la synthèse sonore à l'aide de paramètres de haut-niveau perceptivement pertinents, à savoir des termes communément employés par les musiciens pour décrire le timbre musical, et plus particulièrement le timbre synthétique. L'utilisation de tels paramètres de contrôle en combinaison avec des algorithmes d'apprentissage machine nous permettrait donc de relier cet espace perceptif avec un espace de représentation haut-niveau et ainsi générer des sons hybrides potentiellement intéressants pour les utilisateurs, par exemple un son qui serait à la fois 40% métallique, 20% agressif et 40% évolutif.

On peut ainsi résumer l'objectif principal de ces travaux de thèse par le développement et l'évaluation de nouvelles méthodes d'apprentissage machine pour la synthèse sonore permettant la génération de nouveaux sons de qualité tout en fournissant des paramètres de contrôle pertinents perceptivement. Comme premier pas vers cet objectif ambitieux, nous nous sommes d'abord intéressés à la synthèse de nouveaux timbres par la transformation de sons à l'aide de ces contrôles de haut-niveau.

Trois problématiques majeures faisant appel à des domaines d'expertise variés allant des sciences cognitives à l'apprentissage machine en passant par le traitement du signal se sont alors dégagées. Le premier défi que nous avons relevé a été de caractériser perceptivement le timbre musical synthétique en mettant en évidence un jeu de descripteurs verbaux utilisés fréquemment et de manière consensuelle par les musiciens. Ensuite, une fois les termes

[1] Ces travaux se sont déroulés dans le cadre d'un programme CIFRE proposé par l'ANRT (Association Nationale de la Recherche et de la Technologie).

[2] Laboratoire de recherche en traitement du signal basé à Grenoble.

[3] Entreprise basée à Montbonnot Saint-Martin (proximité de Grenoble) qui produit de l'équipement musical et en particulier des synthétiseurs numériques et analogiques (https://www.arturia.com/).

définis, nous avons exploré l'utilisation d'algorithmes d'apprentissage machine permettant de mettre en relation l'espace sonore avec un espace de représentation haut-niveau présentant des propriétés intéressantes d'interpolation et d'extrapolation. Ceci nous permettrait alors de naviguer de manière continue dans cet espace tout en explorant des sons au-delà des limites de la base de données afin de créer de nouvelles sonorités avec une bonne qualité. Enfin, cet espace de représentation extrait automatiquement ayant peu de chances d'avoir du sens perceptivement, le dernier challenge que nous avons eu à relever consistait à relier cet espace de représentation haut-niveau avec les descripteurs verbaux précédemment mis en évidence en utilisant des algorithmes d'apprentissage faiblement supervisé.

1 Résumé de l'état de l'art sur la perception et la synthèse de sons musicaux

Afin d'appréhender ces trois problématiques et de nous positionner par rapport à la littérature, nous avons tout d'abord réalisé un état de l'art dans les domaines principaux liés à ces travaux de thèse, à savoir la caractérisation du timbre musical et les méthodes de synthèse sonore.

1.1 Perception du timbre musical

Malgré plus de 150 années de recherche sur le timbre musical, ce dernier reste un attribut sonore encore mal défini. Une des raison à cela provient certainement du fait que, contrairement à la hauteur, l'intensité ou la durée d'un son, ce dernier n'est pas relié à une dimension acoustique clairement définie mais est plutôt un attribut perceptif multidimensionnel. Une des définitions classiquement acceptée que nous pouvons tout de même citer est la suivante : "*Le timbre est cet attribut de la sensation auditive qui permet à l'auditeur de différencier deux sons de même hauteur et de même intensité et présentés de façon similaire*". Cependant cette définition n'est pas totalement satisfaisante car ne reflétant pas les différents niveaux d'analyse du timbre : le niveau macroscopique (timbre causal) permettant d'identifier la source sonore, différents instruments par exemple, et le niveau microscopique (timbre qualitatif) qui permet quant à lui de décrire les variations subtiles de qualités sonores, comme les différents modes de jeu d'un même instrument.

