Thèse

présentée pour obtenir le grade de docteur

de l'Université Pierre et Marie Curie

Slim Essid

Titre de la thèse

Classification automatique des signaux audio-fréquences : reconnaissance des instruments de musique

Soutenue le 13 décembre 2005 devant le jury composé de

Jean-Gabriel Ganascia	**Président**
Frédéric Bimbot	
Dirk Slock	**Rapporteurs**
Laurent Daudet	
Geoffroy Peeters	**Examinateurs**
Gaël Richard	
Bertrand David	**Directeurs de thèse**

A Monica et à ma famille,

Remerciements

Mes vifs remerciements vont tout d'abord à mes directeurs de thèse Gaël RICHARD et Betrand DAVID. Au delà de leurs multiples compétences scientifiques et de leur excellente capacité pédagogique, leur disponibilité, leur écoute, leur patience et leur soutien infaillible tout le long de ce travail de thèse, ont été déterminants. Gaël et Bertrand m'ont fait confiance et m'ont laissé beaucoup de liberté dans mon travail tout en me guidant et en m'incitant à me rattacher à des objectifs précis.

Je remercie tout spécialement les membres du jury de m'avoir fait l'honneur de participer à ma soutenance et de s'être pencher de près sur mon travail ; en particulier Laurent Daudet avec qui nous avons eu de nombreuses discussions fructueuses pendant ma dernière année de thèse, Jean-Gabriel Ganascia, président du jury, Frédéric Bimbot et Dirk Slock, rapporteurs et Geoffroy Peeters, examinateur.

Je dois beaucoup à Nicolas Moreau pour son soutien constant, sans lui, il n'y aurait pas eu cette thèse. Je remercie également Bernard Robinet de m'avoir fait confiance et de ses précieux conseils.

Le travail sur les transitoires d'attaque a été réalisé en étroite collaboration avec Pierre Leveau qui m'a apporté une aide précieuse. Les contributions sur la sélection des attributs sont nées d'un échange fructueux avec Marine Campedel, merci Marine pour ton aide.

Je remercie tout particulièrement Sophie-Charlotte Barrière pour ses interventions efficaces qui m'ont sauvées la mise à maintes reprises. C'est sûrement grâce à toi que j'ai pu aller aussi loin dans les simulations.

Merci à Michel Desnoues, Cléo, Chloé et Yves Grenier pour leur participation active aux sessions d'enregistrement de solos. Je suis également reconnaissant à tous ceux qui ont participé à l'élargissement de la base de données sonores à TSI et au test de perception des instruments.

Je remercie Laurence pour avoir toujours été là pour moi, ainsi que Patricia, Catherine et Stéphane Bonenfant.

Je n'aurais pas pu rêver meilleur environnement et meilleurs compagnons que ceux de TSI pendant ces années de dur labeur. J'aimerais dire à mes amis de TSI combien ils ont été importants pour moi : il faudrait des pages pour vous remercier un(e) par un(e) convenablement, je vous dis donc simplement "merci à tous du fond du coeur". Enfin, je ne remercierais jamais

assez Monica pour son aide inestimable, sa présence et son infini patience ainsi que ma famille et mes amis, en particulier, Skander et mes parents, de leurs encouragements et de leur soutien chaleureux et bienveillant.

Table des matières

Introduction Et Préalables

I. Introduction générale

Prélude

Monsieur Mélo a passé toute la journée avec cet air de musique dans la tête. Depuis ce matin il se pose la question : “Mais qu’est-ce que c’est que cette musique ?” et ça commence à l’agacer de ne pas savoir. Il se dit qu’il va essayer de trouver sur Internet mais une fois devant son PC, il se rend compte qu’il ne dispose d’aucun outil lui permettant de trouver ce qu’il cherche et il se dit : “Et pourquoi je ne pourrais pas lui fredonner les musiques que j’ai envie de trouver à cet ordinateur ?”. Il pense que ce serait vraiment bien, d’autant plus qu’il est compositeur de musique électronique et que ça lui arrive souvent de passer des heures à chercher sur ses cinq disques durs de 200Go remplis de musique, cette boucle de batterie qui fait “boum, tsi, boum, boum” ou une ligne de basse qui accompagnerait bien la mélodie qu’il vient de composer sur son clavier. Et d’ailleurs pourquoi ne pourrait t-il pas simplement jouer des motifs rythmiques ou mélodiques sur son clavier et ensuite demander à son PC de trouver les motifs ressemblants dans sa base de sons... C’est sûr que ça lui ferait gagner beaucoup de temps. Ça serait même très utile à son amie, Madame Targui qui travaille dans une boîte d’édition. Ses clients viennent souvent la voir pour lui demander “un extrait de trente secondes de solo de trompette, de préférence en Do mineur”. Elle rêve depuis des années d’un système qui étiquette automatiquement son catalogue sonore en fonction de l’orchestration, du rythme et de la mélodie pour pouvoir retrouver ce solo de trompette en quelques clics. Si ce système était de surcroît capable d’extraire automatiquement la partition de n’importe laquelle des pièces musicales de sa collection, ça aiderait beaucoup les musiciens à qui elle fait régulièrement appel pour enregistrer une variation sur le tube de l’été pour la pub télé qui servira à lancer le dernier produit de son plus grand client.

Le défi que doivent relever les chercheurs est de proposer des solutions qui répondent aux

besoins de Madame Targui, Monsieur Mélo et plus généralement à ceux de millions d'utilisateurs, amateurs et professionnels submergés par un flot de données multimédia, en particulier sous forme sonore, qu'il devient difficile de manipuler en l'absence d'outils appropriés. La nécessité de mettre en œuvre des dispositifs garantissant un accès intelligent et simplifié à un tel foisonnement de contenus a fait émerger une nouvelle discipline : l'*indexation automatique.* L'enjeu est d'une importance telle que la problématique de l'indexation fait l'objet d'un standard international, connu sous le nom de MPEG-7 [ISO/IEC, 2001, Chang *et al.*, 2001], qui s'intéresse à formaliser des schémas de descriptions de contenus multimédia.

Nous nous intéressons en particulier au problème d'indexation des signaux audio-fréquences, ou, plus succinctement, *signaux audio.*

De l'indexation à la classification

L'indexation automatique du signal audio a vocation à extraire d'un enregistrement sonore une représentation symbolique. Cette représentation est organisée par catégories de caractères suivant une structuration qui peut être générale ou détaillée. Dans le cas de la musique, par exemple, sont visés des concepts tels que le *rythme*, la *mélodie*, ou encore l'*instrumentation.* Ceux-ci peuvent prendre une forme hautement structurée : la *partition musicale.*

Comme nous l'avons suggéré précédemment, les applications de l'indexation automatique ne se limitent pas à l'extraction automatique de partitions. On retrouve parmi les plus populaires, des applications s'articulant autour de la recherche, la navigation et l'organisation des bases de données sonores : on parle de *recherche par le contenu.* En effet, l'obtention à partir des signaux, de représentations pertinentes permet d'envisager de retrouver, dans de grandes bases de données, les sons "ressemblant" à un exemple de référence- c'est la *recherche par similarité*- et plus généralement les sons répondant aux critères définis par l'utilisateur. Dans le cas de bases de données musicales, on peut imaginer des requêtes aussi variées que :

- *retrouver une valse* ;
- *retrouver un solo de violoncelle de Rostropovitch* ;
- *retrouver toutes les versions de "Summertime"*;
- *faire écouter les refrains de cet album* ;
- ...

Derrière cet objectif d'indexation se profile un processus fondamental : celui qui organise les évènements sonores en catégories. Les requêtes précédentes, par exemple, s'appuient sur la

définition de catégories de rythmes, d'instruments, d'artistes, etc., associées aux sons. De ce fait, nous considérons les différentes tâches d'indexation comme pouvant être approchées suivant un même paradigme qui consiste à les envisager comme un problème de *classification automatique*. Ce paradigme permet de résoudre plusieurs tâches clés de l'indexation audio. C'est là une idée qu'on retrouve dans le standard MPEG-7 qui prévoit le principe d'un schéma de classification d'un contenu audio en classes emboîtées. La figure I.1 présente un exemple d'une telle réalisation.

Notons que les frontières entre classes ne sont pas toujours définies de façon univoque. Dans la figure I.1, on peut voir, par exemple, que la classe "musique" et la classe "voix humaines" sont recouvrantes puisque la classe "voix chantée" appartient à ces deux catégories. La définition de classes disjointes peut donc s'avérer délicat dans des contextes d'application particuliers.

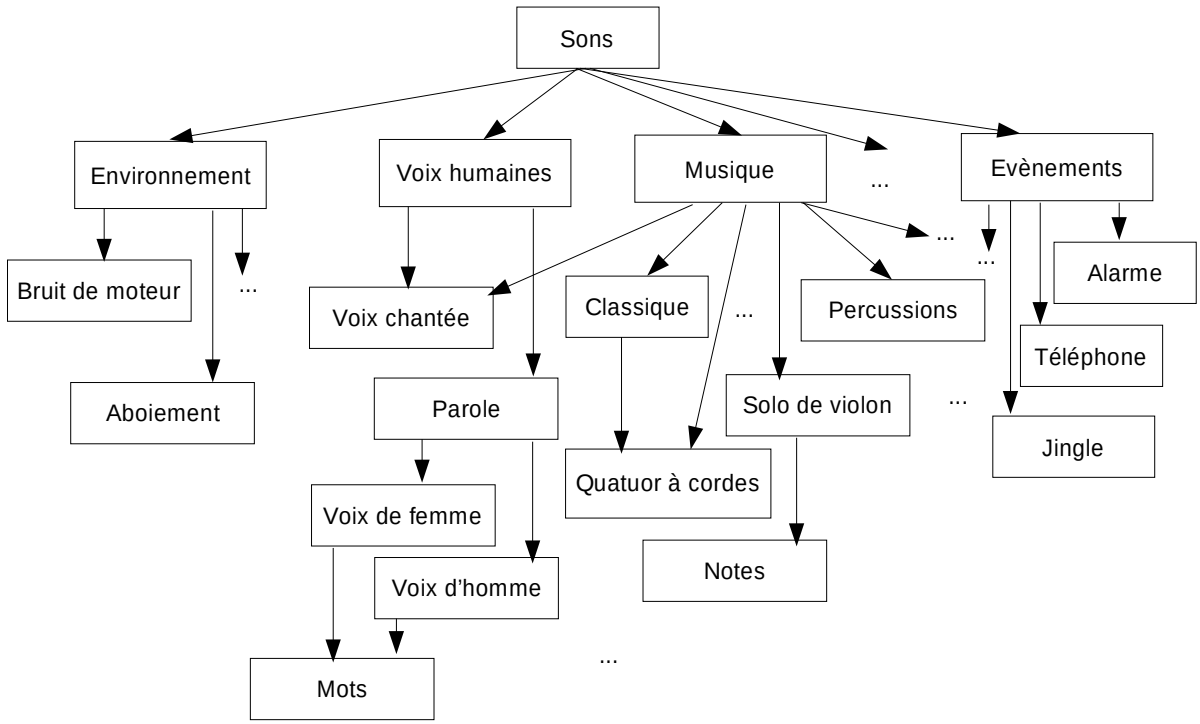

Fig. I.1 Exemple de schéma de classification audio général.

Classification automatique des signaux audio

La classification automatique vise à assigner des objets à des catégories ou *classes*. Dans notre cas, les objets sont des signaux ou des segments de signaux audio qu'il s'agit d'assigner à des classes telles que celles qui apparaissent sur la figure I.1. Le principe général des systèmes de classification audio (*cf.* figure I.2) inclut deux étapes :

- une *étape d'apprentissage* qui peut être vue comme une phase de développement aboutissant à la mise en œuvre d'une stratégie de classification ;
- une *étape de test* par laquelle les performances du système de classification sont évaluées.

En général, un système n'est prêt pour une utilisation réelle qu'après une succession d'étapes d'apprentissage et de test permettant de mettre en place une stratégie de classification efficace.

La phase d'apprentissage comprend :

- l'extraction à partir d'une base de sons de référence appelée *base d'apprentissage*, de descripteurs sous forme de paramètres numériques. Ces paramètres qui sont aussi appelés *attributs* (*features*) sont sensés caractériser des propriétés des signaux pouvant révéler leur appartenance à l'une des classes envisagées.
- La sélection d'attributs efficaces ; en pratique un nombre élevé de descripteurs candidats qui ne servent pas tous les performances de classification est considéré, il est alors intéressant d'avoir recours à des techniques permettant de retenir un sous-ensemble d'attributs (de plus petite taille) qui garantisse les meilleurs résultats de classification.
- l'apprentissage à partir des attributs sélectionnés de *fonctions de classification* ou *fonctions de décision*, lesquelles fonctions serviront à assigner des observations d'attributs de nouveaux exemples (de test) à l'une des classes possibles.

Lors de l'étape de test il n'est nécessaire d'extraire des signaux que les attributs qui ont été retenus et de décider de l'appartenance de ces signaux aux classes possibles en utilisant les fonctions de décisions apprises.

La conception de systèmes de classification audio apparaît ainsi comme un processus complexe qui demande la coopération de techniques issues de diverses disciplines :

- l'acoustique et la perception des sons, qui fournissent les pistes nécessaires au développement de descripteurs adéquats ;
- le traitement du signal, outil incontournable pour l'extraction efficace de ces descripteurs ;
- l'apprentissage automatique, qui permet de mettre en œuvre des stratégies de classification performantes en exploitant les descripteurs obtenus.

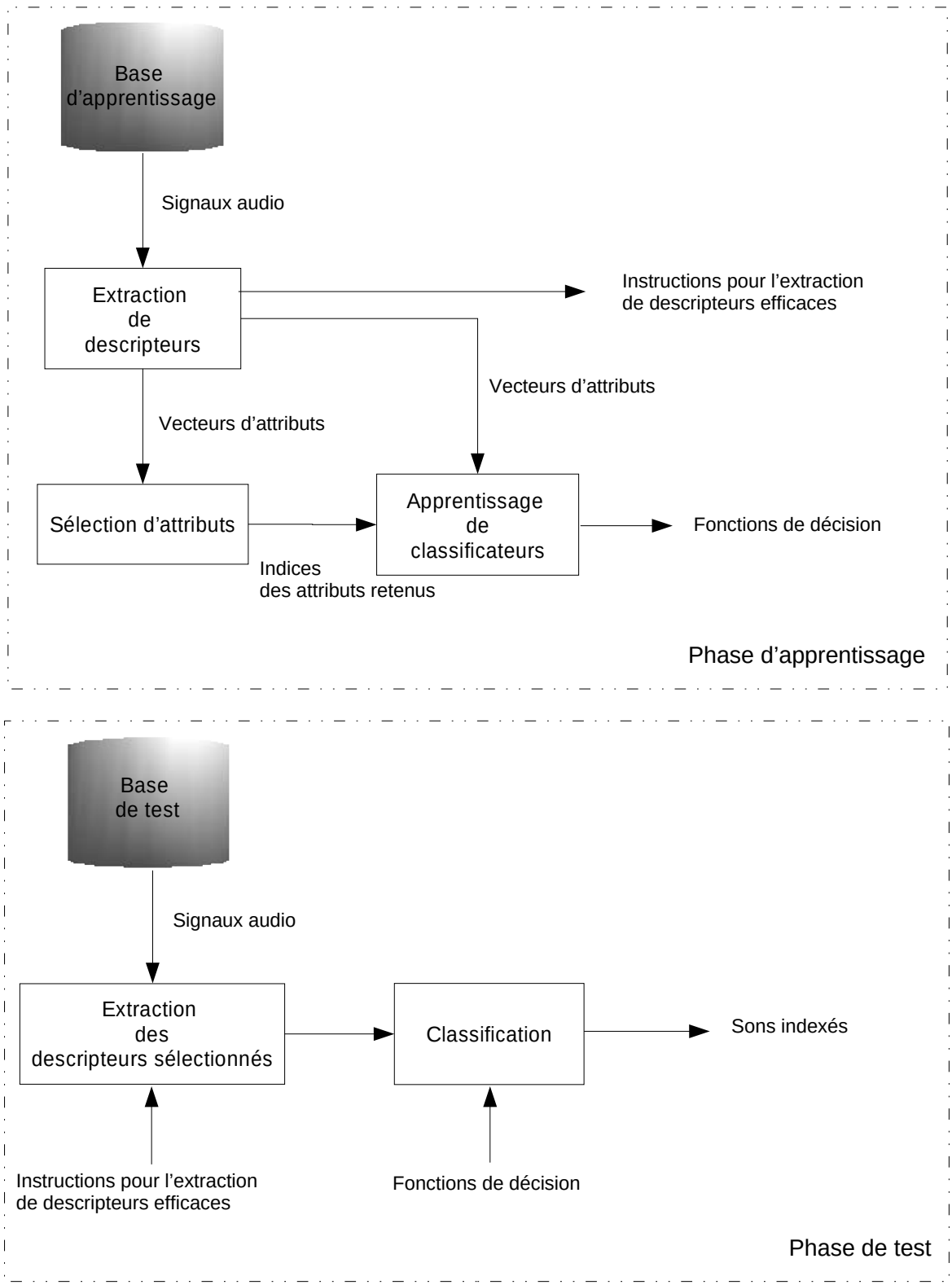

Fig. I.2 Système de classification audio.

Cela donne lieu à un foisonnement d'approches susceptibles d'apporter des solutions au problème posé, et nous nous proposons de contribuer à éclairer la façon dont les différents choix de conception doivent être faits.

Problématique : reconnaissance automatique des instruments de musique

Cette thèse se focalise sur la tâche particulière de la reconnaissance des instruments de musique, abordée au travers d'une approche de classification automatique. Outre le fait de représenter une fin utile en soi, répondant à des requêtes sur l'orchestration des pièces musicales, la possibilité d'identifier les instruments peut servir d'autres problématiques, notamment :

- l'extraction des notes musicales, qui peut bénéficier d'une information sur le nombre de sources musicales en présence et sur les propriétés du spectre des différents instruments ;
- le codage audio paramétrique à bas débit, qui peut adapter les modèles de représentation au contenu instrumental ;
- la synthèse musicale, par concaténation de segments de musique qui ont été préalablement annotés automatiquement.

Le sujet est relativement récent, même s'il s'appuie sur des études de caractérisation du timbre des instruments de musique et de leur perception qui sont quant à elles menées depuis de nombreuses années [Clark *et al.*, 1964,Plomp, 1970,Grey, 1977,Kendall, 1986,Fletcher et Rossing, 1991, Feiten et Ungvary, 1991, DePoli *et al.*, 1993].

Les premières tentatives remontent à une dizaine d'années. La plupart des travaux se sont intéressés à la reconnaissance des instruments à partir de *notes musicales isolées* (en considérant une note à la fois) [Kaminskyj et Materka, 1995, Fraser et Fujinaga, 1999, Martin, 1999, Kaminskyj, 2000, Fujinaga et MacMillan, 2000, Kostek et Czyzewski, 2001a, Agostini *et al.*, 2003, Eronen, 2001a, Peeters, 2003, Krishna et Sreenivas, 2004, Chetry *et al.*, 2005]. Cette approche présente deux avantages majeurs :

- d'abord, la possibilité d'extraire des descripteurs acoustiques sophistiqués qui deviennent difficiles à calculer à partir de phrases musicales impliquant un flux continu de notes (se superposant éventuellement les unes aux autres) ;
- ensuite, plusieurs bases de données publiques de notes isolées [Opolko et Wapnick, 1987, IOWA, 1997, SOL, , Goto *et al.*, 2003] peuvent être utilisées pour ces études.

Cependant, elle comporte aussi plusieurs inconvénients. En effet, l'adoption de ces conditions sur le contenu musical implique la perte d'informations de transition entre notes, connues pour

être particulièrement utiles à l'identification des instruments. De plus, il n'est pas évident que la reconnaissance à partir de notes isolées puisse être exploitée dans un contexte musical réel car il n'est pas toujours possible, au vu de l'état-de-l'art, de procéder efficacement à la segmentation en notes d'un extrait musical, particulièrement en situation de superposition de notes.

Quelques travaux se sont intéressés à la reconnaissance des instruments à partir de *phrases musicales* jouées en *solo* (sans accompagnement)– c'est ce que nous appelons le *cas mono-instrumental*– sans restrictions sur le contenu musical joué [Dubnov et Rodet, 1998, Martin, 1999, Brown, 1999, Marques et Moreno, 1999, Brown *et al.*, 2000, Ventura-Miravet *et al.*, 2003, Livshin et Rodet, 2004a,Livshin et Rodet, 2004b]. Un pas a été ainsi franchi vers des applications réalistes, en traitant des extraits de musique provenant d'enregistrements du commerce.

Des tests de perception ont été entrepris pour tenter de quantifier les capacités humaines à reconnaître les instruments (voir [Brown *et al.*, 2000] pour une synthèse de ces tests). Un test effectué par Martin [Martin, 1999] révèle que le taux de reconnaissance réalisé par les sujets humains (pour la plupart musiciens) est de seulement 67%, lorsqu'ils sont appelés à choisir, parmi 27 instruments possibles, celui qui correspond à l'extrait de 10s de musique mono-instrumentale qu'ils écoutent. Cela donne une idée de la difficulté de la tâche envisagée et des taux de reconnaissance auxquels on peut s'attendre.

Notons que même s'il est possible de retrouver des similarités entre la tâche de reconnaissance des instruments à partir de phrases musicales, et la tâche d'identification du locuteur, qui fait l'objet d'une recherche abondante, des différences notables persistent :

- si l'identité d'un locuteur est unique, un même instrument possède différentes instances, correspondant à des factures différentes de celui-ci, et il peut être joué par différents musiciens qui donnent chacun une empreinte particulière au son produit par l'instrument ;
- il n'existe pas de consensus sur un ensemble d'attributs particulier pouvant garantir de bonnes performances de classification alors que la représentation cepstrale est généralisée pour l'identification du locuteur ;
- il n'existe pas de bases de données communes ni de procédures d'évaluation communes qui permettent de comparer les performances des différents systèmes proposés.

Peu de tentatives ont été effectuées sur la reconnaissance des instruments à partir de musique multi-instrumentale (dans laquelle plusieurs instruments sont joués simultanément). Elles

ont été marquées par de fortes restrictions sur le nombre d'instruments en présence, le type d'instrumentation ou la partition musicale.

Des éléments musicaux assez simples (tels que des notes, des accords de notes ou des mélodies), mixés de façon "artificielle" ont souvent été utilisés dans ces études. Les systèmes proposés relient, dans bon nombre de cas, la tâche de reconnaissance des instruments au problème de la transcription automatique ou la séparation de sources musicales, en requérant que les notes jouées soient connues antérieurement à la phase d'identification des instruments [Kashino et Mursae, 1998, Kinoshita *et al.*, 1999, Kostek, 2004]. Le succès de la tâche est alors intimement lié à l'efficacité de l'étape d'estimation de fréquences fondamentales multiples, problème qui est connu pour être difficile à résoudre.

Quant à l'utilisation d'extraits de musique réaliste, un faible nombre de propositions a été fait. Eggink & Brown ont utilisé une approche par "caractéristiques manquantes" (*missing feature*) pour l'identification de deux instruments joués simultanément [Eggink et Brown, 2003]. Plus récemment, les mêmes auteurs ont présenté un système capable de reconnaître un solo d'instrument en présence d'accompagnement musical après extraction des fréquences fondamentales les plus proéminentes du signal [Eggink et Brown, 2004]. Nous citons également une étude utilisant une analyse en sous-espaces indépendants pour identifier deux instruments dans un duo [Vincent et Rodet, 2004] et une autre utilisant un système de reconnaissance développé dans le contexte mono-instrumental pour identifier à partir de duos, l'un des deux instruments en présence [Livshin et Rodet, 2004a, Livshin et Rodet, 2004b].

Un effort important a été dédié à la conception de descripteurs utiles à la reconnaissance des instruments de musique, incluant des attributs calculés dans des domaines différents (temporel, spectral et perceptuel) ainsi que leur variation et leurs statistiques observées sur des horizons temporels ou fréquentiels choisis. L'effet de combinaison des différents descripteurs a été étudié [Brown *et al.*, 2000, Eronen, 2001b] et des techniques de transformation et de sélection automatique des attributs ont été explorées [Fujinaga, 1998, Martin, 1999, Eronen, 2001a, Peeters et Rodet, 2002, Peeters, 2003].

Différentes stratégies de classification populaires ont été expérimentées (voir [Herrera *et al.*, 2003] pour une synthèse). L'algorithme des K plus proches voisins a été largement utilisé dans les premières études sur la reconnaissance des instruments à partir de notes isolées [Kaminskyj et

Materka, 1995,Fujinaga, 1998,Martin, 1999,Eronen, 2001a,Agostini *et al.*, 2001], mais également à partir de phrases musicales [Livshin et Rodet, 2004b, Livshin et Rodet, 2004a]. L'analyse discriminante a été utilisée à la fois pour un pré-traitement des attributs [Martin, 1999,Peeters, 2003,Livshin et Rodet, 2004b,Livshin et Rodet, 2004a] et pour la classification [Agostini *et al.*, 2001]. Les réseaux de neurones ont été testés dans différentes études (voir [Kostek et Czyzewski, 2001b] par exemple), mais également les modèles de gaussiennes [Martin, 1999, Peeters, 2003] et de mélanges de gaussiennes [Brown, 1999, Brown *et al.*, 2000, Eronen, 2001a], les chaînes de Markov cachées [Lee et Chun, 2002, Kitahara *et al.*, 2003, Eronen, 2003, Ventura-Miravet *et al.*, 2003] et les machines à vecteurs supports [Marques et Moreno, 1999].

Des performances de classification variables ont été rapportées par les différentes études, performances qu'il est difficile de comparer, eu égard à la grande variabilité des conditions expérimentales : du choix des instruments et de leur nombre (de 2 à 27 instruments ont été considérés en fonction des études), aux procédures d'évaluation, en passant par les bases de données utilisées qui sont significativement différentes en taille, diversité et contenu.

Contributions

L'objet de notre travail est de contribuer à améliorer l'identification automatique des instruments dans des contextes réalistes, d'abord sur des solos de musique (sans accompagnement), ensuite sur des pièces multi-instrumentales, sans restrictions sur le contenu musical.

Nous entendons tirer profit de l'effort considérable qui a été consacré dans les travaux précédents à caractériser les instruments de musique, pour concentrer notre attention sur la meilleure exploitation de la masse de descripteurs disponibles, en faisant appel aux outils modernes de l'apprentissage automatique. Nous nous efforçons de rechercher des réalisations performantes des différents modules constituant le système de classification automatique que nous proposons. Cela se traduit d'une part, par l'obtention d'une description adéquate du signal, sous forme d'une sélection efficace d'attributs, d'autre part, par la mise en œuvre d'une stratégie de classification qui permet d'assurer des taux de reconnaissance élevés sur des pièces sonores reflétant la diversité de la pratique musicale et des conditions d'enregistrement rencontrées dans le monde réel.

Plus précisément, nos principales contributions sont les suivantes :

– la constitution d'une base d'extraits de solos d'instruments plus large et plus diversifiée que celles utilisées dans les précédents travaux, qui permet de réaliser un meilleur apprentissage

des classificateurs utilisés, mais surtout une meilleure évaluation des performances du système de reconnaissance, notamment ses capacités de généralisation (*cf.* chapitre II) ;
- la conception de nouveaux descripteurs utiles à la tâche envisagée (*cf.* chapitre IV) ;
- la proposition d'une nouvelle technique de sélection des attributs qui s'avère des plus efficaces à la lumière des résultats de l'étude que nous menons pour comparer plusieurs algorithmes de l'état-de-l'art ; cette technique nous permet en outre de produire une organisation des attributs utilisés pour la classification audio (*cf.* chapitres VI et VIII) ;
- une méthode d'inférence de taxonomies hiérarchiques pour la classification audio, qui est appliquée au problème de la reconnaissance des instruments (*cf.* chapitre IX) ;
- un système de classification des instruments en présence dans les enregistrements où plusieurs instruments sont joués simultanément, qui évite le recours à l'estimation de fréquences fondamentales multiples et ne fait pas de restrictions sur le contenu musical (*cf.* chapitre X).

Organisation du document

Le corps du document est organisé en trois parties :

- dans un premier temps, nous présentons les descripteurs qui sont utilisés dans notre système de classification des instruments, ils sont présentés dans le chapitre IV après que seront indiqués, dans le chapitre III, les pré-traitements réalisés sur le signal audio avant l'extraction de ces descripteurs ;
- nous explorons ensuite les outils de l'apprentissage automatique qui entrent en jeu dans la mise en œuvre de notre système de reconnaissance des instruments, nous commençons par une vue d'ensemble de ces outils et de leurs fondements théoriques (dans le chapitre V), nous nous intéressons spécifiquement à la sélection automatique des attributs, dans le chapitre VI, et au réglage des classificateurs, dans le chapitre VII ;
- nous nous focalisons alors sur la problématique de la reconnaissance des instruments de musique : le chapitre VIII aborde des éléments de caractérisation spécifique à cette problématique, puis un système de classification hiérarchique est proposé pour la reconnaissance des instruments, à partir d'un contenu mono-instrumental, dans le chapitre IX, et à partir de musique multi-instrumentale, dans le chapitre X.

Nous concluons cette partie introductive par une description des bases de données que nous avons constituées pour mener ce travail de thèse. Cette description est donnée dans le chapitre qui suit.

II. Bases de données pour la reconnaissance des instruments de musique

Nous présentons dans ce chapitre, les bases de données sonores que nous utilisons dans notre travail. Deux corpus sont décrits : un corpus de phrases musicales mono-instrumentales et un corpus d'extraits de musique multi-instrumentale. La division de ces corpus en sous-ensembles d'apprentissage, de développement et de test est également spécifiée.

II-1. Introduction

Un effort important a été consacré dans ce travail à la mise en place d'une base de sons instrumentaux, qui puisse être utilisée dans la construction et l'évaluation pertinente d'un système de reconnaissance automatique des instruments. Une telle base doit satisfaire les critères suivants :

- elle doit être de taille assez importante pour permettre un bon apprentissage des classificateurs considérés, et une évaluation des performances qui soit statistiquement significative, c'est-à-dire, telle que les intervalles de confiance soient suffisamment étroits ;
- elle doit permettre de mettre en évidence la capacité de généralisation des systèmes de classification proposés, c'est-à-dire leur capacité à correctement reconnaître les instruments à partir de nouveaux extraits musicaux, faisant intervenir des musiciens et des instances d'instruments distincts de ceux qui sont connus durant la phase d'apprentissage, et notamment enregistrés dans des conditions différentes. Nous utiliserons le terme *source* pour décrire la diversité des enregistrements : une source définit un contexte particulier à partir duquel est obtenu un extrait musical utilisé dans notre étude, de telle sorte que d'une source à l'autre, au moins l'un des trois paramètres "instance de l'instrument", "musicien"

ou “conditions d’enregistrement” varie. Un album peut par exemple constituer “une source”.

Deux corpus de données distincts devaient être constitués :

- un premier corpus de phrases musicales jouées en solo (sans accompagnement) pour l’étude sur la reconnaissance des instruments en contexte mono-instrumental ;
- un deuxième corpus d’extraits de musique multi-instrumentale pour le développement d’un système de reconnaissance des instruments, à partir d’œuvres jouées à plusieurs instruments ;

ils sont décrits dans ce qui suit.

II-2. Corpus mono-instrumental (INS)

Collecter des phrases musicales mono-instrumentales s’avère particulièrement ardu car pour la plupart des instruments, très peu d’œuvres de solo sans accompagnement existent. C’est typiquement le cas pour des instruments tels que le tuba, le basson, le trombone. Une alternative est de procéder à des enregistrements en studio pour les besoins de l’étude, ce que nous avons réalisé pour trois instruments : la clarinette, le saxophone alto et la trompette, au studio de Télécom Paris. Même si elle est intéressante, cette alternative ne résout que partiellement le problème car cela ne permet d’obtenir qu’une source par session d’enregistrement.

Nous avons donc entrepris de collecter des extraits musicaux d’œuvres ou de passages de solo (sans accompagnement) à partir d’enregistrements du commerce en nous fixant pour objectif d’obtenir pour chaque instrument un nombre maximum de sources, tout en assurant une séparation complète entre les sources utilisées dans la phase d’apprentissage et celles utilisées dans la phase de test[1].

Des extraits ont été ainsi obtenus à partir d’enregistrements numériques (CD : Compact Disc) de musique classique, de jazz ou de supports sonores utilisés pour l’enseignement de la musique. Les rares pièces de solo incluses dans la base RWC[2] ont également été exploitées. Ces extraits

[1] nous exigeons qu’en plus de la séparation entre les extraits utilisés pour l’apprentissage et ceux qui sont testés, il y ait une séparation entre les sources dont sont tirés ces extraits.

[2] Il s’agit d’une base de sons musicaux assez variée conçue par des chercheurs japonais pour servir à des travaux sur l’indexation audio [Goto *et al.*, 2002].

ont été encodés en mono (en moyennant les deux canaux gauche et droit) au format PCM[3]. Nous reviendrons sur le choix de la fréquence d'échantillonnage dans la section III-1-A.

Nous nous sommes efforcés de rassembler des solos d'instruments représentant les différentes familles instrumentales : cordes (frappées, pincées et frottées), bois (anches simples et doubles), cuivres et percussions. Des extraits correspondant aux instruments présentés dans le tableau II.1 ont pu être obtenus, à partir d'au moins quatre sources différentes. Nous distinguons la contrebasse jouée *con arco* (Ba)[4] de la contrebasse jouée *pizzicato*[5] car les sons produits dans ces deux configurations sont significativement différents. Nous calculerons néanmoins un taux de reconnaissance unique à partir de ceux obtenus dans ces deux cas (l'instrument étant le même). De plus, nous distinguons les trois saxophones : ténor, alto et soprano. En revanche,

- la classe "clarinette" regroupe des données de la clarinette en Si^b et de la clarinette en Mi^b ;
- la classe "trompette" regroupe essentiellement des extraits de trompette en Do ;
- la classe "trombone" regroupe essentiellement des extraits de trombone ténor.

Notons que pour ces instruments l'information de registre est rarement donnée dans les livrets descriptifs des enregistrements.

Instrument	Code	Instrument	Code
saxophone alto	As	hautbois	Ob
saxophone ténor	Ts	saxophone soprano	Ss
basson	Bo	piano	Pn
contrebasse-*pizzicato*	Bs	contrebasse-*arco*	Ba
clarinette basse	Cb	tuba	Ta
clarinette	Cl	trombone	Tb
violoncelle	Co	trompette	Tr
flûte	Fl	alto	Va
cor	Fh	violon	Vl
guitare acoustique	Gt	batterie	Dr

Tab. II.1 Instruments considérés et les codes que nous leur associons.

Le tableau II.2 résume les caractéristiques du corpus obtenu. On y distingue trois sous-ensembles d'extraits sonores : un *ensemble d'apprentissage*, utilisé comme son nom l'indique

[3]Pulse Coded Modulation

[4]avec l'archet

[5]en pinçant les cordes avec les doigts

dans la phase d'apprentissage, un *ensemble de développement*, utilisé pour effectuer d'éventuels réglages de paramètres durant la phase de *développement des classificateurs* et un *ensemble de test*, qui sert à l'évaluation des performances du système. La répartition des extraits dans ces ensembles a été effectuée pour respecter autant que possible, les contraintes suivantes :

1) disposer d'un ensemble de développement équivalent à l'ensemble d'apprentissage (qui peut être, en cas de besoin, regroupé avec l'ensemble d'apprentissage à la fin du développement) ;
2) utiliser dans l'ensemble de tests, des sources distinctes de celles utilisées dans les ensembles d'apprentissage et de développement ;
3) disposer idéalement d'un minimum de 5 sources pour l'ensemble d'apprentissage/développement et de 5 sources pour l'ensemble de test (au total, au mois 10 sources par instrument) ;
4) pour le test, disposer idéalement de plus de 10 minutes (et au moins de 5 minutes) de musique par instrument afin de permettre une évaluation avec des intervalles de confiance suffisamment étroits (de l'ordre de 0.1% de largeur dans le cas le plus défavorable).

Ces contraintes impliquent que :

– les sources de plus longues durées soient utilisées pour les ensembles d'apprentissage et de développement (puisque ceux-ci peuvent contenir les mêmes sources) ;
– la taille de l'ensemble d'apprentissage peut être inférieure à celle de l'ensemble de test mais la somme des ensembles d'apprentissage et de développement est de taille supérieure à celle de l'ensemble de test.

Elles n'ont malheureusement pas toujours pu être satisfaites du fait de la rareté des extraits pour certains instruments. Ainsi, nous avons dû accepter d'avoir moins de sources et/ou moins de données pour un sous-ensemble d'instruments, en particulier le tuba (Ta), le cor (Fh), le saxophone soprano et la clarinette basse. Notons que dans ce dernier cas nous avons préféré préserver toutes les données pour l'apprentissage, si bien que la reconnaissance de la clarinette basse ne sera pas testée, mais cet instrument fera partie des classes possibles pour le test de tous les autres instruments.

Le corpus obtenu sera désigné par `INS` et ses sous-ensembles d'apprentissage, de développement et de test, respectivement par `INS-A`, `INS-D` et `INS-T`.

Nous n'utiliserons dans certaines expériences préliminaires qu'un sous-ensemble de 8 instruments (pour alléger la charge de calcul), en l'occurrence : le piano, la guitare, le violoncelle, le

Instrument	Sources app./dev.	App.	Dev.	Sources test	Test
Pn	7	22’ 16”	23’	7	14’ 13”
Gt	5	10’ 43”	10’ 37”	5	15’ 58”
Bs	3	7’ 37”	5’ 41”	5	12’ 44”
Ba	3	6’ 44”	8’ 5”	4	6’ 45”
Co	5	15’ 47”	13’ 54”	5	12’ 7”
Va	5	16’ 37”	9’ 35”	5	15’ 57”
Vl	6	34’ 11”	26’ 0”	5	24’ 11”
Ta	2	2’ 49”	0’ 0”	2	1’ 51”
Tb	4	15’ 28”	13’ 41”	4	7’ 1”
Fh	4	3’ 43”	0’ 0”	2	3’ 24”
Tr	5	10’ 46”	11’ 18”	5	11’ 30”
Bo	4	13’ 0”	13’ 43”	4	12’ 14”
Ts	3	11’ 13”	4’ 11”	5	6’ 40”
As	3	20’ 7”	6’ 44”	4	10’ 15”
Ss	2	13’ 49”	0’ 0”	2	7’ 51”
Fl	5	16’ 31”	14’ 15”	5	15’ 56”
Ob	4	14’ 46”	10’ 19”	5	14’ 40”
Cl	5	8’ 34”	9’ 7”	5	13’ 38”
Cb	4	2’ 13”	0’ 0”	0	0’ 0”
Dr	3	3’ 1”	0	1	4’ 24”

Tab. II.2 Notre base de sons mono-instrumentaux. “Sources app./dev.”, respectivement “Sources test”, désigne le nombre de sources distinctes disponibles à l’apprentissage/développement, respectivement au test. “App.”, “Dev.” et “Test” donnent respectivement les durées (en minutes et en secondes) totales des extraits disponibles pour l’apprentissage, le développement et le test. Les instruments en gras font partie du corpus SUB-INS.

violon, la trompette, le cor, le hautbois et la clarinette. Nous désignerons ce sous-corpus par `SUB-INS` et ses sous-ensembles d'apprentissage, de développement et des test, respectivement par `SUB-INS-A`, `SUB-INS-D` et `SUB-INS-T`.

Les propriétés des bases de données utilisées dans d'autres études sur la reconnaissance des instruments à partir de phrases mono-instrumentales sont résumées dans le tableau II.3. Il peut être noté que nous obtenons un corpus plus diversifié et de taille plus importante que les autres études. Cela nous permet d'envisager de réaliser l'apprentissage des classificateurs dans de meilleures conditions mais également de tester de façon plus avancée les capacités de généralisation de nos schémas de classification.

	Classes	Sources	Apprentissage	Test
Brown [Brown *et al.*, 2000]	4	!	0' 54"- 5' 30"	1' - 4'
Martin [Martin, 1999]	11	2 - 8	0' 12" - 35' 30"	0' 54" - 35' 30"
Marques [Marques et Moreno, 1999]	8	2 - 2	3' 25" - 3' 25"	0' 20" - 0' 20"
Miravet [Ventura-Miravet *et al.*, 2003]	6	3 - 9	30' 18" - 34' 4"	15' 45" - 18' 56"
Livshin [Livshin et Rodet, 2004a]	7	!	! - !	! - !
Notre base	19	4-14	2' 13"- 60' 11"	1' 51" - 24' 11"

Tab. II.3 Comparaison des bases de données utilisées dans différentes études - "Classes" est le nombre de classes d'instruments considéré pour lesquelles au moins 2 sources étaient disponibles. "Sources" est le nombre de sources distinctes utilisées. "Apprentissage" et "Test" représentent respectivement les tailles des ensembles d'apprentissage et de test en minutes et secondes ; les durées maximales et minimales sont données. " !" indique une information non clairement déterminée.

II-3. Corpus multi-instrumental (`MINS`)

La difficulté majeure qui est rencontrée dans la construction d'une base de sons pour l'étude sur la reconnaissance des instruments en contexte multi-instrumental, est reliée à la nécessité d'annoter manuellement les segments de musique comprenant des mélanges d'instruments diffé-

rents[6]. En effet, dans un trio composé de piano, contrebasse et batterie par exemple, des segments peuvent impliquer uniquement le piano, uniquement la batterie ou uniquement la contrebasse et la batterie. Un aspect critique de l'annotation concerne la précision avec laquelle les annotateurs réalisent la segmentation. Il est nécessaire de décider de l'horizon temporel minimum devant être utilisé pour la segmentation[7]. Afin de réaliser un compromis entre précision (temporelle) et faisabilité de l'annotation par l'Homme, une longueur minimale de 2s est imposée aux segments annotés en ce sens qu'un nouveau segment est créé s'il implique un changement d'orchestration qui dure au moins 2s.

Nous choisissons de tester notre système sur des ensembles de jazz variant de solos à quartets. Ce choix est motivé par la diversité rencontrée dans ce genre musical que nous estimons représentatif d'un nombre important de compositions musicales. En particulier, nous considérons les ensembles faisant intervenir les instruments suivants : contrebasse, batterie, piano, percussions, trompette, saxophone ténor, guitare électro-acoustique et guitare acoustique. De plus, les voix chantées féminines et masculines sont considérées comme des instruments possibles.

Le tableau II.4 résume les classes de mélanges d'instruments pour lesquelles des données suffisantes ont pu être collectées. Une partie des sons a été extraite à partir d'enregistrements du commerce (en studio ou en *live*). Une autre partie des sons provient de la base de musique jazz RWC [Goto *et al.*, 2002]. Les sons sont encodés en mono aux formats PCM ou mp3 à 64kbps

Il existe toujours une séparation complète entre les données d'apprentissage et les données de test (des extraits distincts sont utilisés dans chaque ensemble) mais aussi une séparation complète, dans la plupart des cas, entre les sources à partir desquelles sont tirés les extraits utilisés dans l'ensemble d'apprentissage et celles utilisées dans l'ensemble de test. Les 2/3 des sons sont inclus dans la base d'apprentissage et le 1/3 restant laissé pour le test, lorsque cela n'est pas en conflit avec la contrainte que les sources d'apprentissage et de test restent distinctes. Quand seulement deux sources sont disponibles, la plus longue est utilisée pour l'apprentissage

[6]L'utilité de disposer d'un alignement des extraits musicaux avec des partitions du type midi est ici mise en évidence. La mise en œuvre de bases de musique annotée constitue un enjeu important pour la communauté de la recherche en indexation audio et les premières tentatives sont prometteuses [Goto *et al.*, 2002].

[7]Il n'est pas réaliste de segmenter la musique à la précision de la fenêtre d'analyse qui ne fait que 32ms de longueur.

et la plus courte pour le test. Enfin, lorsqu'une seule source est disponible, les 2/3 des données de cette source sont utilisés pour l'apprentissage et le 1/3 restant pour le test.

Ainsi, une variabilité importante est introduite dans les données, ce qui permet de tester les capacités de généralisation du système.

Notons qu'étant donnée la procédure d'annotation, nous pouvons nous attendre à un nombre important d'exemples aberrants (*outliers*) parmi les différents ensembles de données. Typiquement, plusieurs segments associés à l'étiquette (contrebasse, batterie, piano et saxophone ténor : BsDrPnTs), contiennent probablement un nombre important de segments de la classe (contrebasse, batterie et piano : BsDrPn).

Ensembles	Sources apprentissage	Apprentissage	Sources test	Test
BsDr	1	6' 51"	4	5' 2"
BsDrPn	11	15' 55"	1	5' 46"
BsDrPnTr	3	5' 10"	1	3' 39"
BsDrPnTs	1	5' 39"	2	1' 30"
BsDrPnVf	4	10'	1	4' 26"
BsDrPnVm	0.5	2' 52"	0.5	1' 26"
BsDrTr	1	3' 1"	1	2' 14"
BsDrTs	1	1' 22"	1	1' 11"
BsEgPn	1	2' 31"	1	0' 42"
BsPn	6	12' 27	1	11' 6"
BsPnVm	1	5' 46"	1	0' 32"
DrGtPrVm	0.5	2' 18"	0.5	1' 9"
EgVf	1	10' 28"	2	3' 16"
GtVf	2	2' 39"	1	2' 49"
PnTr	0.5	6' 38"	0.5	3' 19"
PnVf	6	7' 16"	1	3' 6"
PnVm	1	4' 41"	1	2' 46"
Pn	15	18' 28"	3	12' 30"
Bs	1	2' 14"	5	1' 33"
Dr	2	3' 56"	4	3' 33"

Tab. II.4 Bases de sons multi-instrumentaux utilisée. "Sources apprentissage" et "Sources test" représentent respectivement les nombres de sources distinctes (albums différents) utilisés (0.5 indique qu'une seule source est disponible pour la classe associée et qu'elle est donc utilisée pour fournir les extraits de l'ensemble d'apprentissage et ceux de l'ensemble de test). "Apprentissage" et "Test" indiquent respectivement les longueurs totales (en minutes et secondes) des ensembles d'apprentissage et de test.

Eg	guitare électro-acoustique
Pr	percussions
Vf	voix féminine
Vm	voix masculine
V	voix
W	instrument à vent
M	V, W ou Eg

Tab. II.5 Codes des instruments.

PREMIERE PARTIE

Extraction de descripteurs pour la classification des signaux audio

Introduction de la première partie

De façon générale, l'indexation comme la classification, se basent sur une représentation intermédiaire du contenu à traiter. Cette représentation doit caractériser le contenu et satisfaire des critères dépendant souvent de l'application envisagée [Ellis, 1996, Martin, 1999]. Parmi ces critères, nous retenons par exemple :

- la *pertinence*, qui traduit l'adéquation de la représentation avec les objets qu'elle caractérise, par exemple des propriétés de *vibrato* sont pertinentes dans la caractérisation de notes de violon, mais non pertinentes pour les notes de piano ;
- *la capacité de discrimination*, qui est significative de la spécificité de la représentation et de son efficacité dans la distinction de classes de sons différentes ;
- *le coût d'extraction et de stockage* : une représentation qui peut être calculée et codée efficacement est plus intéressante qu'une autre ;
- *"l'interprétabilité"*, qui qualifie dans quelle mesure une représentation est compréhensible par l'Homme et si elle se prête à une interprétation intuitive (ce qui peut être associé à un degré d'abstraction) ;
- *la scalabilité* qui permet de n'exploiter qu'une sous-partie de la représentation pour obtenir des performances demeurant acceptables.

Cette représentation intermédiaire se compose habituellement d'un ensemble de *descripteurs* appelés encore caractéristiques (*features*), attributs, variables, ou fonctions d'observation, ... qui sont, dans notre cas, une suite de valeurs numériques décrivant des grandeurs associées au signal (obtenues par des mesures), le plus souvent possédant une interprétation physique.

La communauté de l'indexation audio n'est pas parvenue à une représentation consensuelle du contenu qui permette d'atteindre systématiquement des performances satisfaisantes, comme

c'est le cas dans le domaine de la reconnaissance vocale, où la représentation par MFCC (Mel Frequency Cepstral Coefficient, *cf.* section IV-2-A.1) est généralisée. De fait, plusieurs études montrent qu'une paramétrisation basée exclusivement sur les MFCC s'avère inefficace dans la discrimination de nombreuses classes de sons, en particulier des classes instrumentales. Un effort important a été consacré depuis plus d'une dizaine d'années à proposer et étudier des descripteurs utiles à l'indexation audio dans des contextes variés : des études sur la perception du timbre [McAdams *et al.*, 1995] et la classification des sons instrumentaux [Dubnov, 1996, Martin, 1999, Brown, 1999, Brown, 1998, Brown *et al.*, 2000, Eronen, 2001a], au standard de "description de contenus multimédia" MPEG-7 [ISO/IEC, 2001], en passant par l'analyse de scènes sonores [Ellis, 1996], sans oublier les efforts portant sur la discrimination de la parole et de la musique [Scheirer et Slanely, 1997]. Signalons une synthèse efficace, qui nous a été très utile dans notre travail [Peeters, 2004]. Une alternative intéressante a été récemment proposée qui opte pour une génération automatique des descripteurs par programmation génétique [Pachet et Zils, 2003].

Une distinction est classiquement faite entre descripteurs de *bas-niveau* et *descripteurs de haut-niveau* [ISO/IEC, 2001]. Les premiers sont généralement des descripteurs simples (dont la complexité d'extraction reste réduite) et qui ne peuvent pas toujours être clairement associés à une qualité de la source. Les seconds, généralement élaborés à partir d'un ensemble de descripteurs de bas-niveau, caractérisent des concepts moins abstraits, on parle à titre d'exemple de descripteurs de timbre, de rythme, etc. Nous ne faisons pas dans notre travail une telle distinction. Notre approche consiste à examiner, dans un premier temps, un nombre important de descripteurs, de nature différente, pour ensuite utiliser des techniques permettant d'en sélectionner, de façon automatique, un sous-ensemble efficace[8] (*cf.* chapitre VI).

Nous utiliserons le terme "descripteurs" pour désigner un vecteur d'attributs, valeurs scalaires, regroupés selon le type d'analyse effectuée dans le processus de calcul et/ou selon un aspect particulier que nous cherchons à caractériser ; nous parlerons par exemple de descripteurs temporels, de descripteurs de forme spectrale, etc. Notons que ce regroupement n'est effectué que pour servir la présentation puisqu'aucune distinction n'est faite, au niveau des blocs de traitement,

[8] cette sélection de descripteurs pourra être considérée comme un descripteur de haut-niveau associé aux types de classe considérée, par exemple un descripteur de timbre dans le cas de classes instrumentales.

entre les descripteurs : tous les attributs, valeurs scalaires, éventuellement issus de descripteurs (vectoriels) différents (que nous pourrons appeler *paquets d'attributs*), sont traités sur le même plan. Par exemple, le k-ème coefficient MFCC (qui peut être vu comme le k-ème attribut scalaire du descripteur vectoriel MFCC) et le descripteur (ou attribut) scalaire, fréquence de coupure (*cf.* section IV-2-B.3), sont traités pareillement.

Avant de présenter les descripteurs que nous avons expérimenté (au chapitre IV), nous commençons par décrire, au chapitre III, les pré-traitements réalisés sur le signal audio, antérieurement à l'étape d'extraction de ces descripteurs.

III. Pré-traitements et segmentation des signaux audio

Nous présentons dans ce chapitre les divers outils de traitement qui interviennent dans la phase de description du signal audio. Ceux-ci servent à obtenir une version intermédiaire du signal à partir de laquelle sont calculés les différents descripteurs qui seront présentés au chapitre IV.

III-1. Paramètres et outils d'analyse du signal

A. Fréquence d'échantillonnage

Nous traitons des signaux sous-échantillonnés à 32kHz (possédant par suite une bande passante de 16kHz). Cette fréquence d'échantillonnage est souvent retenue en codage et indexation audio puisqu'elle permet une réduction de la complexité des traitements à suivre (par rapport à la fréquence standard du Disque Compact (CD) de 44.1kHz) tout en préservant un signal audio de haute qualité (souvent désignée par qualité FM). S'il est vrai que certains instruments de musique sont capables de produire des composantes spectrales à des fréquences supérieures à 16kHz, des études précédentes sur le sujet ont mis en évidence que la qualité FM était suffisante pour la reconnaissance des instruments [Martin, 1999, Brown, 1999]. Des tests complémentaires ont même montré que le passage à des fréquences d'échantillonnage plus petites (jusqu'à 11.05kHz) ne provoquait pas de dégradations significatives des performances de classification des instruments [Brown, 1999] (pour l'ensemble de descripteurs choisi). Le choix de 32kHz représente un bon compromis[1], qui permet de réduire la complexité tout en préservant le contenu spectral aux

[1] nous estimons de plus que les tendances actuelles reflètent une préférence marquée pour le codage de haute qualité conforté par l'augmentation des capacités de stockage et des débits de transmission.

fréquences supérieures à 5kHz, contenu que nous nous gardons la possibilité de représenter par des descripteurs.

B. Fenêtres d'analyse temporelle

Les propriétés spectrales et temporelles du signal audio $s(n)$ varient de façon significative dans le temps et cette variation est dépendante de la source sonore étudiée. C'est la raison pour laquelle le signal est habituellement analysé sur des horizons temporels assez courts, de l'ordre de la durée de stationnarité du signal.

De fait, il est nécessaire de réaliser un compromis entre résolution temporelle et résolution fréquentielle, la première contrainte demandant l'utilisation de fenêtres d'analyse temporelles courtes et la seconde le recours à des fenêtres longues. Il s'agit d'une problématique largement étudiée dans des travaux précédents et particulièrement en codage audio [Moreau, 1995]. En conséquence nous adoptons en premier lieu, comme de nombreuses études en indexation audio, un fenêtrage hérité du codage, à savoir l'utilisation de *fenêtres d'analyse recouvrantes* de taille $N = 1024$ échantillons, correspondant à une durée de 32ms à la fréquence d'échantillonnage f_s=32kHz, avec un pas d'avancement $H = 512$ échantillons (16ms). Ce choix réalise un compromis temps/fréquence acceptable sans négliger la contrainte de faible complexité (choix d'un faible recouvrement : 50% de recouvrement, et d'une taille de fenêtre en puissance de deux, permettant l'utilisation de la Transformée de Fourier Rapide (FFT- Fast Fourier Transform)).