Afin de caractériser le timbre musical et ses corrélats acoustiques, deux approches différentes sont employées dans la littérature : l'approche psycho-acoustique qui part de la physique et de l'acoustique des sons pour les relier à la perception humaine, et l'approche sémio-acoustique qui consiste à d'abord analyser la perception pour ensuite caractériser le son. On peut alors distinguer deux principales familles de méthodes de caractérisation de timbre faisant appel à l'une ou l'autre de ces deux approches. La première consiste à former des paires de sons et évaluer leur dissemblance à l'aide d'un test perceptif puis mener une analyse multidimensionnelle (MDS) afin de déterminer les principales dimensions perceptives pouvant expliquer cette dissemblance. La deuxième famille de méthodes se base quant à elle sur la qualification verbale du timbre à travers des tests d'écoute de verbalisation libre, de catégorisation libre ou utilisant des échelles sémantiques afin de caractériser des sons par exemple. Malgré leurs évidentes différences, ces méthodes présentent un objectif commun :

trouver les dimensions perceptives sous-jacentes du timbre. Cependant, selon les études et le type de sons utilisés, les dimensions ainsi que les corrélats acoustiques mis en évidence diffèrent grandement.

1.2 Synthèse de sons musicaux

En parallèle de la caractérisation du timbre et de sa perception, les chercheurs se sont également intéressés à l'étude de méthodes permettant à la fois de reproduire électriquement ou numériquement des timbres déjà existants comme ceux d'instruments de musique classiques et de générer des sonorités jusqu'alors inédites. De nos jours, plusieurs formes de synthèse existent et sont embarquées et commercialisées dans des synthétiseurs que l'on peut retrouver dans des studios de professionnels et/ou d'amateurs. On peut distinguer quatre familles principales reposant sur des techniques bien distinctes et produisant des sons avec des caractéristiques très différentes.

La première famille de formes de synthèse classiquement utilisées est celle de la synthèse par algorithmes abstraits. Cette méthode repose sur l'utilisation de formules mathématiques pour générer des sons sans interprétation physique directe. Un des types de synthèse les plus connus faisant appel à cette technique est la synthèse par modulation de fréquence (FM).

Une deuxième famille est celle des méthodes de synthèse par traitement de sons préenregistrés. Il s'agit de créer de nouvelles sonorités en manipulant/transformant des sons déjà existants. Un des exemples les plus connus de ce type de méthodes est la synthèse à base de samples, ou encore la synthèse granulaire qui se base sur l'agglomération de petits grains de sons enregistrés pour créer de nouveaux timbres.

Il existe également la synthèse par modélisation spectrale. Cette forme de synthèse consiste à modéliser les caractéristiques spectrales des sons et ainsi de proposer une synthèse présentant un lien assez direct avec la perception sonore. La synthèse additive ou soustractive sont deux exemples largement utilisés de ce genre de méthodes.

Enfin, la synthèse par modélisation physique est une dernière forme de synthèse classiquement répandue qui se base sur la modélisation mathématique des lois de la physique responsables de la production sonore. Cette technique est utilisée par exemple dans la synthèse modale ou par guide d'onde.

1.3 Synthèse par apprentissage machine

Les méthodes classiques de synthèse sonore précédemment introduites permettent la création d'une grande variété de sons mais souffrent toutes d'inconvénients majeurs tels qu'un trop grand nombre de paramètres de contrôle, une dépendance forte en capacité de stockage (mémoire) ou encore des contrôles trop complexes. La maîtrise de ces outils nécessite alors une expertise en processus de génération sonore freinant souvent l'accès de nombreux musiciens à ces instruments et ainsi à la palette sonore qui leur est associée. C'est pourquoi les chercheurs en synthèse sonore, inspirés par les récents avancements en génération d'images, ont commencé à s'intéresser à l'utilisation de l'apprentissage machine et en particulier des réseaux de neurones profonds pour développer de nouvelles méthodes de génération et/ou transformation de sons en proposant, entre autres, de nouveaux paramètres de contrôle.

2 Caractérisation perceptive du timbre synthétique

Afin de répondre à la première problématique de cette thèse, nous nous sommes donc d'abord intéressés à la caractérisation perceptive du timbre synthétique. Pour cela, nous avons réalisé deux études perceptives complémentaires visant à définir des descripteurs verbaux de timbre adaptés : une étude perceptive de verbalisation libre et une analyse sémantique différentielle.

2.1 Etude perceptive de verbalisation libre

L'objectif de ce premier test perceptif était de collecter des termes utilisés pour décrire des sons de synthétiseurs et parmi ces termes de venir sélectionner les plus fréquemment et transversalement utilisés en français.