S'il demeure satisfaisant pour l'extraction de la majorité des descripteurs, le choix de la taille de fenêtre précédent peut être limitant pour la représentation de caractéristiques particulières. Ainsi, pour décrire des phénomènes de durée plus longue que la durée de stationnarité, par exemple le *trémolo* ou le *vibrato*, nous utilisons des fenêtres d'analyse plus longues, de taille $N_l = 30N$, correspondant à une durée de 960ms avec un recouvrement de 50%. Dans la suite nous désignerons les premières fenêtres (32ms) par *fenêtres courtes* et les deuxièmes (960ms)

par *fenêtres longues*. Sauf indication contraire, les fenêtres courtes seront utilisées par défaut.

Nous adoptons les notations suivantes :

– $x(n, m)$ dénote le segment du signal $s(n)$ correspondant à la fenêtre d'analyse **courte** m :

$$x(n, m) = R_N(n - Hm)s(n), \tag{III.1}$$

où $R_N(n)$ est la fenêtre rectangulaire :

$$R_N(n) = \begin{cases} 1, & n = 0, 1, \ \ldots, \ N-1, \\ 0, & \text{sinon.} \end{cases} \tag{III.2}$$

– $x_l(n, m)$, dénote le segment du signal $s(n)$ correspondant à la fenêtre d'analyse **longue** m. Pour alléger les notations, l'indice de fenêtre m pourra être omis.

C. Analyse spectrale

Pour l'extraction de la plupart des descripteurs spectraux, nous utilisons la Transformée de Fourier à Court Terme (TFCT) [Nawab et Quatieri, 1988] exploitant une fenêtre de pondération de Hamming définie par :

$$W_N(n) = \alpha + (1-\alpha)\cos\left(\frac{2\pi n}{N}\right), \ 0 \leq n \leq N-1 \tag{III.3}$$

avec α=27/50.

Cette transformée présente une résolution fréquentielle constante (de 31.25Hz dans le cas des fenêtres courtes que nous utilisons), ce qui n'est pas toujours satisfaisant pour l'analyse des signaux musicaux. En effet, il est nécessaire de disposer dans ce contexte d'une meilleure résolution en basse fréquence (moins de 2Hz pour séparer les notes de piano les plus basses autour de 27Hz) alors qu'une résolution grossière est suffisante pour la région des hautes fréquences (supérieure à 200Hz, pour distinguer un La7 d'un Si^{b}7). C'est ce qui motive le recours à une transformée à résolution variable, dite à facteur de qualité $\mathcal{Q}$ constant, où $\mathcal{Q} = \frac{f}{\delta f}$ est le rapport entre la fréquence f et la résolution δf. Nous adoptons la stratégie proposée par Brown pour le calcul d'une telle transformée [Brown, 1991] désignée par CQT (Constant $\mathcal{Q}$ Transform). La k-ème composante fréquentielle $X(k)$ du signal $x(n)$ est alors obtenue selon :

$$X(k) = \frac{1}{N_k} \sum_{n=0}^{N_k-1} W_k(n) x(n) \exp\left(-j2\pi \mathcal{Q} n / N_k\right), \tag{III.4}$$

$W_{k,n}$ étant une fenêtre d'analyse de taille $N_k = \frac{f_s}{f_k}\mathcal{Q}$ qui varie en fonction de la fréquence f_k de $X(k)$. Les fréquences f_k sont choisies telles que :

$$f_k = (2^{1/r})^k f_{min}, \tag{III.5}$$

où f_{min} peut correspondre à la fréquence de la note la plus basse admise (nous fixons f_{min} à 27,5Hz), auquel cas, en prenant $r = 12$ dans (III.5) les bins fréquentiels k correspondent aux fréquences fondamentales des notes de la gamme tempérée ($\mathcal{Q} \simeq 17$).

Dans nos expériences, nous utilisons une implémentation en Matlab de la CQT, fournie par Brown [Brown,] et nous faisons varier le paramètre $\mathcal{Q}$. Nous utilisons là aussi des fenêtres d'analyse de Hamming en limitant leur taille maximale à 1024 échantillons.

D. Transformée en Ondelettes Discrète (TOD)

La Transformée en Ondelettes Discrète (TOD) projette le signal sur une base de signaux (appelés *ondelettes*) qui, contrairement aux vecteurs de base de Fourier, peuvent avoir un support variable [Mallat, 2000]. Ces ondelettes, sont obtenues par dilatations et translations d'une *ondelette mère* $\phi(n) \in L_2$. Par exemple,

$$\phi_{s,u}(n) = 2^{-s/2}\phi(2^{-s}n - u);\ s \in \mathbb{Z}, u \in \mathbb{Z}, \tag{III.6}$$

définit une famille d'ondelettes $\{\phi_{s,u}(n)\}$, reliées par des dilatations choisies de façon dyadique. Lorsque l'*échelle* s augmente, l'ondelette de dilate par un facteur de 2, et lorsque u augmente, l'ondelette de décale vers la droite.

Les ondelettes $\{\phi_{s,u}(n)\}$ sont ainsi construites de manière à permettre une analyse du signal en multi-résolution avec un bon compromis temps/fréquence. L'idée est de se doter d'une bonne résolution temporelle pour l'analyse du contenu haute-fréquence et d'une bonne résolution fréquentielle dans la région des basses fréquences.

Plusieurs choix de $\phi(n)$ sont possibles. Nous invitons le lecteur à consulter [Mallat, 2000] pour de plus amples détails.

E. Calcul de l'enveloppe d'amplitude

L'enveloppe d'amplitude temporelle renferme de l'information spécifique à la source sonore (propriétés de l'attaque, trémolo, etc.) qu'il est intéressant de représenter. Nous utilisons une méthode inspirée de [Berthomier, 1983] pour obtenir l'enveloppe.

Nous commençons par calculer le signal analytique $y_l(n)$ associé au signal $x_l(n)$ (observé sur une fenêtre d'analyse longue) :

$$y_l(n) = x_l(n) + i\Psi_l(n), \tag{III.7}$$

où $\Psi_l(n)$ est la transformée de Hilbert[2] du signal $x_l(n)$. Nous déduisons alors l'enveloppe

[2]la transformé de Hilbert peut être vue comme une opération de filtrage par un filtre de réponse fréquentielle $-j\ \mathrm{sgn}(f_k)$, où $j^2 = -1$.

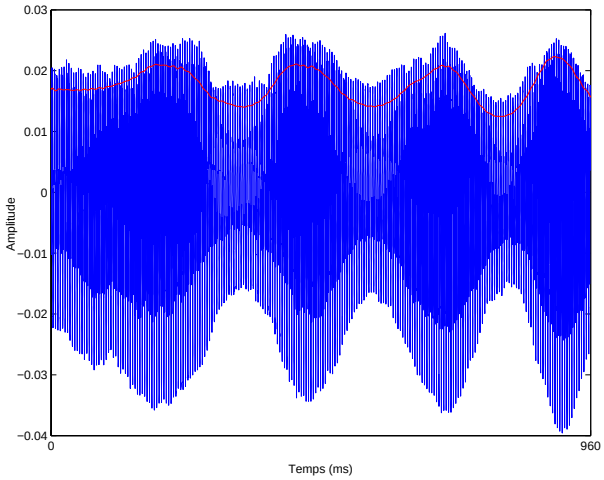

Fig. III.1 Enveloppe d'amplitude (en rouge) extraite à partir d'un signal de violon (en bleu).

d'amplitude $\nu_l(n)$ selon :

$$\nu_l(n) = |y_l(n)| * h(n), \tag{III.8}$$

où $h(n)$ est une demi-fenêtre de Hanning de taille 50ms qui est utilisée comme filtre passe-bas.

III-2. Normalisation du signal

Afin de limiter l'effet de conditions d'enregistrement variables sur les performances de classification, nous reprenons deux opérations de normalisation de la forme d'onde du signal qui ont été utilisées dans des études précédentes [Eronen, 2001a]. La version normalisée $\hat{s}(n)$ s'obtient en faisant :

1) $\tilde{s}(n) = s(n) - \bar{s}(n)$, où $\bar{s}(n)$ est une estimation de la moyenne de $s(n)$ ($\bar{s}(n) = \frac{1}{L}\sum_{n=0}^{L-1} s(n)$, avec L la longueur du signal) ;
2) $\hat{s}(n) = \frac{\tilde{s}(n)}{\max_n |\tilde{s}(n)|}$.

III-3. Segmentation du signal

A. Détection des segments de silence

La détection de segments de silence est un problème qui a été largement étudié, en particulier dans le domaine du traitement de la parole (voir par exemple [Atal et Rabiner, 1976]). On fait généralement appel à des modèles de probabilité gaussiens pour représenter les observations

de paramètres (coefficients d'autocorrélation, log-énergie, résiduel LPC[3], etc.) relatives à des fenêtres de silence et de non-silence. Ces modèles sont utilisés pour détecter les fenêtres de silence d'un nouveau signal en assignant les observations de paramètres correspondant à ces fenêtres à la classe silence, si leur vraisemblance par rapport à cette classe est en dessous d'un seuil préfixé.

Dans le cas de nos signaux, qui sont enregistrés dans des conditions idéales (sans bruit de fond), une approche beaucoup plus simple est satisfaisante. Nous nous basons sur les critères heuristiques suivants : sont considérés comme fenêtres de silence :

- toutes les fenêtres présentant une amplitude maximale (en valeur absolue) mille fois plus petite que le maximum global (30dB en dessous) de $|\hat{s}(n)| = 1$;
- les fenêtres présentant une valeur d'amplitude constante ;
- une succession de moins de 15 fenêtres de non-silence entre deux segments de silence.

B. Détection des segments d'attaques

Les caractéristiques des attaques de notes de musique sont connues pour être des éléments importants de différenciation des timbres d'instruments. Nous nous intéressons par suite à des techniques permettant de détecter les transitoires d'attaque afin de pouvoir envisager un traitement particulier de ces éléments du son.

Plusieurs méthodes ont été proposées dans des travaux précédents (voir [Bello *et al.*, 2005] pour une synthèse). Nous avons pour notre part exploré différentes techniques, des plus simples, se basant sur la variation de l'énergie du signal en amont et en aval de l'attaque, aux plus élaborées, conçues dans le contexte de la détection du rythme et qui font appel à une analyse du signal par banc de filtre (*cf.* [Klapuri, 1999] par exemple). Nous avons retenu une approche qui a été développée par Leveau & Daudet [Leveau *et al.*, 2004] avec qui nous avons collaboré sur cette problématique. Cette approche détecte des instants d'attaque et sélectionne, à partir de ces instants un nombre fixe de fenêtres comme faisant partie du segment transitoire.

L'algorithme de détection des transitoires utilise une fonction de détection basée sur une différence spectrale qui prend en compte un incrément de phase. La version originale de cette

[3] Linear Prediction Coding

méthode a été introduite dans [Bello *et al.*, 2004]. En supposant que le signal se compose de sinusoïdes stationnaires, l'incrément de phase est constant sur deux fenêtres successives : $\phi(k,m) - \phi(k,m-1) = \phi(k,m-1) - \phi(k,m-2)$, et la prédiction au premier ordre du spectre $X(k,m)$, à la fréquence k et sur la fenêtre m est :

$$\hat{X}(k,m) = |X(k,m-1)| \exp\{j[2\phi(k,m-1) - \phi(k,m-2)]\}. \tag{III.9}$$

Lorsqu'un transitoire apparaît, cela provoque une rupture de la "prédictibilité" qui se traduit par un maximum local sur l'erreur de prédiction $\rho(m)$, définie par :

$$\rho(m) = \Sigma_{k=1}^{K} |\hat{X}(k,m) - X(k,m)|. \tag{III.10}$$

Pour une meilleure localisation des instants d'attaques, Leveau & Daudet préconisent l'utilisation d'une fonction de décision modifiée $\gamma(m)$, définie par :

$$\gamma(m) = \max(\delta\rho(m), 0), \tag{III.11}$$

où δ dénote une dérivation temporelle. Les maxima locaux de cette fonction de détection qui se retrouvent au-dessus d'un seuil sont sélectionnés, et les fenêtres correspondantes considérées comme des fenêtres de transitoire d'attaque. Le seuil utilisé est fixé de façon adaptative selon :

$$\theta(m) = \theta_{static} + \lambda \text{ médiane}\{\gamma(m-S), ..., \gamma(m+S)\}, \tag{III.12}$$

où θ_{static} permet de contrôler le compromis entre les fausses détections et les faux rejets de transitoires (fixé à 0.1), S dénote le nombre de fenêtres, précédent et suivant la fenêtre en cours, qui sont utilisées pour l'adaptation du seuil (fixé à 10), et λ permet de "balancer" les deux termes du membre droit de (III.12).

La fenêtre contenant l'attaque ainsi qu'un nombre fixé (de 2 à 4) de fenêtres qui la suivent constituent ainsi un segment que nous marquons comme transitoire.

Cet algorithme se montre performant en comparaison avec d'autres algorithmes de l'état-de-l'art. Pour plus de détails nous invitons le lecteur à consulter [Leveau *et al.*, 2004,Leveau, 2004].

IV. Descripteurs pour la classification audio

Nous présentons dans ce chapitre les descripteurs que nous avons examiné. Ceux-ci ayant fait l'objet d'une littérature abondante, nous adoptons une présentation succincte indiquant brièvement la procédure de calcul et proposant, si possible, une interprétation physique.

Il est important de noter que l'utilisation et le calcul des descripteurs sont rarement rigoureusement justifiés. Les approches suivies sont, en effet, purement heuristiques. Cela n'est pas gênant dans la mesure où nous envisageons une étape de sélection automatique des descripteurs efficaces, à l'issue de la phase d'extraction. De fait, les descripteurs que nous décrivons ici doivent être considérés comme des candidats qui ne seront pas tous retenus dans le schéma de classification.

IV-1. Généralités

Les attributs que nous avons retenus sont mesurés sur des fenêtres d'analyse temporelles successives : il s'agit de *descripteurs instantanés*. La plupart de ces attributs sont calculés sur les fenêtres d'analyse courtes. Ceux qui sont calculés sur les fenêtres longues sont répétés sur autant de fenêtres courtes correspondant au même segment de signal analysé. Cela permet d'intégrer les différents attributs (issus de fenêtres d'analyse courtes ou longues) au sein d'un même vecteur d'observation associé à une fenêtre d'analyse courte. La figure IV.1 illustre cette opération.

Nous avons choisi des attributs qui peuvent être extraits de façon robuste et systématique à partir d'un contenu audio quelconque, éventuellement polyphonique (plusieurs notes simultanément), impliquant des instruments percussifs (par exemple de la batterie), et/ou bruité (enregistrements en direct ou *Live*, compression du signal, etc.). De telles conditions rendent difficile l'extraction de fréquences fondamentales multiples, ce qui explique que les attributs calculés à partir de ces dernières (par exemple, l'inharmonicité, la déviation harmonique ou le tristimulus [Peeters, 2004]) ont été évités.

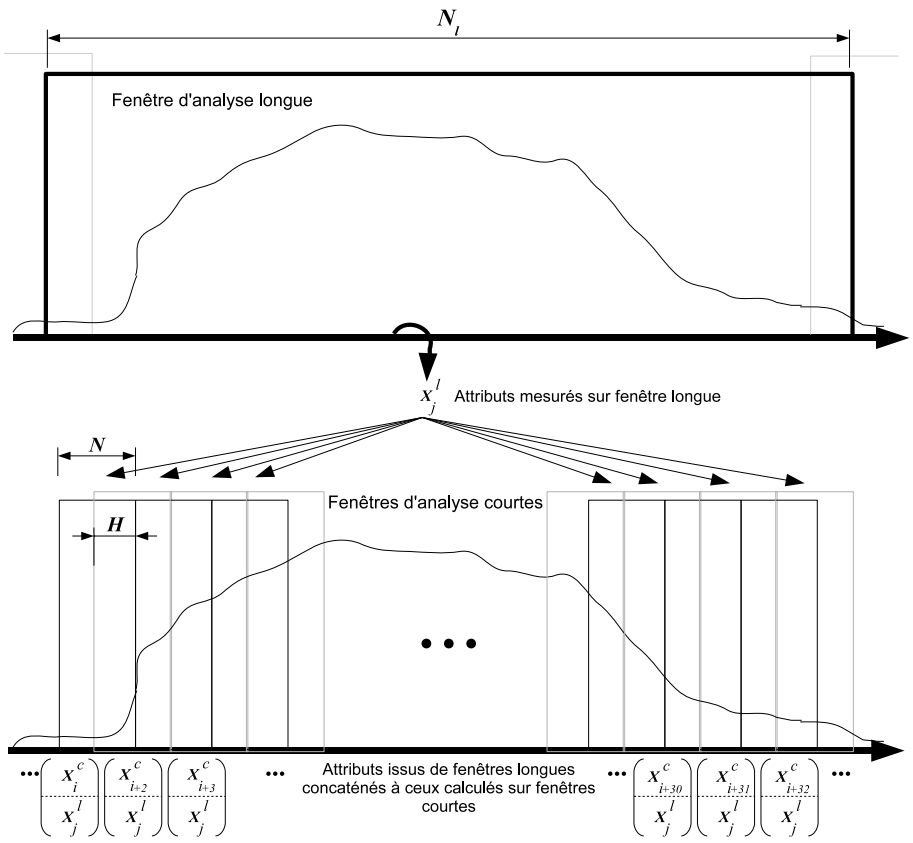

Fig. IV.1 Intégration des descripteurs issus de fenêtres longues et courtes au sein des vecteurs d'observation.

IV-2. Descripteurs classiques

A. Descripteurs cepstraux

Le *cepstre* (réel) s'obtient comme la transformée de Fourier inverse du logarithme du spectre d'amplitude $|X(k)|$:

$$c(p) = \sum_k \log |X(k)| \exp\left(j2\pi \frac{k}{N} p\right),\ p = 1 \dots P \tag{IV.1}$$

Dans une modélisation source-filtre du signal :

$$s(n) = g(n) * h(n), \tag{IV.2}$$

où $g(n)$ est l'excitation et $h(n)$ le filtre, il est montré que les coefficients cepstraux correspondants aux basses *quéfrences* p représentent la contribution du filtre $h(n)$ [d'Alessandro, 2002].

C'est ce qui explique le succès de ce descripteur pour différentes tâches reliées au traitement de la parole. En effet, on dispose dans ce cas de modèles de production assez simples, considérant grossièrement les impulsions de la glotte comme une excitation périodique (source) et le conduit vocal comme un résonateur (filtre).

Cela n'est malheureusement pas le cas pour la plupart des signaux audio et particulièrement pour les instruments de musique, pour lesquels on ne dispose pas de tels modèles de production. Cependant, la représentation cepstrale s'avère efficace pour de nombreuses tâches de classification audio telles que la discrimination parole/musique, la reconnaissance du genre ou encore la reconnaissance des instruments, etc. Il reste que s'il est raisonnable, dans certains cas, de faire référence à un modèle source-filtre (par exemple pour les instruments à vents, notamment les bois [Brown, 1999]), il est souvent difficile de justifier l'utilisation du cepstre pour la classification autrement qu'en considérant qu'il s'agit d'une version lissée de l'enveloppe spectrale.

On associe classiquement aux coefficients cepstraux leurs dérivées temporelles premières (δc) et secondes ($\delta^2 c$), ce qui permet de suivre l'évolution de l'enveloppe au cours du temps. Ces dérivées sont obtenues en utilisant une approximation polynômiale à l'ordre 2 de la trajectoire spectrale (cepstres calculés sur une succession de fenêtres d'analyse). Les détails du calcul peuvent être trouvés dans [R. Rabiner, 1993].

Nous explorons différentes variantes de cette représentation cepstrale qui sont décrites ci-après.

1) Mel-Frequency Cepstral Coefficients (MFCC)

Les MFCC [Davis et Mermelstein, 1980] s'obtiennent en considérant, pour le calcul du cepstre, une représentation fréquentielle selon une échelle perceptive appelée l'échelle des *fréquences MEL*, dont une expression analytique peut être donnée par :

$$\mathrm{m}(f) = 2595 \log\left(1 + \frac{f}{700}\right), \tag{IV.3}$$

où f est la fréquence en valeurs linéaires.

Pour ce faire, nous utilisons un banc de filtres triangulaires MEL. Nous intégrons le spectre d'amplitude $|X(k)|$ par bandes MEL, pour obtenir un spectre d'amplitude modifié $\tilde{a}_m$, $m = 1 \ldots M_l$, où $\tilde{a}_m$ représente l'amplitude dans la bande m. Les MFCC s'obtiennent alors par une transformée en cosinus discrète inverse (type II) du logarithme de $\tilde{a}_m$:

$$\tilde{c}(p) = \sum_{m=1}^{M_l} \log\left(\tilde{a}_m\right) \cos\left[p(m - \frac{1}{2})\frac{\pi}{M_l}\right]. \tag{IV.4}$$

Nous utilisons deux bancs de filtres qui diffèrent par le nombre de bandes MEL considérées, M_l (ils sont représentés dans la figure IV.2). Nous calculons les MFCC en prenant $M_l = 30$ sous-bandes, de largeur 119 MEL (ce que nous désignons par le symbole Cp), mais aussi en utilisant $M_l = 11$ sous-bandes, de largeur 325 MEL (ce que nous désignons par le symbole Cc). Nous gardons dans les deux cas les P premiers MFCC, avec $P = 11$ (à l'exception du coefficient $c(0)$) ainsi que les vitesses et accélérations des coefficients $c(0)$ à $c(10)$.

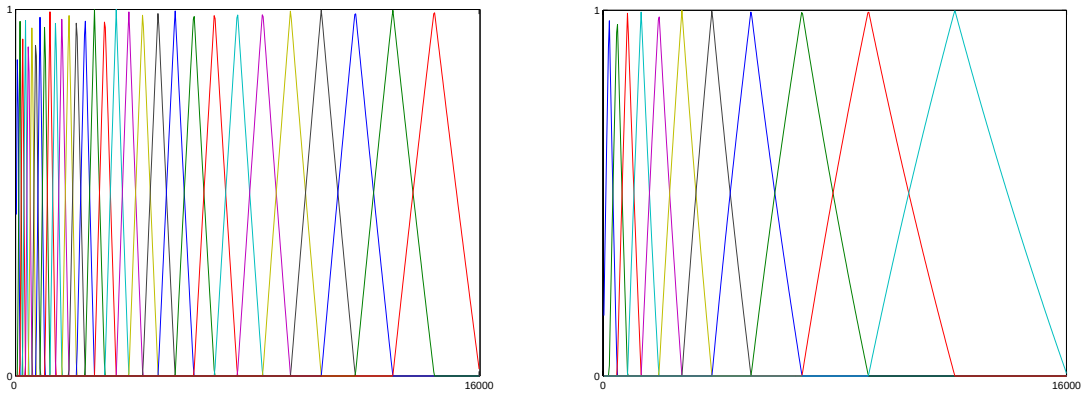

Fig. IV.2 Réponses fréquentielles de bancs de filtres MEL, avec 30 sous-bandes (à gauche) et 11 sous-bandes (à droite).

Signalons que nous utilisons la toolbox Voicebox [Brooks,], en Matlab, pour le calcul des MFCC.

2) Coefficients Cepstraux à partir de la CQT

Brown propose de remplacer, dans le calcul du cepstre, le spectre MEL, $\tilde{a}_m$, par un spectre CQT aligné sur une gamme musicale tempérée et utilisant une résolution d'un tiers d'octave (correspondant grossièrement à celle de l'oreille sur une bonne partie du spectre). Nous suivons cette approche en considérant différentes alternatives. Quatre représentations cepstrales sont ainsi calculées en utilisant des résolutions d'une octave (notée uCq), une demi-octave (notée dCq), un tiers d'octave (notée tCq) et un quart d'octave (notée qCq). En considérant une limite inférieure en fréquence de 27.1Hz (note la plus basse du piano), nous pouvons calculer au maximum 9 coefficients cepstraux uCq et nous gardons les 10 premiers coefficients cepstraux de dCq, tCq et qCq. Nous calculons également les dérivées temporelles premières et secondes de ces coefficients.

B. Descripteurs spectraux

1) Moments spectraux

Les *moments spectraux* permettent de représenter différentes caractéristiques de forme spectrale. Ils ont été utilisés avec succès notamment pour la transcription automatique de boucles de batterie [Gillet et Richard, 2004] et la reconnaissance des instruments de musique. A partir des moments μ_i définis par :

$$\mu_i = \frac{\sum_{k=0}^{K-1}(f_k)^i a_k}{\sum_{k=0}^{K-1} a_k}, \tag{IV.5}$$

avec a_k l'amplitude de la k-ème composante fréquentielle du spectre $X(k)$, de fréquence $f_k = \frac{k}{N}$, les attributs suivants sont définis :

- le *centroïde spectral*, décrivant le centre de gravité du spectre :

$$S_c = \mu_1; \tag{IV.6}$$

  il sert à caractériser la "brillance" d'un son ;

- la *largeur spectrale*, décrivant l'étendue du spectre autour de sa moyenne :

$$S_w = \sqrt{\mu_2 - \mu_1^2}; \tag{IV.7}$$

- l'*asymétrie spectrale*, définie à partir de la *skewness* :

$$S_a = \frac{2(\mu_1)^3 - 3\mu_1\mu_2 + \mu_3}{S_w^3}; \tag{IV.8}$$

  elle permet de représenter la symétrie du spectre autour de sa moyenne ;

– la *platitude spectrale* définie à partir du *kurtosis* :

$$S_k = \frac{-3\mu_1^4 + 6\mu_1\mu_2 - 4\mu_1\mu_3 + \mu_4}{S_w^4} - 3, \tag{IV.9}$$

elle est d'autant plus grande que le spectre est "piqué" autour de sa moyenne, S_k est nulle pour une forme gaussienne (dont le kurtosis vaut 3), positive pour une forme plus piquée et négative pour une forme plus plate.

Nous calculons ces différents attributs ainsi que leur dérivées temporelles premières et secondes afin de suivre la variation de la forme spectrale dans le temps. Ces dérivées sont calculées de la même façon que celle utilisée pour le calcul des dérivées temporelles du cepstre.

2) Mesures de platitude et de crête spectrales

Une alternative à la description de la platitude spectrale par kurtosis peut être obtenue en exploitant le rapport entre la moyenne géométrique et la moyenne arithmétique de l'amplitude spectrale [ISO/IEC, 2001] :

$$S_o = \frac{\prod_k {a_k}^{1/K}}{\frac{1}{K}\sum_k a_k}. \tag{IV.10}$$

Celui-ci peut être vu comme une mesure "d'anti-tonalité" en ce sens que S_o est proche de 0 pour un signal tonal et proche de 1 pour un signal de bruit (S_o=1 pour un spectre plat[1]).

Cet attribut fait partie de la panoplie des descripteurs de bas niveau du standard MPEG-7 [ISO/IEC, 2001] dans lequel il est désigné par *Amplitude Spectral Flatness* (ASF). En fait, le standard recommande de mesurer S_o de façon plus précise sur un ensemble de sous-bandes fréquentielles correspondant à des intervalles d'un-quart d'octave. Nous mesurons une valeur de platitude globale et nous utilisons une implémentation de MPEG-7 (en Matlab) pour mesurer l'ASF sur un total de 23 sous-bandes.

Il est également possible de décrire la "platitude" spectrale au moyen du facteur de crête spectrale [Peeters, 2004] (SCF-*Spectral Crest Factor*) défini, dans la sous-bande sb, comme le rapport entre la valeur maximale du spectre d'amplitude et la moyenne arithmétique de ce dernier :

$$SCF(sb) = \frac{\max_{k\in sb} a_k}{\frac{1}{K}\sum_{k\in sb} a_k}. \tag{IV.11}$$

Nous utilisons le même banc de filtre en quart-d'octave pour mesurer 23 coefficients SCF.

[1] rappelons qu'une moyenne géométrique est toujours plus petite qu'une moyenne arithmétique et qu'on a l'égalité pour une série de valeurs constantes, c'est ce qui explique que $S_o \leq 1$ et S_o=1 pour un spectre plat.

3) Autres descripteurs de la forme spectrale

Coefficients LPC (Linear Prediction Coding) Nous effectuons une analyse Auto-Regressive (AR) à l'ordre 2 du signal, et nous utilisons les deux premiers coefficients du filtre AR obtenu (excepté la constante 1) comme attributs pour décrire de façon grossière l'enveloppe spectrale du signal. Nous faisons implicitement l'approximation que la réponse fréquentielle de ce filtre AR représente l'enveloppe.

Pente Spectrale (Ss) La *pente spectrale* est obtenue au moyen d'une régression linéaire du spectre d'amplitude [Peeters, 2004], elle est donnée par :

$$Ss = \frac{K\sum_{k=1}^{K} f_k a_k - \sum_{k=1}^{K} f_k \sum_{k=1}^{K} a_k}{K\sum_{k=1}^{K} f_k^2 - \left(\sum_{k=1}^{K} a_k\right)^2}. \tag{IV.12}$$

Elle permet de mesurer le taux de décroissance spectrale.

Décroissance spectrale (Sd) Elle est donnée par [Peeters, 2004] :

$$Sd = \frac{1}{\sum_{k=2}^{K} a_k} \sum_{k=2}^{K} \frac{a_k - a_1}{k-1}. \tag{IV.13}$$

Variation temporelle du spectre (Sv) Aussi connue sous le nom de *flux spectral* [Scheirer et Slanely, 1997], elle permet de caractériser la vitesse de variation du profil spectral par le calcul d'une corrélation normalisée entre spectres correspondant à des fenêtres d'analyse successives :

$$Sv = 1 - \frac{\sum_{k=1}^{K} a_k(t-1)a_k(t)}{\sqrt{\sum_{k=1}^{K} a_k(t-1)^2}\sqrt{\sum_{k=1}^{K} a_k(t)^2}}. \tag{IV.14}$$

Fréquence de coupure (Fc) Nous la calculons comme la fréquence en dessous de laquelle 99% de l'énergie spectrale est prise en compte. Elle permet de révéler, par exemple, des informations de registre, un son plus grave présentant une fréquence de coupure plus basse.

Irrégularité spectrale (Si) Il s'agit de représenter les relations entre les partiels d'un son musical. Différentes façons de calculer cette caractéristique ont été proposées. Nous adoptons l'approche de Brown [Brown *et al.*, 2000] qui ne demande pas une étape d'estimation des partiels. L'irrégularité Si est alors calculée comme la dérivée fréquentielle du module de la CQT $X(k)$ du signal (calculée avec une résolution d'un tiers d'octave) :

$$Si(k) = X(k+1) - X(k),\ 0 \leq k \leq 20. \tag{IV.15}$$

C. Descripteurs temporels

1) Taux de passage par zéro ou Zero Crossing Rates (ZCR)

Il s'agit d'une mesure de la fréquence de passage de la forme d'onde temporelle par l'axe d'amplitude nulle [Kedem, 1986]. Le taux de passage par zéro permet de discriminer les signaux bruités (qui présentent des valeurs ZCR élevées) des signaux non bruités (faibles valeurs ZCR). Ainsi, le ZCR permet par exemple de distinguer les sons voisés des sons non-voisés. Il est utile dans la discrimination parole/musique [Scheirer et Slanely, 1997] (sur des signaux non-bruités).

Nous calculons ce descripteur sur les fenêtres d'analyse longues ($lZCR$) et courtes (ZCR) ce qui permet de prendre en compte d'éventuelles différences de durée de stationnarité de la forme d'onde.

2) Moments statistiques temporels

Nous calculons les descripteurs suivants de façon similaire à celle utilisée pour le calcul des moments spectraux (*cf.* section IV-2-B.1) et sur deux horizons temporels :

- sur les fenêtres courtes (32ms), Tc, Tw, Ta et Tk (moments à court-terme) ;
- sur les fenêtres longues (960ms), lTc, lTw, lTa et lTk (moments à long terme) ;
- à partir des envelopes d'amplitude (*cf.* section III-1-E) et sur les fenêtres longues, eTc, eTw, eTa et eTk ;

correspondant respectivement aux quatre premiers moments spectraux. Les dérivées temporelles premières et deuxièmes de ces attributs sont aussi calculées.

3) Coefficients d'Autocorrelation (AC)

Ce descripteur a été utilisé avec succès par Brown pour la reconnaissance automatique des instruments de musique [Brown, 1998]. Il est obtenu en gardant les P_a premiers coefficients de la transformée de Fourier inverse du périodogramme du signal (approximant sa densité spectrale de puissance). De ce fait, il peut être vu comme une représentation de l'enveloppe spectrale. Nous prenons $Pa = 49$.

4) Attributs de Modulation d'Amplitude (AM)

La modulation d'amplitude est mesurée au moyen de la transformée de Fourier de l'enveloppe d'amplitude du signal temporel (que nous observons dans ce cas sur une fenêtre d'analyse longue

de 960ms). Si l'attention est portée à l'intervalle de fréquence 4-8Hz, une caractérisation du *trémolo* est obtenue, alors que des mesures effectuées dans l'intervalle 10-40Hz permettent de décrire la "*granularité*" ou "*rugosité*" des sons [Martin, 1999,Eronen, 2001a]. Nous calculons ainsi un ensemble de six coefficients (tel que décrit dans le travail de Eronen [Eronen, 2001a]). Dans chacun des deux intervalles de fréquences 4-8Hz et 10-40Hz, nous obtenons :

- la fréquence AM ; c'est la fréquence du pic d'amplitude maximale dans l'intervalle considéré ;
- l'amplitude AM ; c'est la différence entre l'amplitude maximale dans l'intervalle d'intérêt et l'amplitude moyenne sur toute la largeur de bande ;
- l'amplitude AM heuristique ; c'est la différence entre l'amplitude maximale dans l'intervalle d'intérêt et l'amplitude moyenne sur ce même intervalle.

Nous introduisons deux coefficients supplémentaires afin de prendre en compte le fait que les fréquences AM sont mesurées systématiquement, même lorsque le signal ne présente pas réellement de modulation d'amplitude, il s'agit des produits de la fréquence AM et de l'amplitude AM (dans les deux intervalles).

D. Descripteurs perceptuels

1) Loudness spécifique relative (Ld)

Nous utilisons l'approximation de la *Loudness* (intensité perceptive) [Moore et Glasberg, 1997] retenue dans [Peeters, 2004] (d'après [Rodet et Jaillet, 2001]). La *Loudness spécifique* est définie dans la bande critique bc par

$$L(bc) = E(bc)^{0.23}, \tag{IV.16}$$

où $E(bc)$ est l'énergie du signal dans la bande bc. Nous mesurons en fait la *Loudness spécifique relative* :

$$Ld(bc) = \frac{L(bc)}{L_T}, \tag{IV.17}$$

$L_T = \sum_{sb} L(sb)$ étant la Loudness totale. Cela permet de rendre Ld indépendante de la Loudness totale (qui est spécifique aux conditions d'enregistrement). De plus, nous mesurons les dérivées temporelles premières et secondes de Ld ce qui permet de rendre compte de l'évolution temporelle de la Loudness.

2) Sharpness (Sh)

La *sharpness* représente une version "perceptuelle" du centroïde spectral calculée à partir de la Loudness spécifique selon [Peeters, 2004] :

$$Sh = 0.11 \frac{\sum_{bc} bc\ g(bc) Ld(bc)}{L_T}, \tag{IV.18}$$

avec $g(bc)$ définie par

$$g(bc) = \begin{cases} 1 & \text{si } bc < 15 \\ 0.066 \exp(0.171 bc) & \text{si } bc \geq 15 \end{cases} \tag{IV.19}$$

En plus de Sh nous calculons sa vitesse et accélération pour suivre le comportement du centroïde perceptuel dans le temps.

3) Largeur perceptuelle (Sp-"Spread")

Il s'agit d'une mesure de l'écart entre la Loudness spécifique maximale et la Loudness totale [Peeters, 2004] :

$$Sp = \left(\frac{L_T - \max_{bc} Ld(bc)}{L_T} \right)^2. \tag{IV.20}$$

Nous procédons à l'extraction de Sp, et des dérivées temporelles δSp et $\delta^2 Sp$.

E. Paramètres basés sur le comportement local de la transformée en ondelettes

Nous utilisons deux ensembles d'attributs calculés à partir d'une TOD :

- un ensemble de 7 coefficients (désigné par W), proposé par [Leveau, 2004] et calculé en utilisant des ondelettes de Haar (*cf.* [Mallat, 2000]) ;
- un ensemble de 28 coefficients (désigné par $DWCH$), issu de [Li et Ogihara, 2005] et calculé à l'aide des ondelettes de Daubechies (*cf.* [Mallat, 2000]).

Descripteur W Soient $d_{s,u}$ les coefficients d'ondelette (projections du signal sur les ondelettes), et soit $B[s]$ la branche reliant les coefficients d'ondelettes correspondant aux différentes échelles s (de la plus grande échelle à la plus petite échelle). La fonction

$$\kappa(s,u) = \sum_{(s,u) \in B[s]} 2^s |d_{s,u}| \tag{IV.21}$$

permet de caractériser les singularités du signal. Les attributs W visent à décrire la répartition de l'énergie selon les branches $B[s]$.

Trois paramètres sont calculés à partir de la branche $B[s]$ correspondant au maximum de singularité (maximum de $\kappa(s,u)$), mais également à partir de la moyenne (sur s) des branches $B[s]$; il s'agit de la pente de l'asymptote (vers les petites échelles) et des deux premiers moments statistiques.

Le septième coefficient W est le moment d'ordre 4 de tous les coefficients d'ondelettes. Nous invitons le lecteur à consulter [Leveau, 2004] pour plus de détails.

Descripteur $DWCH$ Pour le calcul des coefficients $DWCH$, les trois premiers moments statistiques et l'énergie des coefficients d'ondelettes correspondant à une même échelle sont calculés dans 4 bandes de fréquence. On pourra consulter [Li et Ogihara, 2005] pour une description plus détaillée.

IV-3. Nouvelles propositions

A. Intensités des signaux de sous-bandes en octaves ($OBSI$)

L'idée de ce nouveau descripteur est de capturer de façon sommaire la distribution de puissance des différentes harmoniques du son, sans pour autant avoir recours à une étape de détection de la fréquence fondamentale. De fait, une mesure précise des fréquences et amplitudes des différents partiels n'est pas nécessaire à notre tâche. Il suffit de représenter les différences de structure spectrale des sons instrumentaux.

Cela peut être réalisé au moyen d'un banc de filtres approprié, conçu de telle sorte que l'énergie capturée dans chaque sous-bande varie pour deux instruments présentant une distribution d'énergie des partiels différente. Nous considérons donc un banc de filtres en octaves, de réponses fréquentielles triangulaires. Les bords des filtres sont alignés sur des fréquences fondamentales de notes musicales commençant à la note de piano la plus basse La1 (27.5Hz). Pour chaque sous-bande en octave, le maximum de la réponse fréquentielle est atteint au milieu de la sous-bande. Un recouvrement important (d'une demi-octave) est maintenu entre canaux adjacents. Un partiel est ainsi toujours capturé "maximalement" dans une seule sous-bande. La figure IV.3 montre le banc de filtres proposé.

Nous mesurons alors la log-énergie dans chaque sous-bande ($OBSI$- *Octave Band Signal Intensities*) mais également le logarithme du rapport d'énergie de chaque sous-bande sb à la précédente $sb-1$ ($OBSIR$- *Octave Band Signal Intensity Ratios*).

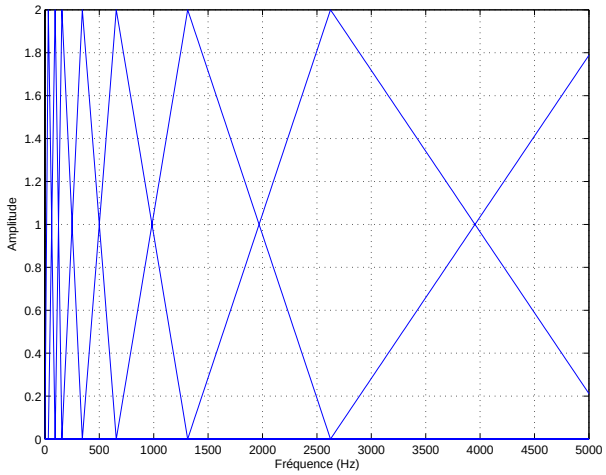

Fig. IV.3 Banc de filtres utilisé pour le calcul des $OBSI/OBSIR$.

Il en résulte que l'énergie capturée dans chaque sous-bande en octave, aussi bien que le rapport d'énergie d'une sous-bande à celle qui la précède, seront distincts pour deux instruments possédant une structure d'harmoniques différente. De plus, dans de nombreux cas, une localisation grossière de la fréquence fondamentale (f_0) est obtenue, puisque l'octave à laquelle elle appartient peut être déduite à partir du premier pic de la fonction $OBSI$. La figure IV-3-A donne une illustration de la discussion précédente en considérant les spectres de clarinette et de saxophone alto jouant la même note La5. On observe que le spectre de la clarinette présente une énergie plus importante que celle du spectre du saxophone dans la deuxième sous-bande apparaissant dans la figure, alors que le spectre du saxophone alto présente une énergie plus importante que celle du spectre de clarinette dans la troisième et la quatrième sous-bande. Il est, en effet, connu que la clarinette est caractérisée par la proéminence de ses harmoniques paires et les attributs $OBSI/OBSIR$ permettent de décrire cette propriété de façon sommaire.

B. Rapports Signal à Masque (SMR)

L'idée est de vérifier si les seuils de masquage [Painter et Spanias, 2000] calculés pour des sources sonores différentes peuvent être utilisés pour les différencier. Nous utilisons simplement une implémentation inspirée du modèle psychoacoustique proposé dans le standard de codage audio MPEG-AAC pour le calcul des Rapports Signal à Masque (SMR- Signal to Mask Ratios) [ISO/IEC, 1997]. Le principe de calcul est très sommairement décrit ci-après.

Une estimation de la densité spectrale de puissance du signal est calculée et projetée à partir

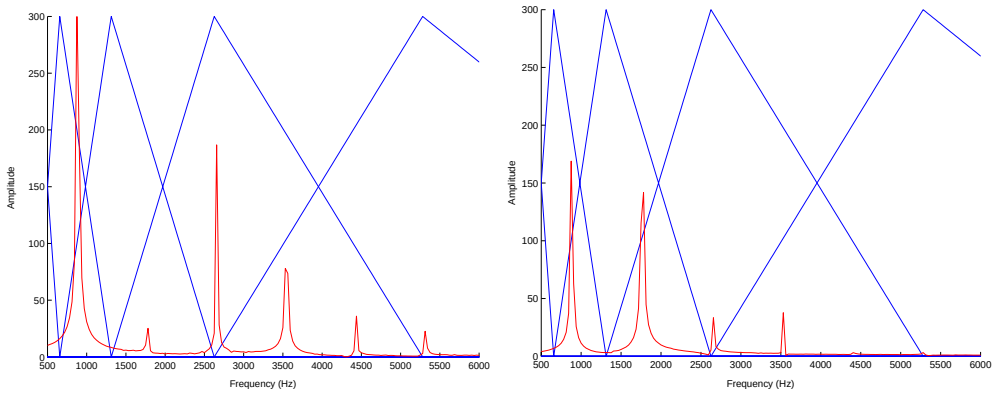

Fig. IV.4 Spectres d'amplitude relatifs au saxophone alto (à gauche) et la clarinette (à droite), jouant la même note La5, et le banc de filtres en octaves en superposition. Dans la deuxième sous-bande, une valeur importante d'$OBSI$ sera mesurée pour la clarinette ; dans les troisième et quatrième sous-bandes, une valeur plus importante d'$OBSI$ pour le saxophone sera mesurée.

du domaine de fréquence linéaire vers un domaine de fréquences dit en *partitions*, où une partition représente une résolution d'environ 1/3 de bande critique. Le nouveau spectre est alors convolué par une *fonction d'étalement* dépendant de la fréquence, pour donner un *spectre d'énergie partitionné*. Une mesure de la *tonalité* des composantes spectrales est alors obtenue et utilisée pour déterminer un facteur d'atténuation. Cette atténuation est appliquée au spectre d'énergie partitionné pour trouver le *seuil de masquage* dans une partition donnée. Enfin, les Rapports Signal à Masque sont calculés sur un nombre de sous-bandes fréquentielles (couvrant tout le domaine fréquentiel) comme le rapport entre l'énergie spectrale et le seuil de masquage (en échelle linéaire) dans chaque sous-bande. 51 coefficients de SMR sont ainsi obtenus.

IV-4. Récapitulation

Le tableau IV.1 récapitule les descripteurs utilisés dans cette étude. Nous allons nous intéresser maintenant à la façon d'utiliser ces attributs au sein d'un système de classification automatique.

Descripteur ou paquet d'attributs	Taille	Synopsis
$Cp = [Cp1, ..., Cp11]$, $(\delta,\delta^2)[Cp0, ..., Cp10]$	33	Coefficients cepstraux à partir de 30 sous-bandes MEL et dérivées temporelles.
$Cc = [Cc1, ..., Cc11]$, $(\delta,\delta^2)[Cc0, ..., Cc10]$	33	Coefficients cepstraux à partir de 11 sous-bandes MEL et dérivées temporelles.
uCq, $(\delta,\delta^2)uCq$	27	Coefficients cepstraux à partir d'une CQT avec résolution d'une octave.
dCq, $(\delta,\delta^2)dCq$	30	Coefficients cepstraux à partir d'une CQT avec résolution d'une demi-octave.
tCq, $(\delta,\delta^2)tCq$	30	Coefficients cepstraux à partir d'une CQT avec résolution d'un tiers d'octave.
qCq, $(\delta,\delta^2)qCq$	30	Coefficients cepstraux à partir d'une CQT avec résolution d'un quart d'octave.
Sx=[Sc,Sw,Sa,Sk]+δ+δ^2	12	Moments spectraux et dérivées temporelles.
ASF= [$A1$,...,$A23$]	23	Platitude spectrale (MPEG-7).
SCF=[$SCF1$,...,$SCF23$]	23	Facteur de crête spectrale.
AR=[$AR1$,$AR2$]	2	Coefficients LPC.
[Ss,Sd,Sv,So,Fc]	5	Pente, décroissance, variation temporelle, platitude du spectre, fréquence de coupure.
Si=[$Si1$,...,$Si21$]	21	Irrégularité spectrale.
$OBSI$=[$O1$,...,$O8$]	8	Intensités en sous-bandes d'octaves.
$OBSIR$=[$OR1$,...,$OR7$]	7	Rapports d'intensité en sous-bandes d'octaves.
Z=[ZCR,$lZCR$]	2	Taux de passage par 0 à partir de fenêtres courtes et longues.
Tx=[Tc,Tw,Ta,Tk]+δ+δ^2	12	Moments temporels et dérivées temporelles à partir de fenêtres courtes.
lTx=[lTc,lTw,lTa,lTk]+δ+δ^2	12	Moments temporels et dérivées temporelles à partir de fenêtres longues.
eTx=[eTc,eTw,eTa,eTk]+δ+δ^2	12	Moments temporels et dérivées temporelles à partir de l'enveloppe d'amplitude.
AC=[$AC1$,...,$AC49$]	49	Coefficients d'autocorrelation.
AM=[$AM1$,...,$AM8$]	8	Paramètres de modulation d'amplitude (tremolo, rugosité).
Ld=[$L1$,...,$L24$]+δ+δ^2	72	Loudness et dérivées temporelles.
[Sh,Sp]+δ+δ^2	6	Sharpness et largeur perceptuelle et dérivées temporelles.
SMR=[$S1$,...,$S51$]	51	Rapports Signal à Masque.
W=[$W1$,...,$W7$], $DWCH$=[$DWCH1$,...,$DWCH7$]	35	Paramètres à partir d'une Transformée en Ondelettes Discrète.

Tab. IV.1 Descripteurs utilisés dans cette étude. Au total nous obtenons 543 attributs.

DEUXIEME PARTIE

Outils utilisés pour l'apprentissage automatique

V. Fondements théoriques

Nous présentons dans ce chapitre les outils de classification que nous utilisons dans notre système de reconnaissance des instruments, en expliquant leurs fondements théoriques. Cela servira à la fois à la compréhension du fonctionnement de l'architecture proposée et à la justification des choix effectués.

V-1. Classification supervisée

L'apprentissage supervisé concerne le cas où les données d'entrée sont organisées en catégories ou *classes* connues d'avance. C'est le cas par exemple pour la tâche de reconnaissance des instruments pour laquelle nous disposons d'observations d'attributs (*exemples d'apprentissage*) clairement associées à chacune des classes d'instruments considérées.

A. Principe de décision

Étant donné un ensemble de classes $\{\Omega_q\}_{1\leq q\leq Q}$, avec Q le nombre de classes, on suppose connues :

- les *densités de probabilité conditionnelles* $p(\mathbf{x}|\Omega_q)$, décrivant la distribution des vecteurs d'attributs $\mathbf{x}$ relatifs à la classe Ω_q ; elles sont aussi appelées *vraisemblance* de Ω_q par rapport à $\mathbf{x}$;
- les *probabilités à priori* $P(\Omega_q)$ de chaque classe Ω_q.

La *règle de décision bayésienne* associe $\mathbf{x}$ à la classe Ω_{q_0} si et seulement si :

$$q_0 = \arg \max_{1\leq q\leq Q} P(\Omega_q|\mathbf{x}). \tag{V.1}$$

On parle de décision au sens du *Maximum A Posteriori* (MAP). Cette règle de décision garantit une probabilité d'erreur minimale étant donnée l'observation $\mathbf{x}$ [Duda *et al.*, 2001].

En vertu de *la formule de Bayes* :

$$P(\Omega_q|\mathbf{x}) = \frac{P(\Omega_q)p(\mathbf{x}|\Omega_q)}{p(\mathbf{x})}, \tag{V.2}$$

(V.1) se récrit :

$$q_0 = \arg \max_{1\leq q\leq Q} P(\Omega_q)p(\mathbf{x}|\Omega_q). \tag{V.3}$$

En faisant l'hypothèse que les classes Ω_q sont équiprobables, c'est-à-dire que $P(\Omega_q)$ est une constante[1] égale à $\frac{1}{Q}$, (V.3) se simplifie :

$$q_0 = \arg \max_{1\leq q\leq Q} p(\mathbf{x}|\Omega_q). \tag{V.4}$$

Enfin, dans le cas où la décision est prise sur un ensemble d'observations $\{\mathbf{x}_1, ..., \mathbf{x}_{N_t}\}$ supposées indépendantes, la règle suivante peut être utilisée :

$$q_0 = \arg \max_{1\leq q\leq Q} \prod_{m=1}^{N_t} p(\mathbf{x}_m|\Omega_q). \tag{V.5}$$

Pour éviter que le produit apparaissant dans (V.5) ne devienne trop petit pour une suite de valeurs petites de $p(\mathbf{x}_m|\Omega_q)$, on préfère généralement utiliser :

$$q_0 = \arg \max_{1\leq q\leq Q} \sum_{m=1}^{N_t} \log p(\mathbf{x}_m|\Omega_q). \tag{V.6}$$

Dans notre cas, ces observations vont correspondre à une suite de N_t fenêtres d'analyse temporelles successives et la question de la validité de l'hypothèse d'indépendance des $\mathbf{x}_m$ se pose. En fait, ces observations sont clairement dépendantes, eu égard à la stationnarité locale du signal audio. Cela n'empêche pas cette hypothèse d'être largement utilisée en classification audio : elle permet en pratique de résoudre efficacement le problème de décision [R. Rabiner, 1993].

[1]Cette hypothèse est largement utilisée pour des cas d'étude. En pratique, il peut être intéressant de tenir compte du contexte d'application et d'exploiter des aprioris différents sur les probabilités des classes : on peut par exemple décider que la présence d'un violon dans les enregistrements de musique est beaucoup plus probable que la présence d'un tuba...

B. Schémas de classification binaire

1) Principe

Nous expérimenterons des schémas de classification décomposant le problème de classification à Q classes en problèmes bi-classes[2] "un contre un". Il s'agit de considérer toutes les combinaisons de deux classes $\{\Omega_p, \Omega_q\}_{1 \leq p < q \leq Q}$ parmi Q possibles (elles sont au nombre de $\frac{Q(Q-1)}{2}$) en construisant les classificateurs $\mathcal{C}_{p,q}$ servant à discriminer Ω_p et Ω_q. La classification de nouveaux exemples est alors réalisée en les testant avec tous les classificateurs $\{\mathcal{C}_{p,q}\}_{1 \leq p < q \leq Q}$ et la décision finale est obtenue en fusionnant les décisions prises dans tous les sous-problèmes bi-classes. Une stratégie de fusion sera donc nécessaire, nous la décrivons dans la section V-1-B.2. La figure V.1 donne un exemple de réalisation du schéma binaire.

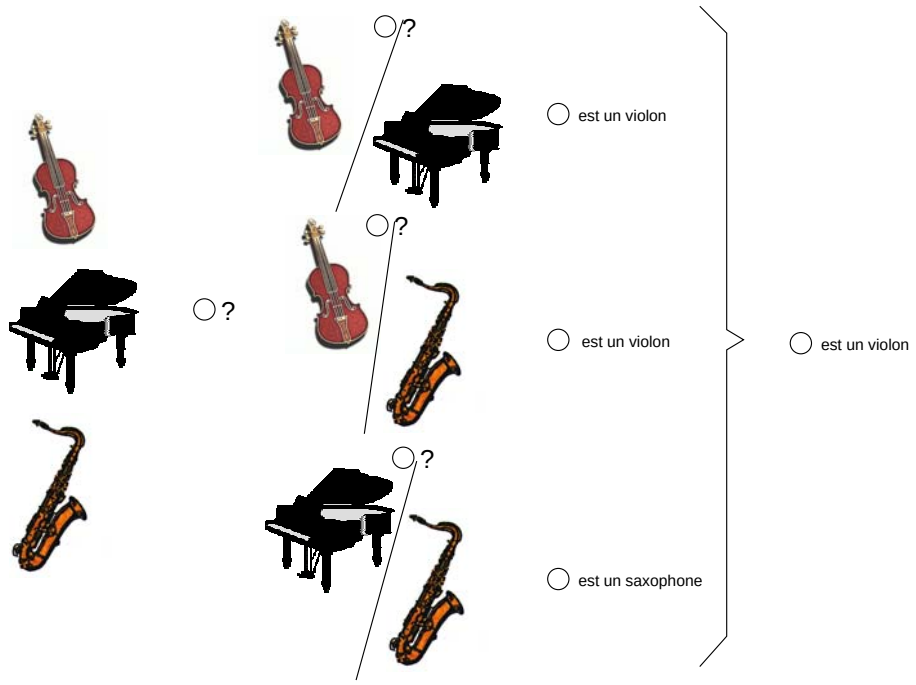

Fig. V.1 Décomposition du problème de classification à 3 classes en 3 sous-problèmes bi-classes.