Pour cela nous avons donc commencé par nous constituer une base de données de sons uniformisés en hauteur, en durée (entre 2 et 2.5 secondes) et normalisés en sonie que nous avons générés à partir des synthétiseurs logiciels d'Arturia. Nous avons ensuite sélectionné pour notre étude 50 sons de cette base de données (contenant 1,233 sons différents au total) de manière à ce qu'ils soient les plus représentatifs possible de l'espace acoustique formé par l'intégralité des sons. Une fois les stimuli sélectionnés, nous avons demandé à chaque participant de décrire librement 20 de ces sons par écrit en remplissant 5 espaces vides.

Après un nettoyage des données collectées auprès de 101 participants (correction des fautes d'orthographe, retrait des accents), nous avons obtenu 784 termes. Nous avons ensuite réalisé une analyse de proximité sémantique afin de regrouper les termes ayant le même sens ou ayant été utilisés dans un même contexte. Pour cela nous avons évalué à la fois la proximité sémantique entre deux termes utilisés par différents participants dans un même contexte sonore, et la proximité sémantique entre les termes au sein des réponses d'un même participant, c'est-à-dire évaluer si deux termes ont été utilisés conjointement de manière systématique (plus de deux fois) pour décrire plusieurs sons. Nous avons par la suite appliqué une méthode de classification ascendante hiérarchique afin de créer des catégories sémantiques.

Enfin, par une analyse de fréquence et de transversalité sur les termes mis en évidence ainsi que les 98 catégories sémantiques formées, nous avons pu dégager 8 dimensions perceptives : *métallique*, *chaud*, *soufflé*, *qui vibre*, *percussif*, *qui résonne*, *qui évolue* et *agressif*. Ces termes correspondent bien au vocabulaire mis en évidence dans la littérature par différentes études perceptives sur le timbre musical, cependant c'est la première fois, à notre connaissance, que sont identifiés les termes les plus fréquemment et consensuellement utilisés pour décrire des sons synthétiques.

2.2 Etude perceptive à échelles sémantiques

Nous avons ensuite réalisé un deuxième test perceptif se servant des résultats obtenus lors du test de verbalisation libre comme étiquettes d'échelles sémantiques. Ce test avait pour objectif d'analyser le degré de consensualité des dimensions perceptives sélectionnées ainsi que d'obtenir une évaluation quantitative d'un sous-ensemble de notre base de données sur ces dimensions en prévision de la (faible) supervision de nos modèles neuronaux de synthèse.

Afin d'analyser la consensualité des dimensions choisies, deux aspects étaient à prendre en

compte : la consistance dans l'utilisation des échelles par chaque participant (consensus intra-sujet), en d'autres termes, si les participants ont bien compris les dimensions; et le consensus entre les différents participants sur l'utilisation des échelles (consensus inter-sujet), c'est-à-dire évaluer si il y a en effet une conception partagée de ces termes. Pour cela, nous avons eu l'idée de mettre en place un test en deux étapes avec une première phase d'apprentissage permettant aux participants de se familiariser avec les sons et les échelles, puis une phase principale contenant à la fois de nouveaux sons et des sons déjà évalués au cours de la phase d'apprentissage afin d'analyser leur utilisation des échelles sémantiques.

Nous nous sommes donc constitués une base de stimuli de 80 sons contenant 10 sons d'apprentissage et 70 sons de test choisis ici aussi pour être les plus représentatifs possible de l'espace acoustique. Chaque participant avait ensuite pour objectif d'évaluer 40 sons, d'abord les 10 d'apprentissage (toujours les mêmes et présentés dans le même ordre) puis 30 sons sélectionnés aléatoirement (25 de test plus les 5 d'apprentissage réintroduits), à l'aide de 8 échelles sémantiques continues étiquetées avec les termes issus du test de verbalisation libre allant de "pas du tout" à "extrêmement".