Il est important de noter que ce schéma sera envisagé pour des classificateurs qui sont par essence binaires (par exemple les SVM, présentés à la section V-2) et d'autres qui sont de nature

[2]Nous verrons par la suite que cette approche peut s'avérer avantageuse (*cf.* chapitres VI et IX).

multi-classes (par exemple les κ-NN, présentés à la section V-1-D, et les GMM présentés à la section V-1-C).

2) Fusion des décisions binaires

La décomposition d'un problème de classification à Q classes en sous-problèmes binaires, nécessite l'emploi d'une stratégie permettant de fusionner les décisions prises par les différents classificateurs bi-classes, pour parvenir à une décision finale. Dans l'exemple, de la figure V.1, une procédure de *vote majoritaire* a été utilisée : l'exemple inconnu a été classé "violon" car 2 classificateurs binaires sur trois ont pris cette décision ; le violon a reçu 2 votes, il l'emporte sur les autres classes. Cette procédure présente deux inconvénients majeurs :

- d'abord, des indéterminations peuvent apparaître : si le premier classificateur (violon vs piano) assigne l'exemple de test à la classe piano, il n'est pas possible de prendre une décision car les trois instruments reçoivent le même nombre de votes ;
- ensuite, on n'obtient pas en sortie du schéma de classification des probabilités d'appartenance aux classes, $P(\Omega_q|\mathbf{x})$ pour un exemple $\mathbf{x}$, ce qui peut être fortement gênant dans de nombreuses situations, par exemple s'il est envisagé de combiner la sortie de ce schéma de classification avec celles d'autres classificateurs.

Hastie & Tibshirani proposent une solution efficace au problème de fusion des décisions binaires [Hastie et Tibshirani, 1998] qui permet d'obtenir des estimations des probabilités $P(\Omega_q|\mathbf{x})$.

Soit $r_{qm}(\mathbf{x})$ la probabilité que la classe correspondant à l'observation $\mathbf{x}$ soit Ω_q dans le problème bi-classes (Ω_q vs Ω_m). $r_{qm}(\mathbf{x})$ s'écrit :

$$r_{qm}(\mathbf{x}) = P(\Omega = \Omega_q | \Omega = \Omega_q \text{ ou } \Omega = \Omega_m, \mathbf{x}) = \frac{p_q(\mathbf{x})}{p_q(\mathbf{x}) + p_m(\mathbf{x})}, \tag{V.7}$$

avec la notation $p_q(\mathbf{x}) = P(\Omega = \Omega_q|\mathbf{x})$.

Le but est de déterminer les probabilités d'appartenance aux Q classes $p_q(\mathbf{x})$, $1 \leq q \leq Q$. Notons qu'elles doivent vérifier $\sum_q p_q(\mathbf{x}) = 1$.

Soient n_{qm} le nombre d'exemples d'apprentissage utilisés pour calculer le classificateur binaire qui prédit r_{qm}. Les $p_q(\mathbf{x})$ sont déterminés par un algorithme du gradient en recherchant les approximations $\hat{r}_{qm}(\mathbf{x}) = \frac{\hat{p}_q(\mathbf{x})}{\hat{p}_q(\mathbf{x}) + \hat{p}_m(\mathbf{x})}$ de $r_{qm}(\mathbf{x})$ qui minimisent la distance de Kullback-Leibler moyenne entre $r_{qm}(\mathbf{x})$ et $\hat{r}_{qm}(\mathbf{x})$. Cette distance, notée $\mathcal{L}(\mathbf{x})$, s'écrit :

$$\mathcal{L}(\mathbf{x}) = \sum_{q<m} n_{qm} \left[r_{qm}(\mathbf{x}) \log \left(\frac{r_{qm}(\mathbf{x})}{\hat{r}_{qm}(\mathbf{x})} \right) + (1 - r_{qm}(\mathbf{x})) \log \left(\frac{1 - r_{qm}(\mathbf{x})}{1 - \hat{r}_{qm}(\mathbf{x})} \right) \right]. \tag{V.8}$$

L'algorithme correspondant est donné ci-après (Algorithme 1).

Algorithme 1 Hastie & Tibshirani.

Entrées: $r_{qm}(\mathbf{x})$, n_{qm}

//Initialisation

Choisir des valeurs initiales pour $\hat{p}_q(\mathbf{x})$ et $\hat{r}_{qm}(\mathbf{x})$.

répéter

Pour chaque $q = 1, ..., Q$

1) $\hat{p}_q(\mathbf{x}) \leftarrow \hat{p}_q(\mathbf{x}) \frac{\sum_{q<m} n_{qm} r_{qm}(\mathbf{x})}{\sum_{q<m} n_{qm} \hat{r}_{qm}(\mathbf{x})}$

2) Renormaliser les $\hat{p}_q(\mathbf{x})$

3) Recalculer les $\hat{r}_{qm}(\mathbf{x})$.

jusqu'à Atteindre la convergence

Sorties: Retourner les $\hat{p}_q(\mathbf{x})$.

Hastie & Tibshirani montrent que la distance de Kullback-Leibler entre $r_{qm}(\mathbf{x})$ et $\hat{r}_{qm}(\mathbf{x})$ décroît à chaque itération. Comme cette distance admet zéro comme borne inférieure, l'algorithme converge.

La décision peut alors être prise en choisissant pour $\mathbf{x}$ la classe Ω_{q_0} telle que $q_0 = \arg\max_q \hat{p}_q(\mathbf{x})$.

C. Le Modèle de Mélange Gaussien (GMM)

Le modèle de mélange Gaussien (GMM- Gaussian Mixture Model) a été largement utilisé dans la communauté de la reconnaissance de la parole et du locuteur depuis son introduction par Reynolds [Reynolds et Rose, 1995]. Il a également été utilisé avec succès pour la reconnaissance des instruments de musique [Brown *et al.*, 2000, Eronen, 2001a]. Nous donnons ici une vue d'ensemble succincte de ce modèle qui est bien connu dans la littérature.

Il s'agit d'une approche paramétrique qui suppose une forme particulière des densités de probabilités conditionnelles $p(\mathbf{x}|\Omega_q)$:

$$p(\mathbf{x}|\Omega_q) = \sum_{m=1}^{M} w_{m,q} b_{m,q}(\mathbf{x}), \tag{V.9}$$

où les $w_{m,q}$ sont des poids scalaires positifs, vérifiant $\sum_{m=1}^{M} w_{m,q} = 1$ et les $b_{m,q}(\mathbf{x})$ sont des densités de probabilité gaussiennes, appelées *composantes du mélange*. $p(\mathbf{x}|\Omega_q)$ s'exprime ainsi

comme une combinaison linéaire de M composantes de densité gaussienne $b_{m,q}(\mathbf{x})$ qui s'écrivent, en fonction de leurs moyennes $\boldsymbol{\mu}_{m,q}$ et de leurs matrices de covariance $\boldsymbol{\Sigma}_{m,q}$:

$$b_{m,q}(\mathbf{x}) = \frac{1}{(2\pi)^{d/2}|\boldsymbol{\Sigma}_{m,q}|^{\frac{1}{2}}} \exp\left[-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_{m,q})^T(\boldsymbol{\Sigma}_{m,q})^{-1}(\mathbf{x}-\boldsymbol{\mu}_{m,q})\right]. \tag{V.10}$$

Les différentes composantes du mélange sont supposées modéliser des régions différentes de l'espace des attributs (*clusters* différents) associées à des classes acoustiques distinctes.

Les paramètres du modèle associé à la classe Ω_q, notés $\lambda_q = \{w_{m,q}\,, \boldsymbol{\mu}_{m,q}\,, \boldsymbol{\Sigma}_{m,q}\}_{m=1,\ldots,M}$, sont estimés en utilisant le fameux algorithme EM (*Expectation Maximisation*) [Dempster *et al.*, 1977, Moon, 1996].

Nous initialisons l'algorithme comme suit :

- les poids $w_{m,q}$ sont initialisés à $\frac{1}{M}$;
- les centroïdes des régions de Voronoï obtenues par un algorithme LBG[3] servent d'initialisation des moyennes $\boldsymbol{\mu}_{m,q}$;
- les matrices de covariances supposées diagonales sont initialisées en utilisant des estimations empiriques de la variance des données dans chaque région de Voronoï.

La règle de classification utilisée est classiquement la règle MAP donnée par (V.6).

Nous serons amenés à utiliser les GMM dans un schéma de classification binaire "1 contre 1" (*cf.* section V-1-B). Pour une observation donnée $\mathbf{x}$, et un contexte bi-classes $\{\Omega_p, \Omega_q\}$ la règle de décision est alors :

$$\Omega_{q_0} = \begin{cases} \Omega_p, & \text{si } \; p(\mathbf{x}|\Omega_p) > p(\mathbf{x}|\Omega_q) \\ \Omega_q, & \text{sinon.} \end{cases} \tag{V.11}$$

Pour aboutir à une décision globale il sera nécessaire de recourir à une technique permettant de fusionner les sorties des différents classificateurs binaires. Nous utilisons pour cela l'approche de Hastie & Tibshirani (*cf.* section V-1-B.2) comme suit. Pour un exemple de test donné $\mathbf{x}$ nous

[3]LBG : algorithme de Linde, Buzo, Gray, connu pour être plus robuste aux effets de l'initialisation. Nous invitions le lecteur à consulter [Linde *et al.*, 1980] pour les détails de cet algorithme.

obtenons $p(\mathbf{x}|\Omega_q)$ et $p(\mathbf{x}|\Omega_q)$ pour chaque paire de classe $\{\Omega_p, \Omega_q\}$ (en utilisant (V.9)) et nous calculons

$$\hat{r}_{pq} = \frac{p(\mathbf{x}|\Omega_p)}{p(\mathbf{x}|\Omega_p) + p(\mathbf{x}|\Omega_q)}, \tag{V.12}$$

et

$$\hat{r}_{qp} = \frac{p(\mathbf{x}|\Omega_q)}{p(\mathbf{x}|\Omega_p) + p(\mathbf{x}|\Omega_q)} = 1 - \hat{r}_{pq}. \tag{V.13}$$

La méthode décrite dans la section V-1-B.2 est alors utilisée pour estimer les probabilités d'appartenance aux classes, $p_q(\mathbf{x})$, en exploitant le modèle (V.7) pour $\hat{r}_{pq}$.

D. Les κ plus proches voisins (κ-NN)

Cette approche fait partie des approches non-paramétriques. Aucune hypothèse n'est faite ici sur les lois régissant les densités de probabilité mises en jeu, ce qui constitue un point fort de ces méthodes du point de vue de leur fondement théorique.

Soit $\mathcal{E} = \{\mathbf{x}_1, .., \mathbf{x}_l\}$ un ensemble d'exemples d'apprentissage. L'algorithme de classification par les κ *plus proches voisins* (κ-NN : κ Nearest Neighbours) affecte à un exemple de test $\mathbf{x}$, la classe la plus fréquemment représentée parmi celles correspondant aux κ points de $\mathcal{E}$ les plus proches de $\mathbf{x}$.

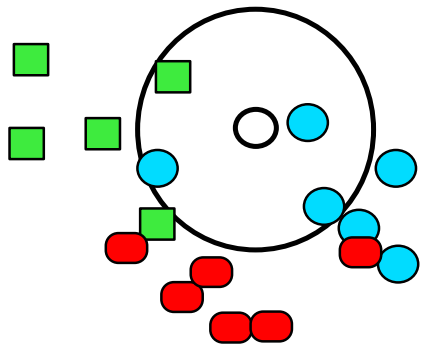

Fig. V.2 Illustration du fonctionnement des κ-NN, avec κ=4. La classe sélectionnée pour l'exemple de test "rond vide" est celle des ronds pleins (bleus).

Les performances de cette méthode sont liées au choix de deux paramètres sensibles : le choix d'une métrique appropriée (pour le calcul des proximités entre exemples) et le choix du paramètre κ.

Le premier choix est souvent arrêté sur la distance euclidienne. Cette dernière n'étant pas toujours adaptée aux données, notamment les données audio [R. Rabiner, 1993], il peut être avantageux de considérer des métriques alternatives (*cf.* section V-3-B).

L'influence du choix de κ peut être cernée en faisant un parallèle avec la règle de décision bayésienne. Nous proposons l'explication suivante d'après [Duda *et al.*, 2001]. Pour un exemple de test $\mathbf{x}_m$, les κ-NN sélectionnent la classe Ω_{q_0}, si la majorité des κ plus proches voisins $\mathbf{x}_v$ sont étiquetés Ω_{q_0}. La règle de décision bayésienne quant à elle sélectionne la classe Ω_{q_0} réalisant le maximum de $P(\Omega_q|\mathbf{x}_m)$ sur les classes possibles Ω_q, $1 \leq q \leq Q$. S'ils sont proches de $\mathbf{x}_m$, les κ plus proches voisins, considérés comme des réalisations de variables aléatoires associées aux classes Ω_q, peuvent servir à obtenir une estimation de la probabilité à posteriori $P(\Omega_{q_0}|\mathbf{x}_m)$. Cette estimation est d'autant plus fiable que κ est grand. Mais dans le même temps les κ plus proches voisins $\mathbf{x}_v$ doivent rester très proches de $\mathbf{x}_m$ pour que $P(\Omega_{q_0}|\mathbf{x}_v)$ soit une bonne approximation de $P(\Omega_{q_0}|\mathbf{x}_m)$. Par suite il est nécessaire de réaliser un compromis, en choisissant une valeur de κ plus petite que le nombre d'exemples l. Nous adoptons un choix de κ connu pour être raisonnable, en prenant :

$$\kappa \approx \sqrt{l}. \tag{V.14}$$

V-2. Les Machines à Vecteurs Supports (SVM)

A. Principe de Minimisation du Risque Structurel (SRM)

Soit $\mathcal{D}$ un ensemble d'exemples d'apprentissage

$$\mathcal{D} = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_l, y_l)\};\ \mathbf{x} \in \mathbb{R}^d,\ y_i \in \{-1, +1\},$$

les $\mathbf{x}_i$ sont ici des vecteurs d'attributs pouvant appartenir à deux classes possibles : une classe positive, notée +1, et une classe négative notée -1. Les y_i représentent donc les étiquettes ou les valeurs cibles associées à $\mathbf{x}_i$. Ces exemples sont supposés tirés à partir d'une distribution de probabilité inconnue $p(\mathbf{x}, y)$. La tâche d'*apprentissage à partir des exemples* consiste à trouver parmi un ensemble $\mathcal{F}$ de fonctions de classification, permettant de prédire l'étiquette y_i de $\mathbf{x}_i$:

$$\mathcal{F} = \{f_\alpha; \alpha \in \Lambda\};\ f_\alpha : \mathbb{R}^d \mapsto \{-1, +1\},\ \text{avec } \Lambda \text{ un ensemble d'indices,}$$

la fonction f_{α^*} qui minimise le *risque fonctionnel*

$$R(\alpha) = \int \frac{1}{2} |f_\alpha(\mathbf{x}) - y| \, dp(\mathbf{x}, y), \tag{V.15}$$

autrement dit, la fonction qui minimise "la probabilité de mal prédire l'étiquette de $\mathbf{x}$". La difficulté rencontrée vient du fait que $R(\alpha)$ est inconnue puisque $p(\mathbf{x}, y)$ est inconnue. Il est donc nécessaire de faire appel à un *principe d'induction*, en se basant sur les données d'apprentissage.

L'approche la plus directe consiste à adopter une stratégie visant à minimiser l'erreur de classification sur l'ensemble d'apprentissage ou le *risque empirique* R_{emp} défini par :

$$R_{emp}(\alpha) = \frac{1}{l}\sum_{i=1}^{l}\frac{1}{2}|f_\alpha(\mathbf{x}_i) - y_i|. \tag{V.16}$$

C'est ce qui fait l'objet de l'approche de *Minimisation du Risque Empirique (ERM)* qui s'appuie sur le fait que $R_{emp}(\alpha)$ tend vers $R(\alpha)$ lorsque l tend vers l'infini (en vertu de la loi des grands nombres).

Lorsque le nombre d'exemples d'apprentissage l est petit, il s'avère que minimiser le risque $R_{emp}(\alpha)$ n'implique pas forcément un risque $R(\alpha)$ minimal. En minimisant le risque empirique il est possible d'obtenir un modèle efficace sur les exemples de l'ensemble d'apprentissage mais ce dernier ne garantit pas des performances satisfaisantes *en généralisation*, c'est-à-dire sur de nouveaux exemples. Ce phénomène est connu sous le terme de *sur-apprentissage* ou *overfitting*.

Le principe de *Minimisation du Risque Structurel*, dû à Vapnik & Chervonenkis, permet de pallier cette difficulté [Vapnik, 1995]. Il repose sur le concept de *dimension VC* (Vapnik Chervonenkis) d'un ensemble de fonctions, notée h, qui permet d'obtenir la borne suivante sur le risque. On obtient avec une probabilité $1 - \eta$:

$$R(\alpha) \leq R_{emp}(\alpha) + \sqrt{\frac{h\left(\log\frac{2l}{h} + 1\right) - \log\left(\frac{\eta}{4}\right)}{l}}. \tag{V.17}$$

La dimension VC décrit la *capacité* (de séparation) d'un ensemble de fonctions considérées par un algorithme d'apprentissage. Pour un problème bi-classes, h est le nombre maximum de points k qui peuvent être séparés, par le biais de ces fonctions, en deux classes, et ce de toutes les façons possibles (2^k façons). Ce concept est illustré à la figure V.3.

L'inégalité (V.17) indique que l'erreur en généralisation, *i.e.* $R(\alpha)$, peut être maîtrisée en contrôlant d'une part, le risque empirique, d'autre part, une quantité qui dépend du rapport $\frac{l}{h}$, appelée *intervalle de confiance* (c'est la différence entre le risque fonctionnel et le risque empirique). Si ce rapport est suffisamment grand, le *risque garanti* (c'est ainsi que l'on désigne le membre droit de l'inégalité (V.17)) est dominé par le risque empirique, et il est suffisant de minimiser R_{emp} pour garantir un risque fonctionnel minimum. Sinon, l'approche ERM n'est pas satisfaisante.

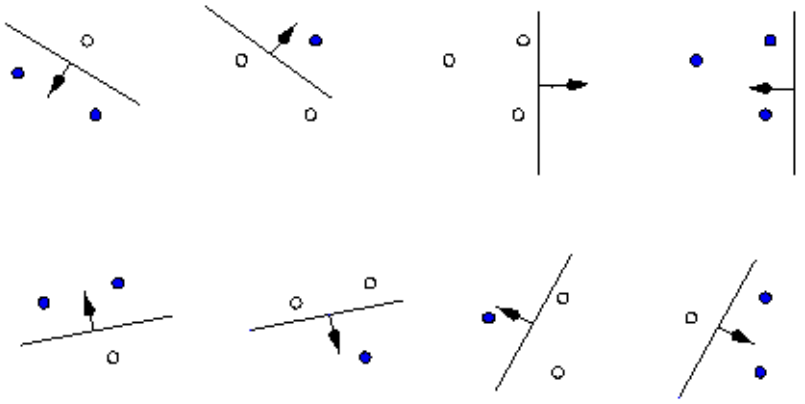

Fig. V.3 Illustration du concept de dimension VC, d'après [Burges, 1998]. Dans $\mathbb{R}^2$, en considérant un ensemble de fonctions $\{f_\alpha\}$ représentant des droites orientées, de telle manière que tous les points d'un côté de la droite soient étiquetés par +1 et tous ceux de l'autre côté de la droite étiquetés par -1, il n'est pas possible de trouver plus de trois points séparables de toutes les façons possibles. Par suite la dimension VC de l'ensemble des droites orientées dans $\mathbb{R}^2$ est trois.

L'approche SRM adopte la stratégie qui consiste à minimiser le risque en contrôlant la dimension VC. Cela est réalisé en exploitant une structuration de $\mathcal{F}$ en sous-ensembles emboîtés $\mathcal{F}_m = \{f_\alpha^m; \alpha \in \Lambda_m, \Lambda_m \subset \Lambda\}$ tels que

$$\mathcal{F}_1 \subset \mathcal{F}_2 \subset ... \subset \mathcal{F}_m \subset ... \tag{V.18}$$

Les dimensions VC correspondantes vérifient alors

$$h_1 \leq h_2 \leq ... \leq h_m \leq ... \tag{V.19}$$

Il s'agit maintenant de choisir la fonction f_α^m dans l'ensemble $\mathcal{F}_m$ qui réalise la plus petite valeur du risque garanti. Cependant, il ne suffit pas de retenir le sous-ensemble associé à la plus petite des valeurs h_m puisqu'en pratique les plus petites dimensions VC correspondent à des valeurs élevées du risque empirique et *vice-versa*, d'où la nécessité de trouver une valeur de compromis. Ainsi, il est possible de produire des algorithmes de classification dont l'efficacité statistique peut être contrôlée en se donnant une classe de fonctions dont la capacité peut être mesurée.

B. Principe des Machines à Vecteurs Supports (SVM) linéaires

Les Machines à Vecteurs Supports sont de puissants classificateurs inspirés par le principe SRM qui ont prouvé leur efficacité pour diverses tâches de classification, parmi lesquelles : l'identification/vérification du locuteur, la catégorisation de textes, la reconnaissance des visages, ... et récemment la reconnaissance des instruments de musique [Marques et Moreno, 1999]. Elles présentent l'avantage d'être *discriminatives* par opposition aux approches *génératives* (telles que les approches GMM, *cf.* section V-1-C) qui présupposent une structure particulière (souvent mal justifiée) des formes de densité des données, et exhibent en pratique une très bonne capacité de généralisation.

Les SVM sont par essence des classificateurs bi-classes qui visent à séparer les exemples de chaque classe Ω_q, $1 \leq q \leq 2$, au moyen d'un hyperplan $\mathcal{H}_{\mathbf{w}_0,b_0}$ choisi de manière à garder un maximum de marge de séparation entre n'importe quels exemples d'apprentissage et $\mathcal{H}_{\mathbf{w}_0,b_0}$.

De façon plus formelle, en se donnant les exemples $\{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_l, y_l)\}$ dans $\mathbb{R}^d \times \{-1, +1\}$ et en munissant $\mathbb{R}^d$ d'un produit scalaire (noté .), il s'agit de déterminer l'*hyperplan optimal*

$$\mathcal{H}_{\mathbf{w}_0,b_0} : \mathbf{w}_0.\mathbf{x} + b_0 = 0; \mathbf{w}_0 \in \mathbb{R}^d, b_0 \in \mathbb{R}, \tag{V.20}$$

solution de :

$$\max_{\mathbf{w}\in\mathbb{R}^d, b\in\mathbb{R}} \min_{\mathbf{x},i} \{||\mathbf{x} - \mathbf{x}_i||; \mathbf{x} \in \mathbb{R}^d, \mathbf{w}.\mathbf{x} + b = 0, i = 1, ..., l\}. \tag{V.21}$$

En utilisant une mise à l'échelle appropriée de $\mathbf{w}$ et b et en supposant dans un premier temps que les données sont linéairement séparables, il est possible de contraindre les exemples de chaque classe à satisfaire les conditions :

$$\mathbf{w}.\mathbf{x}_i + b \geq +1, \text{ pour } y_i = +1, \tag{V.22}$$

$$\mathbf{w}.\mathbf{x}_i + b \leq -1, \text{ pour } y_i = -1, \tag{V.23}$$

qui peuvent être combinées en une même inégalité :

$$y_i(\mathbf{w}.\mathbf{x}_i + b) - 1 \geq 0, \forall i. \tag{V.24}$$

Les deux hyperplans :

$$\mathcal{H}_1 : \mathbf{w}.\mathbf{x}_i + b = +1, \tag{V.25}$$

$$\mathcal{H}_2 : \mathbf{w}.\mathbf{x}_i + b = -1 \tag{V.26}$$

permettent de définir la marge. Remarquons que $\mathcal{H}_1$ et $\mathcal{H}_2$ sont parallèles (ils ont la même normale $\mathbf{w}$) et qu'il n'existe aucun point entre les deux, grâce à (V.22) et (V.23). Par suite la marge n'est autre que la distance entre $\mathcal{H}_1$ et $\mathcal{H}_2$ qui vaut $\frac{2}{||\mathbf{w}||}$. La figure V.4 en donne une illustration.

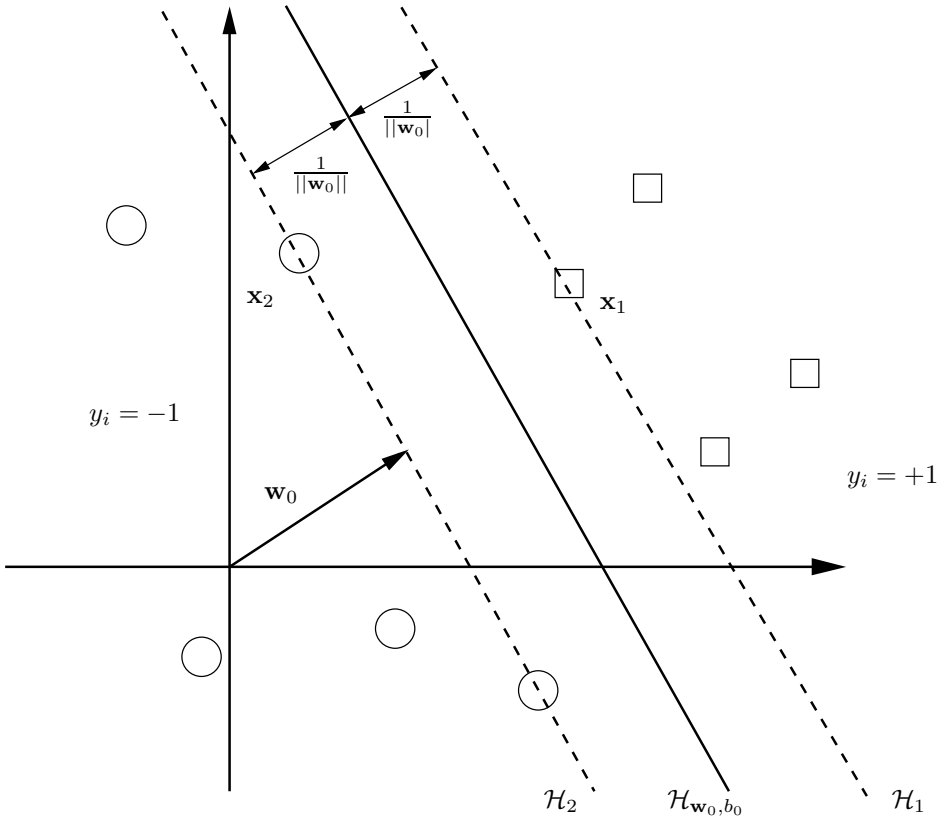

Fig. V.4 Hyperplan optimal et marge d'un classificateur SVM. Les "ronds" représentent des exemples de la classe -1 et les carrés, des exemples de la classe +1. $\mathbf{w}_0.\mathbf{x}_1 + b_0 = 1$, $\mathbf{w}_0.\mathbf{x}_2 + b_0 = -1 \Rightarrow \mathbf{w}_0.(\mathbf{x}_1 - \mathbf{x}_2) = 2 \Rightarrow \frac{\mathbf{w}_0}{||\mathbf{w}_0||}.(\mathbf{x}_1 - \mathbf{x}_2) = \frac{2}{||\mathbf{w}_0||}$.

Les points qui se trouvent sur les hyperplans $\mathcal{H}_1$ et $\mathcal{H}_2$ sont appelés les vecteurs supports (SV-*Support Vectors*). Le problème posé ne dépend en fait que de ces points particuliers en ce sens que si tous les autres points sont éliminés, la solution du problème reste la même.

Ainsi, l'hyperplan optimal est solution du problème d'optimisation

$$\begin{cases} \text{minimiser} & \tau(\mathbf{w}) = \frac{1}{2}||\mathbf{w}||^2 \ ; \ \mathbf{w} \in \mathbb{R}^d, b \in \mathbb{R} \\ \text{sous les contraintes} & y_i(\mathbf{w}.\mathbf{x}_i + b) - 1 \geq 0 \ , \ \forall i = 1, ..., l. \end{cases} \tag{V.27}$$

Notons que l'on choisit de minimiser $\frac{1}{2}||\mathbf{w}||^2$ plutôt que $\frac{1}{2}||\mathbf{w}||$ car cela facilite la résolution du problème. Nous reviendrons plus tard sur la façon de résoudre (V.27). Intéressons nous pour

l'instant à la situation plus réaliste de données non séparables par un hyperplan.

Un premier remède consiste à rendre moins rigides les contraintes (V.24) en introduisant des variables d'écart positives ξ_i pour que les contraintes deviennent

$$\mathbf{w}.\mathbf{x}_i + b \geq 1 - \xi_i, \text{ si } y_i = 1, \tag{V.28}$$

$$\mathbf{w}.\mathbf{x}_i + b \leq -1 - \xi_i, \text{ si } y_i = -1, \tag{V.29}$$

$$\xi_i \geq 0, \forall i. \tag{V.30}$$

Pour qu'un exemple d'apprentissage $\mathbf{x}_i$ soit mal classifié, il faut que le ξ_i correspondant soit supérieur à 1. Par suite $\sum_i \xi_i$ est une borne supérieure sur le nombre d'erreurs de classification qui peuvent être pénalisées en modifiant la fonction objectif $r(\mathbf{w})$ par :

$$r(\mathbf{w}, \boldsymbol{\xi}) = \frac{||\mathbf{w}||^2}{2} + C(\sum_i \xi_i), \tag{V.31}$$

où $\boldsymbol{\xi} = [\xi_1, ..., \xi_l]^T$ et $C > 0$ est un paramètre permettant de contrôler le compromis entre le fait de maximiser la marge et minimiser les erreurs de classification commises sur l'ensemble d'apprentissage. On parle alors de classificateur à *marge souple* [Shölkopf et Smola, 2002]. Notons qu'il est souvent préférable de tolérer certaines erreurs, au bénéfice d'une marge plus grande car ces erreurs peuvent être dûes à des *outliers*, observations aberrantes, non-significatives de la classe qui leur est associée. Nous reviendrons sur l'influence du paramètre C dans la partie expérimentale au chapitre VII.

Il existe une autre réponse au problème de données non linéairement séparables qui mène à l'obtention de surfaces de décisions non-linéaires. Nous exposerons cela dans la section V-2-D.

C. Calcul des SVM

Le problème (V.27) est un problème d'*optimisation sous contraintes* qui est résolu en introduisant des *multiplicateurs de Lagrange* $(\alpha_i)_{1 \leq i \leq l}$ et un *Lagrangien* :

$$L(\mathbf{w}, b, \boldsymbol{\alpha}) = \frac{1}{2}||\mathbf{w}||^2 - \sum_{i=1}^{l} \alpha_i [y_i(\mathbf{w}.\mathbf{x}_i + b) - 1], \tag{V.32}$$

où $\boldsymbol{\alpha} = [\alpha_1 ... \alpha_l]^T$.

Le Lagrangien L doit être minimisé par rapport aux variables dites *primales* $\mathbf{w}$ et b, et maximisé par rapport aux *variables duales* α_i : ce sont les conditions de Karush-Kuhn-Tucker (KKT) [Shölkopf et Smola, 2002].

Dans le cas où la *fonction objectif*, ici $\tau(\mathbf{w})$, et les contraintes, ici $c_i(\mathbf{x_i}) = y_i(\mathbf{w}.\mathbf{x}_i + b) - 1$, sont convexes, les conditions KKT sont nécessaires et suffisantes, et la solution du problème est telle que :

$$\frac{\partial L(\mathbf{w}, b, \boldsymbol{\alpha})}{\partial b} = 0 \tag{V.33}$$

$$\frac{\partial L(\mathbf{w}, b, \boldsymbol{\alpha})}{\partial \mathbf{w}} = 0 \tag{V.34}$$

$$y_i(\mathbf{w}.\mathbf{x}_i + b) - 1 \geq 0 \tag{V.35}$$

$$\alpha_i[y_i(\mathbf{w}.\mathbf{x}_i + b) - 1] = 0 \tag{V.36}$$

$$\alpha_i \geq 0. \tag{V.37}$$

Les conditions (V.33) et (V.34) donnent respectivement :

$$\sum_{i=1}^{l} \alpha_i y_i = 0 \tag{V.38}$$

$$\sum_{i=1}^{l} \alpha_i y_i \mathbf{x}_i = \mathbf{w}. \tag{V.39}$$

De plus, (V.36) implique que tous les points $\mathbf{x}_i$ qui ne sont pas vecteurs supports, *i.e.* ceux qui ne vérifient pas l'égalité $y_i(\mathbf{w}.\mathbf{x}_i + b) - 1 = 0$, sont associés à des α_i nuls. Ainsi, on retrouve que l'hyperplan optimal ne dépend que des n_s vecteurs supports du problème ($n_s \leq l$) :

$$\mathbf{w} = \sum_{i=1}^{n_s} \alpha_i y_i \mathbf{x}_i, \tag{V.40}$$

et la fonction de décision est définie par le signe de :

$$f(\mathbf{x}) = \mathbf{w}.\mathbf{x} + b = \sum_{i=1}^{n_s} \alpha_i y_i (\mathbf{x}.\mathbf{x}_i) + b. \tag{V.41}$$

Le paramètre b peut être déterminé au travers de la condition (V.36) en choisissant un indice i tel que $\alpha_i \neq 0$, ou encore en moyennant les valeurs obtenues en utilisant tous les points $\mathbf{x}_i$ associés à des α_i non nuls (pour une meilleure robustesse numérique).

En utilisant (V.32) et (V.40), on obtient la *formulation duale* du problème :

maximiser

$$L_D(\boldsymbol{\alpha}) = \sum_{i=1}^{l} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{l} \alpha_i \alpha_j y_i y_j \mathbf{x}_i.\mathbf{x}_j \tag{V.42}$$

sous les contraintes

$$\sum_{i=1}^{l} \alpha_i y_i = 0,$$
$$\alpha_i \geq 0.$$

Remarquons que $\mathbf{w}$ et b ont été éliminés et qu'il s'agit désormais de déterminer les α_i.

Revenons maintenant au cas non séparable. Le Lagrangien primal est

$$L(\mathbf{w}, b, \boldsymbol{\alpha}, \boldsymbol{\xi}, \boldsymbol{\beta}) = \frac{1}{2}||\mathbf{w}||^2 + C\sum_{i=1}^{l} \xi_i - \sum_{i=1}^{l} \alpha_i[y_i(\mathbf{w}.\mathbf{x}_i + b) - 1 + \xi_i] - \sum_i \beta_i \xi_i \tag{V.43}$$

où les β_i sont des multiplicateurs de Lagrange permettant de prendre en compte la condition $\xi_i \geq 0$ et $\boldsymbol{\beta} = [\beta_1, ..., \beta_l]^T$. Le problème dual est plus simple, il prend la forme :
maximiser

$$L_D(\boldsymbol{\alpha}) = \sum_{i=1}^{l} \alpha_i - \frac{1}{2}\sum_{i,j=1}^{l} \alpha_i \alpha_j y_i y_j \mathbf{x}_i.\mathbf{x}_j \tag{V.44}$$

sous les contraintes

$$\sum_{i=1}^{l} \alpha_i y_i = 0,$$
$$0 \leq \alpha_i \leq C. \tag{V.45}$$

Par rapport au cas séparable, une contrainte supplémentaire sur les α_i a été introduite : ils admettent à présent la borne supérieure C. En récrivant les conditions KKT on retrouve la même solution

$$\mathbf{w} = \sum_{i=1}^{n_s} \alpha_i y_i \mathbf{x}_i,$$

qui ne dépend que des vecteurs supports (SV) à la différence que dans cette réalisation "souple" des SVM, $\mathbf{w}$ dépend, en plus des SV se trouvant à la marge (sur les hyperplans $\mathcal{H}_1$ et $\mathcal{H}_2$), de vecteurs supports se retrouvant à l'intérieur de la marge (appelés *erreurs de marge*) qui sont associés à des multiplicateurs $\alpha_i = C$; ils sont désignés par BSV (*Bounded Support Vectors*).

Les conditions KKT permettent en outre de déduire que les variables d'écart ξ_i sont nulles pour tous les vecteurs supports associés à des multiplicateurs α_i tels que $0 < \alpha_i < C$, ce qui permet de calculer b de la même façon que dans le cas séparable.

Pour plus de détails concernant le calcul des SVM nous invitons le lecteur à consulter [Burges, 1998, Shölkopf et Smola, 2002].

Nous terminons cette partie par des considérations pratiques sur l'implémentation des SVM. Le problème d'optimisation se récrit matriciellement sous la forme :

maximiser

$$L_D(\boldsymbol{\alpha}) = \boldsymbol{\alpha}^T \mathbf{1} + \frac{1}{2}\boldsymbol{\alpha}^T \mathbf{H} \boldsymbol{\alpha} \tag{V.46}$$

sous les contraintes

$$\boldsymbol{\alpha}^T \mathbf{y} = 0, \tag{V.47}$$

$$0 \leq \boldsymbol{\alpha} \leq C, \tag{V.48}$$

où $\mathbf{H}$ est la matrice définie par $(\mathbf{H})_{i,j} = y_i y_j \mathbf{x}_i.\mathbf{x}_j$ et $\mathbf{1}$ est un vecteur de taille l dont tous les éléments sont égaux à la constante 1. La taille de la matrice $\mathbf{H}$ est l^2. Pour les tâches d'apprentissage impliquant un nombre élevé d'exemples l, il devient impossible de stocker $\mathbf{H}$ en mémoire (et il est très coûteux de recalculer cette matrice à plusieurs reprises). En conséquence, il est nécessaire de faire appel à des techniques permettant de contourner ce problème. La stratégie qui est utilisée dans l'implémentation des SVM que nous utilisons[4] consiste à décomposer le problème en sous-problèmes de tailles plus petites qui sont résolus successivement jusqu'à l'obtention d'une solution optimale [Joachims, 1999]. L'algorithme correspondant est donné ci-après (Algorithme 2).

Algorithme 2 Calcul des SVM par décomposition.

tant que (les conditions d'optimalité ne sont pas remplies) **faire**

- Sélectionner θ variables α_i pour l'ensemble de travail $\mathcal{B}$, les $l-\theta$ variables restantes gardent leur valeurs en cours.
- Optimiser $L_D(\boldsymbol{\alpha})$ sur $\mathcal{B}$.

fin tant que

Sorties: Arrêter les itérations et retourner $\boldsymbol{\alpha}$.

Nous reviendrons sur le choix du paramètre θ dans la partie expérimentale VII.

[4] il s'agit de *SVMLight* [Joachims,]

D. SVM non-linéaires

1) Principe

Il s'agit de doter les SVM d'un mécanisme permettant de produire des surfaces de décision non-planes. L'idée est de transformer les données de l'espace de départ $\mathbb{R}^d$ dans un espace de Hilbert $\mathbb{E}$ de dimension supérieure (possiblement infinie) dans lequel les données transformées deviennent linéairement séparables. Ainsi, en exploitant une application

$$\Phi : \mathbb{R}^d \rightarrow \mathbb{E}, \tag{V.49}$$

l'algorithme SVM linéaire appliqué aux données $\Phi(\mathbf{x}_i)$ dans l'espace $\mathbb{E}$ produit des surfaces de décision non-planes dans l'espace $\mathbb{R}^d$ (mieux appropriées aux données de départ pour un choix judicieux de Φ).

Cette procédure peut être rendue très efficace en utilisant une astuce permettant d'effectuer les calculs nécessaires à l'algorithme dans l'espace de départ $\mathbb{R}^d$ sans passer explicitement dans $\mathbb{E}$.

Du fait que les données apparaissent dans tous les calculs uniquement sous forme de produits scalaires $(\mathbf{x}_i.\mathbf{x}_j)$, il suffit de trouver une façon efficace de calculer $\Phi(\mathbf{x}_i).\Phi(\mathbf{x}_j)$. Cela est réalisé en faisant appel à une fonction *noyau* $k(\mathbf{x}_i, \mathbf{x}_j)$, définie par :

$$k(\mathbf{x}_i, \mathbf{x}_j) = \Phi(\mathbf{x}_i).\Phi(\mathbf{x}_j). \tag{V.50}$$

Tout le développement présenté dans la section V-2-C reste valable en remplaçant simplement les termes $\mathbf{x}_i.\mathbf{x}_j$ par $k(\mathbf{x}_i, \mathbf{x}_j)$. La nouvelle fonction de décision est définie par le signe de :

$$f(\mathbf{x}) = \sum_{i=1}^{n_s} \alpha_i y_i k(\mathbf{s}_i, \mathbf{x}) + b \tag{V.51}$$

où les $\mathbf{s}_i$ sont les vecteurs supports.

L'avantage d'une telle approche réside dans le fait qu'il n'est pas nécessaire de connaître Φ explicitement. Il suffit d'obtenir des noyaux convenables. C'est ce que nous discutons dans la section suivante.

2) Noyaux

Sous quelles conditions une fonction $k(\mathbf{x}, \mathbf{y})$ symétrique est-elle associée à un espace $\mathbb{E}$ et une transformation Φ vers cet espace ?

La réponse est donnée par les conditions de Mercer qui stipulent qu'il existe une application Φ et un développement de $k(\mathbf{x}, \mathbf{y})$ de la forme :

$$k(\mathbf{x}, \mathbf{y}) = \sum_{i=1}^{+\infty} \beta_i \Phi_i(\mathbf{x}).\Phi_i(\mathbf{y}),\ \beta_i \in \mathbb{R}^d \tag{V.52}$$

ce qui traduit le fait que $k(\mathbf{x}, \mathbf{y})$ décrit un produit interne dans un espace $\mathbb{E}$, si et seulement si pour toute fonction $g(\mathbf{x})$ sur $\mathbb{R}^d$, de norme L_2 finie (*i.e.* $\int g(\mathbf{x})^2 d\mathbf{x}$ est finie) la condition suivante est satisfaite :

$$\int k(\mathbf{x}, \mathbf{y}) g(\mathbf{x}) g(\mathbf{y}) d\mathbf{x} d\mathbf{y} \geq 0. \tag{V.53}$$

Différentes formes de noyau (vérifiant les conditions de Mercer) ont été proposées. Nous examinerons :

- le noyau linéaire :

$$k(\mathbf{x}, \mathbf{y}) = \mathbf{x}.\mathbf{y}, \tag{V.54}$$

- le noyau polynômial de degré δ :

$$k(\mathbf{x}, \mathbf{y}) = (\mathbf{x}.\mathbf{y})^{\delta}, \tag{V.55}$$

- le noyau radial (RBF- *Radial Basis Function*) exponentiel :

$$k(\mathbf{x}, \mathbf{y}) = \exp\left(-\frac{||\mathbf{x} - \mathbf{y}||^2}{2\sigma^2}\right). \tag{V.56}$$

Voici quelques propriétés intéressantes de ces deux derniers noyaux.

Noyau polynômial Le noyau polynômial de degré δ correspond à une transformation Φ par laquelle les composantes des vecteurs transformés $\Phi(\mathbf{x})$ sont tous les monômes d'ordre δ formés à partir des composantes de $\mathbf{x}$. Par exemple, pour $d = \delta = 2$, le noyau

$$k(\mathbf{x}, \mathbf{y}) = (\mathbf{x}.\mathbf{y})^2 \tag{V.57}$$

correspond à la transformation

$$\Phi : \mathbb{R}^2 \rightarrow \mathbb{E} = \mathbb{R}^3$$

$$\mathbf{x} = [x_1, x_2]^T \mapsto [x_1^2, x_2^2, x_1 x_2]^T.$$

Le noyau polynômial permet ainsi d'effectuer la classification sur des nouveaux attributs qui sont tous les produits d'ordre δ des attributs de départ.

Il est possible dans ce cas de calculer la dimension $d_{\mathbb{E}}$ de l'espace transformé $\mathbb{E}$ correspondant à un noyau polynômial de degré δ en comptant le nombre de monômes d'ordre δ possibles. Il vient

$$d_{\mathbb{E}} = \mathrm{C}^{\delta}_{\delta+d-1} = \frac{(\delta+d-1)!}{\delta!(d-1)!}. \tag{V.58}$$

A titre d'exemple, pour des vecteurs d'attributs d'entrée de dimension 40, la dimension de l'espace transformé avec un noyau polynômial de degré 4 est égale à 123,410.

Un exemple de réalisation des SVM munies d'un noyau polynômial de degré 2, sur des données audio réelles est donné dans la figure V.5.

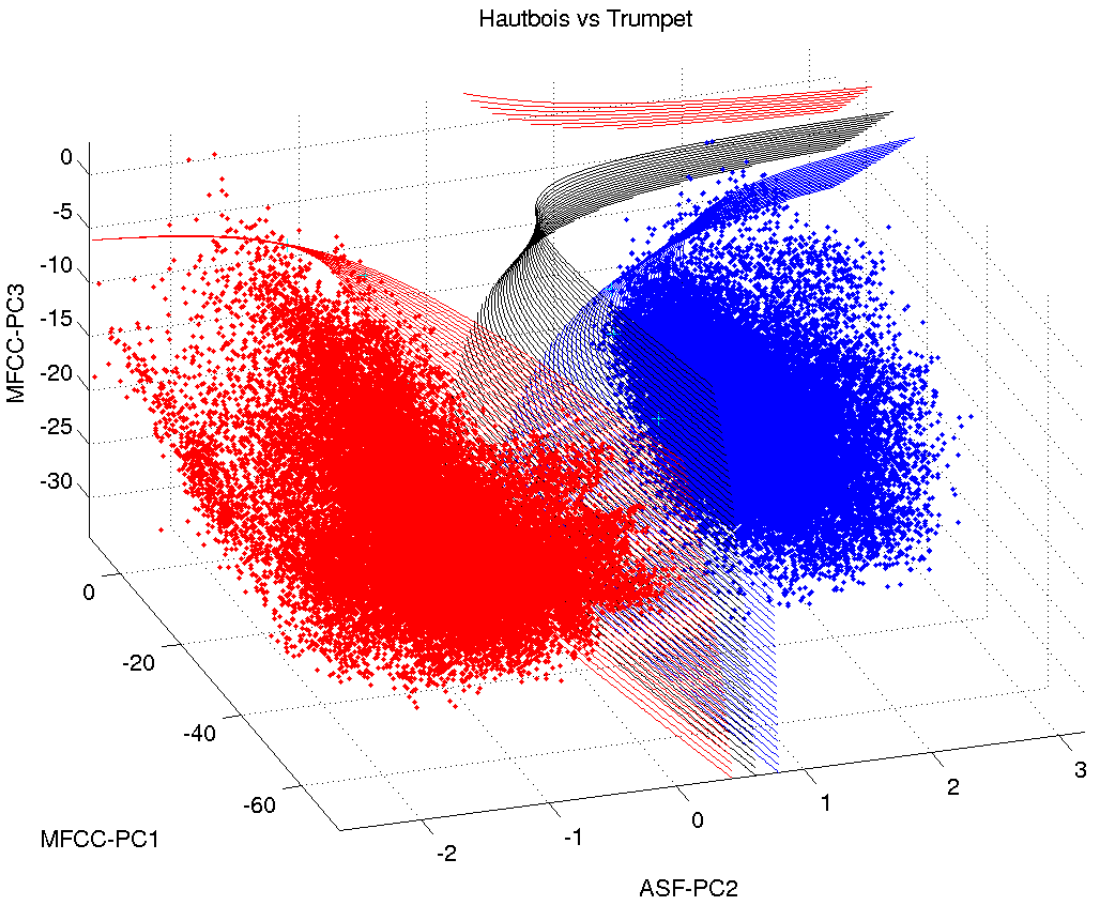

Fig. V.5 Un exemple sur des données audio réelles. Visualisation des surfaces de décisions induites par un noyau polynômial de degré 2 pour la SVM hautbois contre trompette. En bleu (respectivement rouge), les exemples d'apprentissage, ici des vecteurs d'attributs tridimensionnels, de la classe hautbois (respectivement trompette) et les surfaces correspondant aux hyperplans $\mathcal{H}_1$ et $\mathcal{H}_2$. Les surfaces induites par l'hyperplan optimal sont tracées en noir.

Signalons qu'il est également possible de recourir à des noyaux polynômiaux dits *in-homogènes* de la forme :

$$k(\mathbf{x}, \mathbf{y}) = (\mathbf{x}.\mathbf{y} + 1)^{\delta}, \tag{V.59}$$

qui permettent de prendre en compte tous les monômes d'ordre inférieur ou égal à δ.

Noyau exponentiel La figure V.6 montre les surfaces de décision correspondant à des valeurs croissantes de σ. On peut constater que ce paramètre permet de contrôler la courbure des surfaces de décision. A des σ élevés correspondent des surfaces présentant des courbures plus importantes.

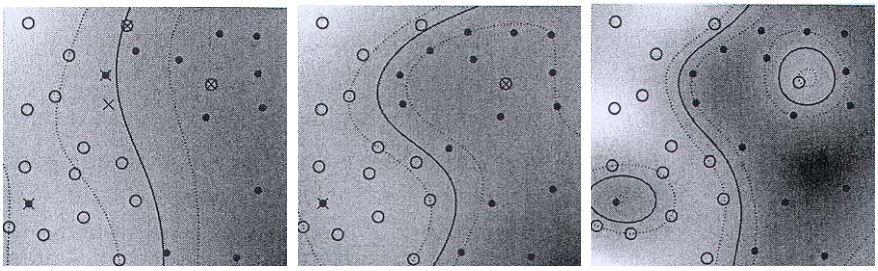

Fig. V.6 Effet du paramètre σ, d'après [Shölkopf et Smola, 2002]. De gauche à droite le paramètre σ^2 est diminué. Les lignes continues indiquent les surfaces de décision et les lignes interrompues les bords de la marge. Notons que pour les grandes valeurs de σ^2, le classificateur est quasi linéaire et la surface de décision ne parvient pas à séparer les données correctement. A l'autre extrême, les valeurs trop faibles de σ^2 donnent lieu à des surfaces de décision qui suivent de trop près la structure des données d'apprentissage et il y a un risque de sur-apprentissage. Il est donc nécessaire de réaliser un compromis tel que celui réalisé dans l'image du milieu.

Il est montré que les exemples transformés $\Phi(\mathbf{x}_1), ..., \Phi(\mathbf{x}_l)$ sont linéairement indépendants. Ils génèrent un sous-espace de $\mathbb{E}$ de dimension l. Par suite, le noyau gaussien défini sur un nombre infini d'exemples d'apprentissage transpose les attributs dans un espace de dimension infinie.

Espace RKHS (Reproducing Kernel Hilbert Space) Étant donnés un noyau k et des exemples $\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_l \in \mathbb{R}^d$, la matrice de Gram de k par rapport à $\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_l$ est définie par

$$\mathbf{K}(\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_n) = \begin{pmatrix} k(\mathbf{x}_1, \mathbf{x}_1) & k(\mathbf{x}_1, \mathbf{x}_2) & \dots & k(\mathbf{x}_1, \mathbf{x}_l) \\ k(\mathbf{x}_2, \mathbf{x}_1) & k(\mathbf{x}_2, \mathbf{x}_2) & \dots & k(\mathbf{x}_2, \mathbf{x}_l) \\ \vdots & \vdots & \vdots & \vdots \\ k(\mathbf{x}_l, \mathbf{x}_1) & \dots & \dots & k(\mathbf{x}_l, \mathbf{x}_l) \end{pmatrix} \tag{V.60}$$

Lorsque $\mathbf{K}$ est définie positive, le noyau k est dit *défini positif*. L'intérêt d'un tel noyau est qu'il permet de définir de façon assez simple une application Φ vers un espace muni d'un produit scalaire décrit par k, en considérant :

$$\begin{aligned} \Phi : \quad & \mathcal{X} \to \mathbb{R}^{\mathcal{X}} \\ & \mathbf{x} \mapsto k(., \mathbf{x}) \end{aligned}$$

où $\mathcal{X}$ est un ensemble non-vide d'exemples et $\mathbb{R}^{\mathcal{X}} := \{f : \mathcal{X} \mapsto \mathbb{R}\}$. $\Phi(\mathbf{x})$ est ainsi la fonction associant à chaque exemple $\mathbf{x}_i$ la fonction $k(\mathbf{x}_i, \mathbf{x})$. $\mathbb{R}^{\mathcal{X}}$ est un espace de fonctions appelé *Reproducing Kernel Hilbert Space* (dans le cas où toutes les fonctions évaluant les éléments de $\mathbb{R}^{\mathcal{X}}$ sur les exemples $\mathbf{x}_i$ sont continues). Pour plus de détails concernant ces espaces, nous invitons le lecteur à consulter [Shölkopf et Smola, 2002].

E. Performances en généralisation des SVM

Les SVM présentent en pratique de très bonnes performances en généralisation (c'est-à-dire sur le classification de nouveaux exemples de test). Intuitivement, on sent que la marge joue en cela un rôle important. Il est en effet raisonnable de penser que si l'on parvient à séparer les exemples d'apprentissage (supposés significatifs des classes auxquelles ils appartiennent) avec une grande marge, il y a de fortes chances pour que de nouveaux exemples soient bien classés, ces derniers se situant dans les cas les plus défavorables à l'intérieur de la marge (ceux se retrouvant loin de la marge et du bon côté de l'hyperplan ne posant pas de problèmes).