Nous avons collecté les réponses de 71 participants sur lesquelles nous avons d'abord réalisé une analyse du consensus intra-sujet pour chaque échelle indépendamment des autres. A l'issue de cette étape, nous avons retiré pour chaque échelle les participants pour lesquels les résultats montraient une utilisation non consistante de cette dernière afin de ne pas détériorer la suite des analyses. Une fois certains participants ainsi écartés de l'étude (pour une échelle donnée), nous avons évalué le consensus inter-sujet et appliqué, pour chaque dimension perceptive, une classification ascendante hiérarchique afin de regrouper les participants ayant une même conception/utilisation de l'échelle. Nous avons ainsi pu faire ressortir une grande différence entre les échelles, certaines étant beaucoup plus consensuelles (par exemple *agressif* ou *percussif*) que d'autres (*qui vibre*, *qui résonne*). De plus, pour ces deux dernières dimensions perceptives, nous avons pu mettre en évidence un aspect dual qui pourrait expliquer, dans une certaine mesure, ces résultats limités et ouvre la discussion sur la potentielle utilisabilité de ces dimensions comme paramètres de contrôle d'un synthétiseur commercialisable.

Enfin, en sélectionnant judicieusement pour chaque échelle les participants au sein d'un même groupe, et partageant ainsi une même conception de la dimension perceptive associée, nous avons pu obtenir une évaluation quantitative de 80 sons et ainsi nous constituer un sous-espace de notre base de données étiqueté selon nos 8 dimensions perceptives.

3 Extraction d'un espace de contrôle haut-niveau de la synthèse sonore par apprentissage non supervisé

Dans un second temps, nous avons exploré l'utilisation d'algorithmes d'apprentissage machine pour l'extraction d'un espace de représentation haut-niveau avec des propriétés intéressantes d'interpolation et d'extrapolation à partir d'une base de données de sons, le but étant de mettre en relation cet espace avec les dimensions perceptives mises en évidence plus tôt.

S'inspirant de précédentes études sur la synthèse sonore faisant appel à des réseaux de neurones profonds, nous nous sommes concentrés sur des modèles de type autoencodeurs[4] et

[4]Réseaux de neurones profonds constitués d'un encodeur qui projette les données d'entrée dans un espace de représentation que l'on appelle espace latent, et d'un décodeur qui permet de revenir à l'espace d'origine en

avons réalisé une étude comparative approfondie de plusieurs variantes de ces modèles sur deux jeux de données différents.

3.1 Méthode d'analyse-transformation-synthèse

Tout au long de ce chapitre, afin d'extraire cet espace de haut-niveau, nous avons appliqué une méthode d'analyse-transformation-synthèse. En concordance avec les études précédentes, nous avons fait le choix d'utiliser le domaine temps-fréquence pour représenter nos données en appliquant la transformée de Fourier à court-terme à nos signaux originaux. La méthode consiste ensuite à projeter trame à trame notre son dans l'espace de représentation haut-niveau (latent) grâce à l'encodeur de notre modèle, obtenant ainsi pour notre spectrogramme d'amplitude une trajectoire de vecteurs latents (*analyse*). C'est alors dans cet espace que l'utilisateur va pouvoir appliquer une transformation, par exemple élargir ou réduire la trajectoire, interpoler entre les trajectoires latentes de deux sons différents, ou encore faire un fondu entre elles (*transformation*). Ensuite, cette nouvelle trajectoire latente va être décodée en un nouveau spectrogramme d'amplitude sur lequel on va appliquer un algorithme de reconstruction de phase (nous avons exploré l'algorithme de Griffin et Lim et la reconstruction par dépliement linéaire de phase) pour finalement reconstruire le signal temporel à l'aide de la transformée de Fourier à court-terme inverse avec *overlap-add* (*synthèse*).

3.2 Etude comparative des différents modèles autoencodeurs sur deux jeux de données

Au cours de cette thèse, nous avons étudié et comparé différentes variations de modèles autoencodeurs : un modèle linéaire correspondant à une analyse en composantes principales (ACP) pour l'encodeur et une analyse inverse pour le décodeur; des modèles d'autoencodeurs classique ou profonds; un modèle récurrent (qui possède un état interne permettant de capturer la dynamique du signal); et un modèle probabiliste, l'autoencodeur variationnel (VAE). Ce dernier modèle présente une contrainte supplémentaire de régularisation appliquée à l'espace latent et extrait à la fois les paramètres de la distribution de probabilité correspondant à l'encodeur et ceux de la distribution correspondant au décodeur, ce qui en fait donc un modèle (doublement) génératif. Pour chacun de ces modèles, nous avons exploré différents jeux de paramètres en faisant par exemple varier le nombre de neurones par couche (notamment la taille de l'espace latent) ou encore le nombre de couches, voire certains paramètres comme le coefficient de pondération appliqué à la régularisation dans le VAE.