Une autre caractéristique frappante des SVM est qu'ils sont connus pour défier ce que l'on appelle "*the curse of dimensionality*" puisqu'ils sont capables de fournir des bonnes performances de classification à partir d'un nombre réduit d'exemples d'apprentissage tout en agissant dans des espaces de dimensions très élevés. Cela s'explique en partie par le fait que les SVM peuvent être considérés comme une réalisation du principe SRM. C'est ce que nous présentons dans la section suivante.

1) Utilisation du principe SRM

En faisant quelques hypothèses sur la structure des données d'apprentissage, il est possible de présenter les SVM comme une réalisation du principe de Minimisation du Risque Structurel.

Cela permet de se faire une idée[5] des performances auxquelles on peut s'attendre en utilisant ces classificateurs, au travers de la borne que l'on peut obtenir sur le risque.

On suppose dans un premier temps qu'on peut trouver la plus petite boule $B(\mathbf{a}, r)$ de centre $\mathbf{a}$ et de rayon r :

$$B(\mathbf{a}, r) = \{\mathbf{x} \in \mathbb{R}^d; ||\mathbf{x} - \mathbf{a}|| < r\}$$

contenant les points $\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_l$. En considérant les fonctions de décision $f_{\mathbf{w},b}$ telles que :

$$\begin{aligned} f_{\mathbf{w},b} : \ B(\mathbf{a}, r) &\rightarrow \{-1, +1\}, \\ \mathbf{x} &\mapsto f_{\mathbf{w},b}(\mathbf{x}) = \text{signe}(\mathbf{w}.\mathbf{x} + b), \end{aligned} \tag{V.61}$$

avec la contrainte

$$||\mathbf{w}|| \leq A, \tag{V.62}$$

$A \in \mathbb{R}_+$, une structure sur les hyperplans est introduite. Il est montré [Vapnik, 1995] que les fonctions de décision ainsi construites ont des dimensions VC, h vérifiant :

$$h \leq r^2 A^2. \tag{V.63}$$

La contrainte (V.62) permet ainsi de contrôler la dimension VC des classificateurs obtenus et de déterminer une borne sur le risque fonctionnel associé (en exploitant (V.17)). Pour cela h est estimée par [Shölkopf et Smola, 2002] :

$$h \approx r^2 ||\mathbf{w}||^2, \tag{V.64}$$

en supposant que les bornes (V.62) et (V.63) sont atteintes.

2) Erreur de classification $\xi\alpha$

Joachims propose une autre manière d'estimer les performances en généralisation des SVM [Joachims, 2000] qui ne se base pas sur des hypothèses structurelles. Il définit une estimation de l'erreur de classification, appelée $\xi\alpha$, par :

$$E_{\xi\alpha} = \frac{\eta}{l} \quad \text{avec } \eta = \text{card}(\{i \ ; (2\alpha_i r_\Delta^2 + \xi_i) \geq 1\}), \tag{V.65}$$

[5] Il est admis que les arguments qui vont suivre ne permettent pas d'expliquer *rigoureusement* les bonnes performances en généralisation des SVM [Burges, 1998] obtenues en pratique. Cela est dû au fait que les hypothèses faites sur la structure des données ne sont pas toujours strictement vérifiées. Ils restent néanmoins raisonnables et utiles comme nous allons le voir.

où r_Δ^2 est une borne supérieure sur $k(\mathbf{x},\mathbf{x})$ et $k(\mathbf{x},\mathbf{y})$ pour tout $\mathbf{x}$, $\mathbf{y}$.

η est le nombre d'exemples d'apprentissage pour lesquels l'inégalité $2\alpha_i r_\Delta^2 + \xi_i \geq 1$ est vérifiée. L'idée de cette borne est que tout exemple $\mathbf{x}_i$ mal-classé par la SVM entraînée sur le sous-ensemble d'apprentissage contenant tous les points à l'exclusion de $\mathbf{x}_i$, vérifient cette inégalité. Par conséquent, η est une borne supérieure sur le nombre d'erreurs commises dans les schémas classant chaque exemple d'apprentissage à l'aide de machines calculés à partir de tous les autres exemples (*leave-one-out errors*).

Joachims montre que l'estimation de l'erreur en généralisation ainsi obtenue, s'avère efficace, en particulier pour prédire les performances des SVM dans la tâche de la classification de textes.

Cette technique d'estimation de l'erreur en généralisation des SVM, ainsi que celle présentée dans la section V-2-E.1, seront mises à profit pour sélectionner, à partir de l'ensemble d'apprentissage, les noyaux à utiliser dans notre système de classification. Nous verrons cela dans la partie expérimentale, au chapitre VII.

F. Réalisations multi-classes des SVM et SVM probabilisés

Les SVM peuvent être utilisés dans des schémas de classification multi-classes en exploitant une stratégie du type "un contre un" ou "un contre tous"[6]. Nous adoptons l'approche "un contre un". Une méthode permettant de fusionner les décisions prises dans les différents schémas binaires est de nouveau requise. Nous utilisons pour cela l'approche de Platt [Platt, 1999], qui permet d'obtenir des sorties probabilistes pour les SVM. Les probabilités à posteriori $P(y=1|f)$ (f étant donnée par (V.51)) sont alors modélisées par

$$P(y=1|f) = \frac{1}{1+\exp(Af+B)}, \tag{V.66}$$

où A et B sont des paramètres à déterminer. Platt montre la pertinence de ce modèle et propose un algorithme permettant de déterminer les valeurs optimales de A et B. Il devient ainsi possible de fusionner les sorties probabilistes par le biais de la méthode de Hastie & Tibshirani, décrite dans la section V-1-B.2.

[6] dans ce cas on construit des machines qui séparent chaque classe de toutes les autres.

V-3. Clustering

Le *clustering* intervient dans des tâches d'apprentissage non-supervisé où les étiquettes des exemples d'apprentissage ne sont pas connues à priori. Le but est de trouver une organisation du nuage de points correspondant aux exemples d'apprentissage, en M régions ou cellules, appelées *clusters*. Nous ferons appel au clustering dans un contexte particulier, dans lequel il ne sera pas appliqué "directement" à des vecteurs d'attributs, mais à des classes connues à priori : nous verrons que nous aurons besoin de réorganiser ces classes, en regroupant celles qui sont les plus "proches" les unes des autres, dans des clusters (de classes), formant ainsi des *super-classes*. Cela interviendra à différentes étapes :

- pour le clustering des attributs (*cf.* section VI-7-A), où les classes à regrouper seront des classes d'attributs ;
- pour la construction d'une taxonomie des instruments de musique (*cf.* section IX-3-B) et des ensembles d'instruments (*cf.* section X-2-A), où les classes à regrouper seront des classes d'instruments ou de mélanges d'instruments.

Nous utiliserons une approche de clustering particulière : le *clustering hiérarchique*, que nous décrivons ci-après.

A. Principe du clustering hiérarchique

Le *clustering hiérarchique* permet d'obtenir une hiérarchisation des clusters. Cela nous sera utile dans la construction de *taxonomies hiérarchiques*. Outre cette organisation particulière des données, cette approche présente l'avantage de ne nécessiter aucune étape d'initialisation et de retarder le choix des clusters à considérer à la fin du traitement qui mène à l'obtention d'une hiérarchie de clusters emboîtés les uns dans les autres [Duda *et al.*, 2001, Theodoridis et Koutroumbas, 1998].

La version *agglomérative* de ces algorithmes démarre avec autant de clusters M_c que de classes originales ($M_c^1 = Q$ à la première itération, où Q est le nombre de classes à organiser), mesure les proximités J_{pq} entre toutes les paires de clusters $\{\mathrm{C}_p,\mathrm{C}_q\}$ et regroupe les paires les plus proches dans de nouveaux clusters, pour en produire M_c^m nouveaux à l'itération m, et ce jusqu'à ce que toutes les classes de départ se retrouvent au sein d'un même cluster (à l'itération Q).

Pour mieux comprendre le résultat de cette procédure, on le représente généralement à l'aide d'un graphe, appelé *dendrogramme* qui fait apparaître les relations et les proximités entre les

clusters emboîtés obtenus. Un exemple est donné dans la figure V.7. Les clusters qui sont regroupés à des niveaux supérieurs sont reliés par des lignes en U. Les clusters de départ (ce sont les classes de départ) sont donnés le long de l'axe vertical, alors que l'axe horizontal représente les distances entre clusters. La distance entre deux clusters C_p et C_q est calculée ici comme la distance moyenne entre toutes les paires de classes appartenant à C_p et C_q. Par exemple, le dendrogramme donné nous renseigne sur le fait que les classes de départ C_1 et C_3 ont été regroupées en un nouveau cluster qui, à son tour, est relié à la classe C_6.

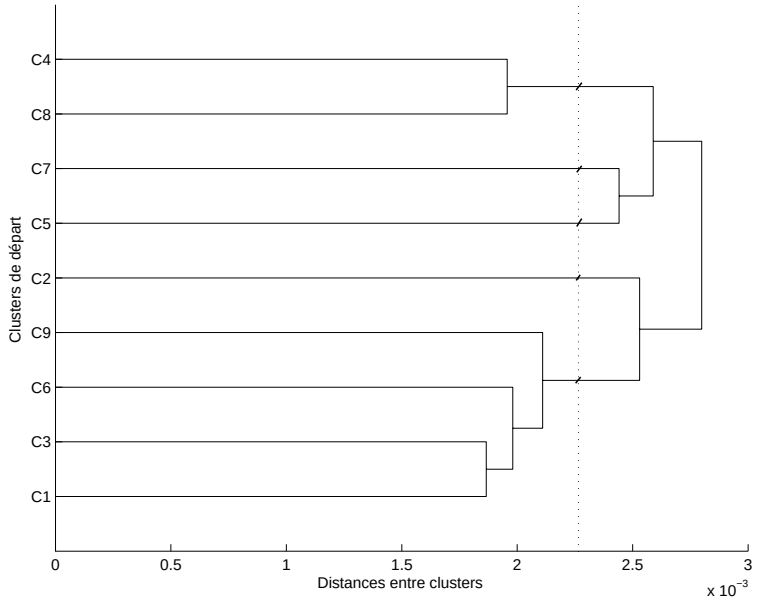

Fig. V.7 Exemple de dendrogramme.

La pertinence de l'*arbre de clusters* ainsi obtenu peut être évaluée par le biais du *coefficient de corrélation cophénétique*. Ce coefficient corrèle des distances J_{pq} entre n'importe quels clusters initiaux (*i.e.* classes initiales) C_i et C_j aux *distances cophénétiques* δ_{pq}, c'est-à-dire les distances entre les clusters C_p et C_q contenant ces deux classes et reliés ensemble à un niveau donné de la hiérarchie. Par exemple, la distance cophénétique entre C_1 et C_6 est la distance entre les clusters C_6 et C_{13}, où C_{13} est le cluster contenant C_1 et C_3. Le coefficient de corrélation cophénétique est défini par :

$$c = \frac{\sum_{p<q}(J_{pq} - \bar{J})(\delta_{pq} - \bar{\delta})}{\sqrt{\sum_{p<q}(J_{pq} - \bar{J})^2 \sum_{p<q}(\delta_{pq} - \bar{\delta})^2}} , \tag{V.67}$$

où $\bar{J}$ et $\bar{\delta}$ sont respectivement les moyennes de J_{pq} et de δ_{pq}, $1 \leq p < q \leq M$. Plus le coefficient

cophénétique est proche de 1, meilleure est l'adéquation entre le dendrogramme obtenu et la structure des données de départ.

En réalisant une coupe du dendrogramme selon une valeur particulière de l'axe horizontal, une solution de clustering est réalisée. Par exemple, la ligne verticale en pointillé apparaissant dans la figure V.7, produit 5 clusters. Ainsi, il est possible d'obtenir un nombre de clusters souhaité, simplement en ajustant la position de cette ligne verticale.

B. Critères de proximité

Le choix d'un critère de proximité entre classes, *i.e.* la distance J_{pq} à utiliser pour le clustering, est critique. Nous avons besoin d'une distance (entre classes) robuste, capable de limiter l'effet des attributs bruités. Une solution convenable consiste à faire appel à des distances probabilistes, c'est-à-dire des distances entre distributions de probabilités des classes [Theodoridis et Koutroumbas, 1998, Duda *et al.*, 2001]. Il s'agit d'une alternative intéressante à celle plus classique, qui exploite la distance euclidienne entre vecteurs de données des différentes classes, connue pour être sous-optimale dans le cas de données audio-fréquences. Plusieurs variantes de ces distances ont été définies dans différentes branches de la recherche [Zhou et Chellappa, 2006]. Nous choisissons d'expérimenter, dans notre étude, la distance de Bhattacharryya et la divergence (version symétrisée de la distance de Kullback-Leibler). Ce choix est motivé par la simplification des calculs qui en résulte.

La divergence J_D entre deux densités de probabilités p_1 et p_2 est définie par

$$J_D(p_1,p_2) = \int_{\mathbf{x}} [p_1(\mathbf{x}) - p_2(\mathbf{x})] \log \frac{p_1(\mathbf{x})}{p_2(\mathbf{x})} d\mathbf{x}. \tag{V.68}$$

La distance de Bhattacharryya est définie par :

$$J_B(p_1,p_2) = -\log\left(\int_{\mathbf{x}} [p_1(\mathbf{x})p_2(\mathbf{x})]^{\frac{1}{2}} d\mathbf{x}\right). \tag{V.69}$$

Si les densités de probabilités peuvent être considérées comme gaussiennes, les distances ci-dessus admettent des expressions analytiques et peuvent être calculées selon :

$$J_D(p_1,p_2) = \frac{1}{2}(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)^T(\boldsymbol{\Sigma}_1^{-1} + \boldsymbol{\Sigma}_2^{-1})(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2) + \frac{1}{2}\mathrm{tr}(\boldsymbol{\Sigma}_1^{-1}\boldsymbol{\Sigma}_2 + \boldsymbol{\Sigma}_2^{-1}\boldsymbol{\Sigma}_1 - 2\mathbf{I}_D), \tag{V.70}$$

$$J_B(p_1,p_2) = \frac{1}{8}(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)^T[\frac{1}{2}(\boldsymbol{\Sigma}_1 + \boldsymbol{\Sigma}_2)]^{-1}(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2) + \frac{1}{2}\log\frac{\left|\frac{1}{2}(\boldsymbol{\Sigma}_1 + \boldsymbol{\Sigma}_2)\right|}{|\boldsymbol{\Sigma}_1|^{\frac{1}{2}}|\boldsymbol{\Sigma}_2|^{\frac{1}{2}}}, \tag{V.71}$$

où $(\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1)$ et $(\boldsymbol{\mu}_2, \boldsymbol{\Sigma}_2)$ sont, respectivement, les vecteurs de moyennes et les matrices de covariance des densités de probabilité gaussiennes multivariées, décrivant respectivement la classe Ω_1 et la classe Ω_2 dans $\mathbb{R}^D$.

Cependant, l'hypothèse de gaussianité est souvent mise à mal (c'est le cas par exemple pour les distributions des vecteurs d'attributs des classes d'instruments). Or il s'avère coûteux de calculer ces distances dans le cas non-gaussien.

Nous suivons donc l'approche de Zhou & Chellapa qui font appel à une méthode à noyau [Zhou et Chellappa, 2006]. L'idée est de transformer, à l'aide d'un noyau (*cf.* section V-2-D), les données de départ (de $\mathbb{R}^d$) dans un espace de dimension supérieure $\mathbb{R}^F$ ($F >> d$), où elles deviennent linéairement séparables. Les auteurs présentent une discussion sur le fait que dans l'espace transformé, les densités de probabilité des données peuvent être considérées comme gaussiennes. Par conséquent, si une estimation des moyennes et des matrices de covariances dans l'espace de dimension supérieure est obtenue, une estimation robuste des distances probabilistes requises peut être calculée en utilisant les expressions (V.70) et (V.71).

L'avantage de l'approche proposée est qu'il n'est pas nécessaire de connaître explicitement ni la structure des densités de probabilités originales, ni la transformation vers l'espace de dimension supérieure. En effet, il est montré que tous les calculs peuvent être effectués en utilisant "l'astuce du noyau" (*the kernel trick*).

Pour obtenir les distances (V.70) et (V.71) dans l'espace transformé, Zhou & Chellapa exploitent l'estimation au sens du maximum de vraisemblance des moyennes et des covariances dans $\mathbb{R}^F$ à partir de l vecteurs d'observations de $\mathbf{x}_i \in \mathbb{R}^D$:

$$\boldsymbol{\mu}_q = \frac{1}{l} \sum_{i=1}^{l} \Phi(\mathbf{x}_i) \ , q \in \{1, 2\} \tag{V.72}$$

$$\boldsymbol{\Sigma}_q = \frac{1}{l} \sum_{i=1}^{l} (\Phi(\mathbf{x}_i) - \boldsymbol{\mu}_q)(\Phi(\mathbf{x}_i) - \boldsymbol{\mu}_q)^T. \tag{V.73}$$

La difficulté majeure qui est rencontrée vient du fait que la matrice de covariance $\boldsymbol{\Sigma}_q$ doit être inversée alors qu'elle est déficiente en rang car $F >> l$. Zhou & Chellapa parviennent à obtenir une approximation de $\boldsymbol{\Sigma}_q$ qui est inversible et des expressions des distances probabilistes ne faisant intervenir que la connaissance du noyau. Les expressions de ces distances sont données dans l'annexe A.

VI. Sélection automatique des attributs

Dans ce chapitre nous étudions des algorithmes récents de sélection automatique des attributs. Nous comparons leur efficacité sur les données audio et proposons des améliorations structurelles permettant d'atteindre de meilleures performances de classification.

VI-1. Introduction

Dans la plupart des problèmes de classification, un nombre important d'attributs potentiellement utiles peut être exploré. Ce nombre atteint, dans plusieurs cas d'application, les quelques centaines, voire quelques milliers (en particulier, dans le domaine de la bioinformatique). L'objet de la sélection d'attributs est de produire à partir des D variables initialement considérées, un sous-ensemble "optimal" de d attributs (généralement $d << D$). Il s'agit là d'une problématique de recherche qui suscite depuis une dizaine d'années un intérêt croissant de la part de la communauté de l'apprentissage artificiel" [Kohavi et John, 1997, Blum et Langley, 1997, Liu et Motoda, 2000, Guyon et Elisseeff, 2003]. Nous l'introduisons en nous posant les deux questions suivantes :

- *Pourquoi réduire l'ensemble d'attributs de départ ?*
- *Qu'est-ce qu'une sélection d'attributs "optimale" ?*

Ces deux questions admettent plusieurs réponses et nous proposons ici quelques unes des plus intuitives en renvoyant le lecteur aux références [Liu et Motoda, 2000, Theodoridis et Koutroumbas, 1998, Guyon et Elisseeff, 2003] pour un traitement plus complet du sujet.

La réduction de la complexité s'impose comme une réponse évidente à la première question. Une dimension élevée implique une charge de stockage et de calcul, et des temps de réponse importants qui peuvent être intolérables pour l'utilisateur final. Il serait aberrant de ne pas réduire la dimension du problème si des performances équivalentes peuvent être atteintes en

travaillant en dimension plus faible. En outre, une dimension trop élevée conduit à de moins bonnes performances en généralisation (avec la plupart des classificateurs) puisqu'il devient de plus en plus compliqué de modéliser l'espace des attributs, qui est d'autant plus étendu que la dimension est élevée.

Savoir qu'il existe parmi les descripteurs disponibles de nombreux attributs non-pertinents (issus de descripteurs mal-construits ou mal-appropriés à la tâche de classification considérée), bruités (par manque de robustesse de leur extraction) et/ou redondants les uns avec les autres, conduit à adopter une stratégie permettant de sélectionner les plus "efficaces". Cela nous amène à réfléchir à la deuxième question.

Une distinction intéressante peut être faite entre "sélection d'attributs pertinents" et "sélection efficace" ou "utile" [Kohavi et John, 1997, Blum et Langley, 1997]. Par "sélection efficace" on entend produire un sous-ensemble d'attributs conduisant aux meilleures performances de classification, ce qui signifie qu'il n'est pas nécessaire de garder toutes les variables pertinentes, particulièrement en présence d'attributs redondants. Une "sélection efficace" ne sélectionne pas des attributs redondants même si ceux-ci peuvent être pertinents, puisque de bonnes performances de classification sont atteintes en utilisant un sous-ensemble d'attributs complémentaires. Il n'en reste pas moins que la présence de variables redondantes ne fait que consolider la séparabilité des classes (*cf.* [Guyon et Elisseeff, 2003]). Il est également prouvé que si des attributs sont inefficaces isolément, leur combinaison peut, quant à elle, s'avérer très utile.

Ainsi, les Algorithmes de Sélection d'Attributs (ASA) s'organisent en trois groupes principaux :

- les "*filters*" exploitent les attributs disponibles de façon intrinsèque, indépendamment du traitement envisagé par la suite ; ils effectuent un classement des attributs basé sur l'obtention d'un score individuel de pertinence ;
- les "*wrappers*" sélectionnent un sous-ensemble d'attributs qui permet d'atteindre les meilleures performances finales dans le cadre de l'application envisagée, dans notre cas, les performances de classification ;
- enfin, les "*embedders*" dont l'idée est assez proche des *wrappers*, intègrent en un seul processus, l'optimisation conjointe du sous-ensemble d'attributs et du classificateur.

Quelques travaux sur la reconnaissance des instruments de musique ont eu recours à la sélection automatique des attributs [Fujinaga, 1998, Martin, 1999, Eronen, 2001a, Peeters et Rodet, 2002, Peeters, 2003].

Nous proposons ici une étude comparative du comportement de différents algorithmes de sélection (choisis parmi les *filters* et les *wrappers*) sur les données audio. Certains de ces algorithmes ont été utilisés dans le cadre de la reconnaissance automatique des instruments, d'autres sont des algorithmes récents connus pour leur efficacité. En outre, des améliorations structurelles sont proposées qui permettent d'atteindre de meilleures performances de classification.

Nous commençons par une description des pré-traitements effectués sur les données (préalablement à la sélection). Nous introduisons une technique alternative de réduction de la dimension des données par Analyse en Composantes Principales et nous proposons une brève présentation des algorithmes étudiés et de critères d'évaluation de leurs performances. Ensuite, nous analysons les résultats de l'étude expérimentale entreprise et nous proposons des améliorations au fonctionnement des algorithmes de sélection.

VI-2. Normalisation des données

Les valeurs de plusieurs attributs, notamment issus de descripteurs de nature physique différente, présentent souvent des dynamiques assez hétérogènes. A titre d'exemple, les variables mesurant la variation d'attributs sur des trames successives (dérivées temporelles) présentent typiquement des valeurs très petites par rapport aux valeurs intra-trames. Les attributs possédant des valeurs plus grandes risquent alors d'avoir une influence plus importante sur le comportement des différents traitements à suivre (sélection, transformation, classification), même si cela ne reflète pas forcément leur pertinence pour la tâche envisagée.

Afin de contourner ce problème, il est classiquement fait appel à des techniques de normalisation permettant d'uniformiser les dynamiques des différentes variables. Habituellement, cette normalisation est réalisée de façon linéaire en exploitant les estimations empiriques (à partir de l'ensemble d'apprentissage) des moyennes et des variances des attributs [Theodoridis et Koutroumbas, 1998] définies pour le j-ème attribut et pour l exemples par :

$$\mu_j = \frac{1}{l}\sum_{k=1}^{l} x_{k,j} \quad , \quad 1 \leq j \leq D \tag{VI.1}$$

$$\sigma_j^2 = \frac{1}{l-1}\sum_{k=1}^{l}(x_{k,j} - \mu_j)^2. \tag{VI.2}$$

La normalisation que nous désignons par "normalisation $\mu\sigma$" consiste alors à prendre

$$\hat{x}_{k,j} = \frac{x_{k,j} - \mu_j}{\sigma_j}, \tag{VI.3}$$

ce qui a pour effet d'assurer que les attributs normalisés possèdent une moyenne nulle et une variance unitaire.

Alternativement, la normalisation peut être effectuée en ramenant la dynamique des attributs dans l'intervalle [-1,1]. Cela s'obtient en estimant (à partir de l'ensemble d'apprentissage) les valeurs maximales de chaque attribut (en valeur absolue) :

$$\tilde{x}_j = \max_{1 \leq j \leq N} |x_{k,j}| \tag{VI.4}$$

et en prenant

$$\breve{x}_{k,j} = \frac{x_{k,j}}{\tilde{x}_j}. \tag{VI.5}$$

Cette normalisation sera désignée par "normalisation min-max". Ces deux types de normalisation seront étudiées et comparées dans la suite.

Signalons que d'autres pré-traitement peuvent être considérés [Theodoridis et Koutroumbas, 1998], parmi lesquelles la sélection d'exemples d'apprentissage (observations de vecteurs d'attributs) peut s'avérer utile dans le cas où les traitements envisagés présentent une sensibilité accrue aux attributs bruités. Ces méthodes se basent souvent sur le fait qu'une large proportion des exemples d'apprentissage sont proches de la moyenne. Par exemple, pour une distribution gaussienne, 95% des exemples sont distants de la moyenne de moins de deux fois l'écart-type : $|x_i - \mu_i| \leq 2\sigma_i$ (une distance de trois fois l'écart-type couvre 99% des points). Les points situés "trop loin" de la moyenne sont alors éliminés, typiquement en exploitant un seuil de "quelques fois" l'écart-type.

VI-3. Transformation des attributs par Analyse en Composantes Principales (PCA)

L'*Analyse en Composantes Principales* (PCA- Principal Component Analysis) permet de transformer les vecteurs d'attributs de telle sorte que les vecteurs transformés concentrent le maximum d'information sur leurs premières composantes. Dans un premier temps, une décomposition en valeurs propres de la matrice de covariance $\mathbf{R_x}$ des vecteurs d'apprentissage est calculée :

$$\mathbf{R_x} = \mathbf{V}\mathbf{\Lambda}\mathbf{V}^t, \tag{VI.6}$$

où $\mathbf{\Lambda}$ est la matrice des valeurs propres, que nous supposons ordonnées par ordre décroissant dans $\mathbf{\Lambda}$, et $\mathbf{V}$ est la matrice des vecteurs propres. La matrice $\mathbf{W} = \mathbf{V}^T$ est alors utilisée pour transformer les vecteurs d'attributs $\mathbf{x}_i$ selon :

$$\mathbf{y}_i = \mathbf{W}\mathbf{x}_i, \tag{VI.7}$$

où les $\mathbf{y}_i$ sont les vecteurs transformés. Les composantes de ces vecteurs sont des combinaisons linéaires des D attributs de départ. Cette transformée est connue en codage sous le nom de Transformée de Karhunen-Loeve (KLT). Les vecteurs $\mathbf{y}_i$ peuvent être tronqués à la dimension d en supposant que l'énergie utile est concentrée sur leurs d premières composantes. Contrairement à la sélection d'attributs, la transformation par PCA ne vise pas à assurer une bonne séparabilité des vecteurs d'attributs en sortie, mais plutôt à obtenir une représentation efficace des attributs.

L'approche PCA nécessite, à l'étape de test, l'extraction de tous les attributs dans des vecteurs de dimension D, avant que la matrice de transformation $\mathbf{W}$ (obtenue à l'étape d'apprentissage) ne puisse être utilisée pour transformer ces vecteurs et réduire leur dimension à d. Cela représente un inconvénient majeur par rapport à la sélection d'attributs qui permet d'éviter le calcul de variables inutiles à l'**étape de test** (seuls les d attributs sélectionnés sont alors effectivement extraits).