Pour cette étude comparative, nous nous sommes intéressés à deux jeux de données différents : NSynth, une base de données de sons standardisée disponible publiquement qui nous a permis de nous comparer à l'état de l'art; et notre base de données Arturia que nous avons déjà introduite précédemment et qui correspond au type de sons auquel nous nous intéressons.

Nous avons alors comparé ces différents modèles sur ces deux jeux de données selon 3 critères. Tout d'abord, nous avons évalué les modèles en termes d'erreur de reconstruction atteinte dans un contexte d'analyse-synthèse (sans transformation) en nous basant à la fois sur une mesure physique objective, l'erreur quadratique moyenne (RMSE en dB), et une mesure objective liée à la perception de la qualité sonore, PEMO-Q. Nous avons ensuite

reconstruisant le signal à partir de ses coordonnées dans cet espace de représentation extrait par le modèle.

comparé la structure de l'espace latent extrait par chaque modèle en calculant les coefficients de corrélation entre les différentes dimensions extraites. Enfin, comme premier pas vers la navigation dans l'espace latent pour la synthèse de nouvelles sonorités, nous avons exploré l'utilisation de cet espace pour l'interpolation entre deux sons et comparé les résultats obtenus avec les différents modèles.

A partir des tests comparatifs que nous avons menés, nous pouvons tirer quelques conclusions. Tout d'abord, que ce soit sur un jeu de données standardisé ou plus réaliste compte tenu de notre application, les modèles autoencodeurs se sont avérés bien adaptés pour proposer une synthèse sonore de qualité tout en extrayant un espace de représentation haut-niveau dans lequel il est possible de naviguer de manière continue. En particulier, ces modèles ont démontré une bonne capacité à créer des sons hybrides pertinents en interpolant dans l'espace latent entre deux sons différents de l'ensemble de données, ce qui constitue une preuve de leur capacité à générer de nouvelles sonorités. De plus, grâce à leur aspect probabiliste, les modèles variationnels assurent une certaine décorrélation entre les coefficients latents, ce qui en fait de bons candidats pour extraire des paramètres de contrôle. Cependant, aucun lien entre ces espaces latents et les dimensions perceptives mises en évidence plus tôt, ni même aucune dimension perceptivement pertinente, n'est apparu naturellement, soulignant ainsi la nécessité d'insérer une supervision perceptive lors de l'apprentissage du modèle.

4 Vers la régularisation perceptive de modèles autoencodeurs par apprentissage faiblement supervisé

Pour finir, nous nous sommes donc intéressés à la dernière problématique de cette thèse et avons essayé d'insérer une forme de supervision pendant l'apprentissage des modèles en encourageant 8 des dimensions latentes extraites à être perceptivement pertinentes en coïncidant avec les 8 dimensions perceptives mises en évidence.

4.1 Méthodologie de régularisation perceptive

Inspirés par différentes études et compte-tenu de la taille de notre jeu de données étiqueté, nous avons proposé une méthode d'apprentissage semi-supervisé faisant appel à une procédure d'apprentissage en deux étapes pour régulariser perceptivement un autoencodeur variationnel. Cette méthode consiste d'abord à réaliser un pré-apprentissage non supervisé du modèle sur la base de données Arturia complète (comme dans la section précédente). Ensuite, en ajoutant une nouvelle contrainte sur l'espace latent extrait afin que certaines de ses dimensions correspondent aux évaluations obtenues au cours du deuxième test perceptif, il s'agit d'affiner le modèle en réalisant l'apprentissage sur le sous-ensemble étiqueté de notre base de données.

4.2 Evaluation du modèle perceptivement régularisé

Nous avons ensuite évalué notre méthode en mesurant sa capacité à maintenir une qualité audio convenable tout en régularisant correctement le modèle. De plus, étant donnée la taille du sous-ensemble de données étiqueté, nous voulions vérifier que notre modèle était capable de généraliser et qu'il n'avait pas seulement sur-appris.

Nous avons donc commencé par réaliser une première analyse objective afin d'évaluer la qualité audio atteinte par les modèles dans un contexte d'analyse-synthèse. Pour cela, nous avons comparé plusieurs modèles de VAE perceptivement régularisés en faisant varier les divers hyper-paramètres des modèles (dimension de l'espace latent, coefficient de pondération de la contrainte perceptive supplémentaire, nombre d'itérations de la procédure d'apprentissage) avec un modèle classique de VAE (sans contrainte perceptive). Cette comparaison a été effectuée, comme précédemment, à l'aide d'une mesure physique objective, l'erreur quadratique moyenne, et d'une mesure perceptive de l'erreur, PEMO-Q.

Nous avons ensuite voulu évaluer qualitativement si la régularisation perceptive avait une influence sur la structure de l'espace latent extrait. Pour cela, nous avons encodé la base de données étiquetée à l'aide d'un modèle de VAE régularisé (perceptivement) et d'un modèle non régularisé, et avons comparé leurs projections dans un espace à 2 dimensions obtenues à l'aide de l'algorithme de t-SNE (*t-distributed stochastic neighbor embedding*).

Finalement, après avoir choisi le modèle représentant le meilleur compromis en termes de qualité de reconstruction et de régularisation perceptive (en trouvant les hyper-paramètres optimaux), nous avons "bouclé la boucle" en évaluant la méthode proposée perceptivement à l'aide d'un dernier test d'écoute. Pour ce test, nous nous sommes concentrés sur les 5 dimensions perceptives n'étant pas reliées à la dynamique temporelle du signal (étant donné que notre modèle est statique), à savoir : *métallique*, *chaud*, *soufflé*, *qui vibre* et *agressif*. Pour chacune d'elles, nous avons sélectionné 12 sons de notre base de données, 6 de la base de données non étiquetée et 6 du sous-ensemble annoté dont 3 très représentatifs de la dimension en question et 3 non-représentatifs. Pour chacun de ces sons, nous avons appliqué deux composantes continues différentes sur la trajectoire latente correspondant à la dimension à analyser afin de créer deux sons, l'un étant plus *métallique* que l'autre par exemple. Nous avons ensuite demandé aux participants, pour chaque paire de sons créée, de déterminer lequel des deux était le plus *métallique* ou *agressif*.

A partir de ces tests préliminaires, nous avons pu valider notre méthode comme appropriée pour la génération de sons avec une qualité correcte et capturer les propriétés acoustiques de certaines (2 plus une à étudier davantage) des dimensions perceptives tout en permettant de généraliser son comportement dans un contexte d'analyse-transformation-synthèse sur des échantillons sonores jusque-là jamais rencontrés.

Conclusion

L'objectif de cette thèse était de développer et d'évaluer de nouvelles méthodes de synthèse sonore par apprentissage machine permettant de mettre en relation l'espace sonore avec un espace haut-niveau de contrôle présentant des propriétés intéressantes d'interpolation et d'extrapolation tout en proposant de nouveaux paramètres de contrôle plus intuitifs. Au cours de ces travaux, nous avons pu identifier 8 descripteurs verbaux pertinents perceptivement et utilisés fréquemment et transversalement pour décrire des sons de synthétiseurs. Nous avons ensuite mené une étude comparative approfondie de plusieurs modèles d'autoencodeurs nous permettant de valider ces modèles comme appropriés pour extraire un espace de représentation avec les propriétés souhaitées à l'exception de la pertinence perceptive des dimensions mises en évidence. Nous avons donc ajouté de la supervision lors de l'apprentissage de ces

modèles afin de faire coïncider certaines des dimensions de cet espace de représentation avec les dimensions perceptives identifiées. Nous avons validé notre méthodologie sur quelques unes de ces dimensions via un test d'écoute et avons ainsi montré la pertinence de notre méthode malgré la taille réduite de notre jeu de données étiqueté.

Ces travaux réalisés en se concentrant sur la transformation d'un son de départ selon des dimensions perceptives de timbre constituent un premier pas vers un contrôle plus intuitif de la synthèse sonore en proposant des paramètres de contrôle pertinents perceptivement. De futurs travaux à mener pourront pousser l'analyse des descripteurs verbaux afin de confirmer leur utilisation comme adaptée ou non au contrôle de la synthèse sonore ainsi qu'aller plus loin dans la démarche et explorer la synthèse directe à partir de ces paramètres.