Signalons que la PCA, ici utilisée pour **transformer** les attributs, peut servir de base à la **sélection** d'attributs ; on parle alors de Principal Feature Analysis (PFA). Dans [Cohen *et al.*, 2002], par exemple, il est décidé qu'une valeur élevée du i-ème coefficient de l'une des composantes principales implique que la composante $x_{n,i}$ du vecteur d'attributs $\mathbf{x}_n$ est dominante selon cet axe principal. Les variables correspondant aux plus grands coefficients de projection sur les axes principaux dominants sont ainsi sélectionnées.

VI-4. Algorithmes de Sélection des Attributs (ASA)

Dans cette partie nous donnons un bref aperçu des algorithmes de sélection étudiés et testés pour la tâche de classification audio. La plupart de ces algorithmes produisent un score w_i relatif à chaque attribut i, $1 \leq i \leq D$, pour garder ceux qui présentent les scores les plus élevés (les d attributs les mieux classés).

A. Algorithme de Fisher

Cet algorithme s'inspire de l'*Analyse Linéaire Discriminante* (ALD) également appelée Analyse Discriminante de Fisher [Duda *et al.*, 2001]. Contrairement à la PCA, qui cherche à trouver les directions de l'espace "utiles à la représentation des données", l'ALD permet de trouver les directions "utiles à une bonne discrimination des classes".

Dans le cas le plus simple, à deux classes, il s'agit de séparer les exemples de chaque classe, représentés par des points de l'espace affine $\mathbb{R}^D$, à l'aide d'un hyperplan $\mathcal{H}$ défini par :

$$\mathcal{H} : \mathbf{w}.\mathbf{x} + b = 0 \ , \ \ \mathbf{w} \in \mathbb{R}^D, b \in \mathbb{R}. \tag{VI.8}$$

Un exemple $\mathbf{x}_i$ est alors classifié d'après le signe de la fonction

$$g(\mathbf{x}_i) = \mathbf{w}.\mathbf{x}_i + b, \tag{VI.9}$$

donnant la position de $\mathbf{x}_i$ par rapport à l'hyperplan. Cet hyperplan est choisi de façon à obtenir la meilleure projection des données sur une droite (dans la direction de $\mathbf{w}$), celle qui permet le maximum de séparation entre les projections des points appartenant à chaque classe. Cela est réalisé en maximisant le rapport

$$r(\mathbf{w}) = \frac{|\tilde{\mu}_1 - \tilde{\mu}_2|^2}{\tilde{\sigma_1}^2 + \tilde{\sigma_2}^2} \ , \tag{VI.10}$$

appelé *discriminant de Fisher*, où $\tilde{\mu}_q$ et $\tilde{\sigma}_q^2$ sont respectivement la moyenne et la variance empiriques des projections sur $\mathbf{w}$ des exemples appartenant à la classe Ω_q ($1 \leq q \leq 2$). Cela revient à maximiser le rapport entre la *dispersion inter-classe* et *la dispersion intra-classe*.

L'algorithme de sélection que nous utilisons (et dont une implémentation est fournie par la toolbox Spider [Spider,]), s'inspire de ce principe, sans qu'aucune projection des données ne soit réalisée. Cet algorithme procède comme suit :

1) pour chaque attribut i ($1 \leq i \leq D$), des scores intermédiaires w_i^q sont estimés à partir des données de chaque classe Ω_q, $1 \leq q \leq Q$, selon :

$$w_i^q = \sum_{p=1}^{Q} \frac{|\mu_i^p - \mu_i^q|}{\sigma_i^p + \sigma_i^q}. \tag{VI.11}$$

w_i^q est, à une constante de normalisation près, la moyenne des discriminants de Fisher[1]

[1] discriminants approximatifs puisqu'obtenus directement à partir des données sans aucune projection.

relatifs à tous les problèmes bi-classes "1 contre 1" dans lesquelles la classe Ω_q est impliquée.

2) Les éléments de l'ensemble $\mathcal{W} = \{w_i^q\}_{1 \leq i \leq D; 1 \leq q \leq Q}$ sont triés par ordre décroissant et les d attributs occupant les premiers rangs du tri sont sélectionnés, en s'assurant que ceux-ci soient distincts (l'indice du même attribut i apparaissant Q fois dans l'ensemble $\mathcal{W}$, une fois par classe).

B. Inertia Ratio Maximization using Feature Space Projection (IRMFSP)

Cette approche du type *filter* a été proposée et utilisée avec succès pour la reconnaissance automatique des instruments de musique [Peeters, 2003]. Il s'agit d'un algorithme itératif dans lequel, à chaque itération k, un sous-ensemble $\mathcal{S}_{d_k}$ de $d_k = k$ attributs est construit en incluant un attribut supplémentaire au sous-ensemble précédemment sélectionné $\mathcal{S}_{d_k-1}$. A l'itération d, $d_k = d$, et le nombre d'attributs ciblé est atteint.

Soient Q le nombre de classes, l_q le nombre de vecteurs d'attributs (vecteurs d'apprentissage) associés à la classe Ω_q et l le nombre total de vecteurs d'apprentissage ($l = \sum_{q=1}^{Q} l_q$).

Soit $\mathbf{x}_{i_q,d_k}$ le i_q-ème vecteur d'attributs de la classe Ω_q (contenant les d_k attributs sélectionnés à l'itération k), et soit $\boldsymbol{\mu}_{q,d_k}$, respectivement $\boldsymbol{\mu}_{d_k}$, le vecteur de moyenne des exemples $(\mathbf{x}_{i_q,d_k})_{1 \leq i_q \leq l_q}$, respectivement le vecteur de moyenne de tous les exemples $(\mathbf{x}_{i_q,d_k})_{1 \leq i_q \leq l_q;\ 1 \leq q \leq Q}$.

Les attributs sont sélectionnés en se basant sur le rapport r_{d_k} entre l'inertie inter-classes B_{d_k} et le "rayon moyen" de la dispersion intra-classe[2] R_{d_k}, défini par :

$$r_{d_k} = \frac{B_{d_k}}{R_{d_k}} = \frac{\sum_{q=1}^{Q} \frac{l_q}{l} \left\| \boldsymbol{\mu}_{d_k,q} - \boldsymbol{\mu}_{d_k} \right\|}{\sum_{q=1}^{Q} \left(\frac{1}{l_q} \sum_{i_q=1}^{l_q} \left\| \mathbf{x}_{d_k,i_q} - \boldsymbol{\mu}_{d_k,q} \right\| \right)} \tag{VI.12}$$

Le principe est encore inspiré de l'ALD. L'idée est de sélectionner les attributs qui permettent une bonne séparation entre classes (décrite par B_{d_k}) tout en minimisant la dispersion intra-classe (décrite par R_{d_k}). Par conséquent, chaque attribut supplémentaire sélectionné doit réaliser le maximum du rapport r_{d_k}.

Se contenter de ce critère peut donner lieu à une sélection d'attributs redondants, qui ne caractérisent que des propriétés restreintes des classes (même s'ils peuvent conduire à des

[2] Il s'agit là d'une variation sur l'algorithme proposé initialement par Peeters.

valeurs élevées de r_d). Pour prendre en compte la contrainte de non-redondance des attributs sélectionnés, Peeters introduit dans l'algorithme une étape d'orthogonalisation, qui garantit qu'à chaque itération le dernier attribut sélectionné est décorrélé de ceux précédemment sélectionnés [Peeters, 2003]. Cela consiste à rendre, à chaque itération, les vecteurs colonnes de la matrice

$$\boldsymbol{X} = \begin{pmatrix} \mathbf{x}_1^T \\ \vdots \\ \mathbf{x}_o^T \\ \vdots \\ \mathbf{x}_l^T \end{pmatrix} = \begin{pmatrix} x_{1,1} & \dots & x_{1,j} & \dots & x_{1,D} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ x_{o,1} & \dots & x_{o,j} & \dots & x_{o,D} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ x_{l,1} & \dots & x_{l,j} & \dots & x_{l,D} \end{pmatrix}$$

orthogonaux au vecteur formé par toutes les observations de l'attribut sélectionné, $[x_{1,j_0} \dots x_{l,j_0}]^T$, avec j_0 l'indice de colonne, dans $\mathbf{X}$, de l'attribut sélectionné. L'orthogonalisation est réalisée par une procédure de Gram-Shmidt. L'algorithme résultant est présenté ci-après (*cf.* Algorithme 3).

Peeters suggère un critère permettant de déterminer automatiquement le nombre d'attributs d à sélectionner, en arrêtant les itérations lorsque le ratio r_{d_k} mesuré à l'itération k devient beaucoup plus petit que le ratio r_1 mesuré à l'itération 1, c'est-à-dire lorsque $\frac{r_{d_k}}{r_1} < \epsilon$, pour un ϵ fixé. Ce critère présente l'inconvénient que dans le cas où les classes sont difficilement séparables, il devient peu robuste. Des valeurs de d_k trop grandes sont alors atteintes sans que le rapport $\frac{r_{d_k}}{r_1}$ ne devienne assez petit, et l'algorithme se met à sélectionner des attributs bruités (faute de pouvoir sélectionner des attributs redondants, de par sa construction). Par conséquent, nous ne retenons pas ce critère. La valeur de d sera choisie comme pour les autres algorithmes parmi un ensemble de valeurs à tester (*cf.* section VI-6).

C. Algorithme SVM-RFE (Recursive Feature Elimination)

SVM-RFE est un algorithme de type *wrapper* exploitant les SVM de façon récursive pour estimer des scores $\{w_i\}_{1 \leq i \leq D}$ relatifs à chaque attribut. Le score w_i correspondant à l'attribut i est obtenu en moyennant les scores $\{w_i^{pq}\}_{1 \leq p < q \leq Q}$ calculés pour chaque problème bi-classes (Ω_p vs Ω_q) à partir de la machine à vecteurs supports correspondante. Ces scores sont ici simplement les composantes du vecteur de poids $\mathbf{w}^{pq}$, définissant l'hyperplan optimal obtenu pour la paire $\{\Omega_p, \Omega_q\}$ (*cf.* section V-2-B), qui est, nous le rappelons, une combinaison linéaire des n_s vecteurs

Algorithme 3 IRMFSP

Entrées: $\mathbf{X} \leftarrow [\mathbf{x}_1^T, \dots \mathbf{x}_o^T, \dots \mathbf{x}_l^T]^T$ //Exemples d'apprentissage

d //Nombre d'attributs à sélectionner

//Initialisation

$\mathcal{S}_0 \leftarrow \{\}$ //Sous-ensemble des attributs sélectionnés

$d_0 \leftarrow 0$

tant que $d_k < d$ **faire**

$j = 1$

tant que $j < D$ **faire**

$\mathcal{S}' \leftarrow \mathcal{S}_{d_k-1} \bigcup (x_{i,j})_{1 \leq i \leq l}$

Évaluer r_{d_k} sur $\mathcal{S}'$ en utilisant (VI.12)

$j \leftarrow j + 1$

fin tant que

$j_0 \leftarrow$ indice de l'attribut qui maximise r_{d_k}

$\mathcal{S}_{d_k} \leftarrow \mathcal{S}_{d_k-1} \bigcup (x_{i,j_0})_{1 \leq i \leq l}$

Orthogonaliser les colonnes de $\mathbf{X}$ par rapport à $[x_{1,j_0} \dots x_{l,j_0}]^T$

$d_k \leftarrow d_k + 1$

fin tant que

Sorties: $\mathcal{S}_d$ //Sélection de d attributs.

supports $\mathbf{x}_i$ du problème, faisant intervenir les multiplicateurs de Lagrange α_i :

$$\mathbf{w}^{pq} = \sum_{j=1}^{n_s} \alpha_j y_j \mathbf{x}_j. \tag{VI.13}$$

L'idée est que les attributs qui correspondent à des directions de l'espace selon lesquelles le vecteur $\mathbf{w}^{pq}$ admet une faible énergie, ne sont pas aussi utiles au problème que les autres attributs (puisqu'ils contribuent faiblement à la définition de l'hyperplan optimal).

A chaque récursion de l'algorithme SVM-RFE, l'attribut possédant le score le plus faible est éliminé. Le processus est arrêté lorsque le nombre d'attributs restant atteint d. Il est possible d'éliminer plus d'un attribut à la fois pour réduire la complexité de l'algorithme, qui est assez élevée ($O(l^2 D)$).

Nous proposons une description de cette approche sous forme d'algorithme, d'après [Guyon *et al.*, 2002] (voir Algorithme 4).

Algorithme 4 SVM-RFE pour un problème bi-classes.

Entrées: $\mathbf{X} \leftarrow [\mathbf{x}_1, \mathbf{x}_2, \dots \mathbf{x}_l]^T$ //Exemples d'apprentissage

$\mathbf{y} \leftarrow [y_1, y_2, \dots y_l]^T$ //Étiquettes de classe

d //Nombre d'attributs cible

//Initialisation

$\mathcal{S} \leftarrow \{1, 2, \dots D\}$ //Sous-ensemble des attributs "survivants"

$\mathcal{R} \leftarrow \{\ \}$ //Classement des attributs

tant que $s \neq \{\ \}$ **faire**

$\mathbf{X} \leftarrow \mathbf{X}(:, \mathcal{S})$ //Restreindre aux attributs utiles

$\boldsymbol{\alpha} \leftarrow$ SVM-train$(\mathbf{X}, \mathbf{y})$ //Apprentissage SVM linéaire

$\mathbf{w} \leftarrow \sum_k \alpha_k y_k \mathbf{x}_k$ //Calcul du vecteur de poids de dimension égale à card$(\mathcal{S})$

$f \leftarrow \arg\min\{w_i^2 \mid i \in \mathcal{S}\}$ //Trouver l'attribut ayant le score (w_i^2) le plus bas

$\mathcal{R} \leftarrow \mathcal{R} \bigcup \mathcal{S}(f)$ //Mise à jour du classement des attributs

$\mathcal{S} \leftarrow \mathcal{S} \setminus f$ //Suppression de l'attribut ayant le score le plus bas

fin tant que

Sorties: Classement des attributs $\mathcal{R}$

D. Algorithme MUTINF, basé sur l'information mutuelle

Soit Ω la variable aléatoire discrète associée aux classes Ω_q. L'entropie de Ω est définie par

$$H(\Omega) = -\sum_{q=1}^{Q} P(q) \log P(q), \tag{VI.14}$$

$P(q)$ étant la probabilité de l'observation $\Omega = \Omega_q$. $H(\Omega)$ peut être vue comme une mesure d'incertitude sur la valeur de Ω. On souhaite réduire cette incertitude en observant des attributs adéquats x_i qui sont modélisés par des variables aléatoires $\mathcal{X}_i$. L'information mutuelle définie par

$$I(\Omega, \mathcal{X}_i) = H(\Omega) - H(\Omega|\mathcal{X}_i), \tag{VI.15}$$

permet de mesurer *la réduction de l'incertitude sur* Ω *apportée par la connaissance de* $\mathcal{X}_i$.

Ainsi, l'idée de l'algorithme de sélection basé sur l'information mutuelle [Zaffalon et Hutter, 2002] est de choisir prioritairement les attributs x_i les plus informatifs (sur Ω), *i.e.* ceux qui réalisent les scores $w_i = I(\Omega, \mathcal{X}_i)$ les plus élevés.

$I(\Omega, \mathcal{X}_i)$ peut être obtenue selon

$$I(\Omega, \mathcal{X}_i) = \sum_{q,\, x_i} p(x_i, q) \log \frac{p(x_i, q)}{p(x_i) P(q)}, \tag{VI.16}$$

faisant intervenir la probabilité conjointe de q et x_i. En pratique des estimations empiriques des probabilités intervenant dans (VI.16) sont utilisées, ce qui représente l'inconvénient majeur de cette approche puisque la précision de l'estimation obtenue est fortement dépendante de l'échantillon utilisé, notamment de sa taille.

VI-5. Critères d'évaluation

Dans le but de comparer les performances des différentes approches de sélection, des critères de nature heuristique peuvent être exploités. Leur calcul est assez simple et ils permettent d'acquérir une première évaluation de l'efficacité des attributs produits par les algorithmes de sélection (les résultats de classification produits par chaque sélection restent bien sûr un critère privilégié). Nous utilisons deux critères : l'un permettant de se faire une idée du pouvoir de séparation des attributs sélectionnés, l'autre de la redondance de ces attributs, le but étant d'obtenir une sélection d'attributs qui soient à la fois non-redondants et qui présentent un fort pouvoir de séparation.

A. Critère de séparabilité des classes

Différents critères de séparabilité des classes peuvent être définis (*cf.* [Duda *et al.*, 2001, Theodoridis et Koutroumbas, 1998]). Nous choisissons d'utiliser un critère linéaire assez simple inspiré du principe de l'ALD [Mitra *et al.*, 2002]. Soit $\mathbf{\Sigma}_q$ la matrice de covariance obtenue à partir des l_q vecteurs d'attributs associés à la classe Ω_q définie par les éléments :

$$\mathbf{\Sigma}_q(i,j) = \frac{(\mathbf{x}_i^q - \boldsymbol{\mu}^q)^t(\mathbf{x}_j^q - \boldsymbol{\mu}^q)}{l_q - 1},\ 1 \leq i,j \leq d \tag{VI.17}$$

avec $\boldsymbol{\mu}^q = \frac{1}{l_q}\sum_{k=1}^{l_q}\mathbf{x}_k^q$. Soient $\pi_q = \frac{l_q}{l}$ (l'estimation de $P(\Omega = \Omega_q)$) et $\boldsymbol{\mu} = \frac{1}{l}\sum_{k=1}^{l}\mathbf{x}_k$ (l'estimation de la moyenne des observations). On pose

$$\mathbf{S}_w = \sum_{q=1}^{Q}\pi_q\mathbf{\Sigma}_q$$

et

$$\mathbf{S}_b = \sum_{q=1}^{Q}(\boldsymbol{\mu} - \boldsymbol{\mu}_q)(\boldsymbol{\mu} - \boldsymbol{\mu}_q)^t.$$

S_w est la matrice de dispersion intra-classe et S_b est la matrice de dispersion inter-classes [Mitra *et al.*, 2002]. Une bonne séparabilité des classes est obtenue avec une grande dispersion inter-classes (les points de classes différentes sont alors éloignés les uns des autres) et une petite dispersion intra-classe (le nuage de points relatif à une même classe est alors compact). La séparabilité S peut donc être définie à partir de $\mathbf{S}_w^{-1}\mathbf{S}_b$ [Mitra *et al.*, 2002], en prenant :

$$S = \mathrm{tr}(\mathbf{S}_w^{-1}\mathbf{S}_b). \tag{VI.18}$$

La trace permet d'obtenir une mesure scalaire robuste de la séparabilité. En d'autres termes, une valeur de séparabilité S élevée traduit un bon compromis entre distances intra-classes (à minimiser) et distances inter-classes (à maximiser).

B. Critère d'entropie de représentation

L'entropie de représentation H s'obtient en calculant les valeurs propres λ_i de la matrice de covariance des attributs de taille $D \times D$. Après normalisation de ces valeurs propres selon :

$$\tilde{\lambda_i} = \frac{\lambda_i}{\sum_{j=1}^{D}\lambda_j}, \tag{VI.19}$$

H se calcule selon :

$$H = -\sum_{j=1}^{D}\tilde{\lambda_j}\log\tilde{\lambda_j}. \tag{VI.20}$$

L'entropie est d'autant plus grande que la variance se répartit uniformément sur l'ensemble des attributs. De ce fait, H peut être vue comme une mesure de la redondance de l'ensemble d'attributs considéré. Par conséquent, on obtient un sous-ensemble d'attributs non-redondants en maximisant H [Mitra *et al.*, 2002].

VI-6. Comparaison du comportement des Algorithmes de Sélection d'Attributs

Nous proposons une étude comparative des performances des algorithmes de sélection décrits précédemment et de leur comportement vis-à-vis de pré-traitements tels que la normalisation, le sous-échantillonnage ou la quantification des données. Nous examinons également leur complexité et leur efficacité en relation avec les techniques de classification envisagées. Nous proposons ensuite des améliorations structurelles à ces algorithmes permettant d'atteindre de meilleures performances de classification.

La question de l'efficacité relative des différents descripteurs pour la tâche de classification des instruments de musique sera abordée au chapitre VIII.

Les expériences présentées ci-après ont été menées sur le corpus `SUB-INS` (décrit au chapitre II) comprenant huit classes d'instruments pour un total de 229543 exemples d'apprentissage et 270898 exemples de test. Afin d'expérimenter un grand nombre de variations sur les algorithmes dans des délais acceptables, nous allégeons la charge totale de calcul en considérant ici uniquement 162 attributs (D=162), parmi les 543 attributs dont nous disposons. Ceux-ci sont issus de 12 descripteurs différents. Dans un premier temps, nous réglons les ASA pour obtenir un sous-ensemble de d=40 variables en sortie[3]. Pour l'algorithme SVM-RFE, la valeur du paramètre C est fixée à 1000.

A. Influence de la taille de l'échantillon et de la normalisation

Nous effectuons deux types de sous-échantillonnage des données :

– *par tirage aléatoire* de 5000 exemples par classe (noté 8×5000 (RN)) ;

[3]Nous reviendrons ultérieurement sur le choix de d.

– *par Quantification Vectorielle (QV)* (LBG) en utilisant 1024 centroïdes par classe (noté 8×1024 (QV)).

Les algorithmes de sélection sont exécutés à la fois sur l'échantillon disponible dans son intégralité (noté ∞) et sur les deux sous-ensembles RN et QV, et ce avec et sans normalisation des données (normalisations "min-max" et "$\mu\sigma$", *cf.* section VI-6-A).

En raison de la complexité importante de l'algorithme SVM-RFE, celui-ci n'a pu être testé que sur les sous-échantillons de données (RN et QV)[4]. En outre, cette approche n'a pas produit de solution (dans un délai acceptable) en absence de normalisation.

1) Sorties des algorithmes de sélection

Dans un premier temps nous examinons la variation des attributs sélectionnés en sortie des algorithmes, en fonction de l'échantillon utilisé et de la normalisation. Les résultats sont synthétisés dans le tableau VI.1, dans lequel nous indiquons par un même symbole ("×","∗", etc.) le même sous-ensemble d'attributs sélectionnés.

Les remarques suivantes peuvent être faites concernant l'effet de la normalisation :

– l'algorithme Fisher n'est pas sensible à la normalisation des données, les mêmes attributs en sortie sont obtenus avec ou sans normalisation. Le fait de ne pas retrouver les mêmes sous-ensembles d'attributs par QV est plutôt dû à l'impact de la normalisation sur le processus de quantification. Le résultat est prévisible puisque la normalisation ne change pas la tendance du critère optimisé.
– La normalisation "min-max" ne modifie pas le résultat de la sélection IRMFSP effectuée sans normalisation. Par contre, la sortie est modifiée par la normalisation $\mu\sigma$. L'étape d'orthogonalisation intervenant dans cet algorithme fait qu'il ne se déroule pas avec la normalisation $\mu\sigma$ de la même façon qu'en absence de normalisation (ou avec la normalisation "min-max"), à cause de l'opération de soustraction des moyennes des attributs.
– Tous les autres algorithmes sont réactifs à la normalisation : les attributs sélectionnés varient sensiblement pour des normalisations différentes (les normalisations modifient les tendances des critères optimisés).

[4]Cet algorithme a été initialement proposé pour des problèmes dans lesquels $D > N$, ce qui n'est pas le cas ici.

Nb exemples	8×5000 (RN)			229543 (∞)			8×1024 (QV)		
Normalisation	-	min-max	$\mu\sigma$	-	min-max	$\mu\sigma$	-	min-max	$\mu\sigma$
Fisher	×	×	×	∗	∗	∗			×
IRMFSP	○	○		⋆	⋆				
MUTINF									
SVM-RFE	■			■	■	■	■		

Tab. VI.1 Impact de la normalisation et la taille de l'échantillon sur le résultat de la sélection d'attributs. "min-max" désigne le procédé de normalisation en amplitude et "$\mu\sigma$" la normalisation par rapport à la moyenne et l'écart-type (*cf.* section VI-2). Un même symbole ("×","∗", etc.) indique un même sous-ensemble d'attributs sélectionnés. Lorsqu'une case est vide, c'est que les attributs sélectionnés sont différents. Les calculs non-aboutis sont indiqués par des cases noires.

Par ailleurs, toutes les approches sont sensibles au sous-échantillonnage des données. Notons que l'approche Fisher semble la plus robuste, puisque les sous-ensembles d'attributs obtenus en utilisant le sous-échantillon 8×5000 (RN) ne diffèrent que de deux attributs (2/40) par rapport au sous-ensemble sélectionné en exploitant l'échantillon complet.

2) Performances des ASA relativement à la normalisation et l'échantillon

Afin de mesurer efficacement les performances des algorithmes de sélection considérés, nous exploitons les résultats de classification de 8 classes d'instruments par κ-NN, GMM, et SVM (*cf.* section V) parallèlement aux critères heuristiques proposés (*cf.* section VI-5). L'attention est ici portée sur les performances relatives des ASA, par conséquent nous exploitons des réglages "génériques" des classificateurs, permettant une faible complexité tout en évitant les problèmes de sur-apprentissage (*overfitting*)[5]. Ainsi :

- pour les κ-NN, le paramètre κ est choisi comme la racine carrée du nombre d'exemples d'apprentissage (κ=489) ;
- pour les GMM, nous utilisons M=8 composantes de mélange ; des valeurs plus élevées ne permettent pas forcément d'améliorer les performances) ;
- pour les SVM, nous exploitons un noyau linéaire et un paramètre de pénalité C adaptatif (réglé à partir des données selon (VII.1)).

[5]Nous reviendrons sur le réglage "optimal" des classificateurs au chapitre VII.

Le tableau VI.2 présente pour chaque ASA les normalisations et les échantillons de données produisant les "meilleures" valeurs des critères ainsi que celles qui sont jugées les moins satisfaisantes par ces critères.

Critère	Séparabilité (S)		Entropie (H)	
	Pire	Meilleur	Pire	Meilleur
PCA	RN, - 0.004	QV, - 0.006	QV, - 0.8	∞, $\mu\sigma$ <u>4.1</u>
Fisher	RN, (*) 0.045	QV, - <u>0.056</u>	QV, - 0.3	∞, (*) 2.5
IRMFSP	RN, (*) 0.038	QV, $\mu\sigma$ 0.049	QV, - 0.4	∞, (*) 2.9
MUTINF	∞, - 0.040	QV, - 0.053	∞, - 0.9	RN, $\mu\sigma$ 2.6
SVM-RFE	RN, mn-mx 0.036	QV, mn-mx 0.052	QV, $\mu\sigma$ 1.6	RN, $\mu\sigma$ 2.8

Tab. VI.2 Extrêma des critères heuristiques pour les différents ASA. Les colonnes "Meilleur" (respectivement, "Pire") présentent les cas les plus performants (respectivement, les moins performants) en indiquant la valeur des critères ainsi que la normalisation et l'échantillon utilisé par l'ASA (échantillon,normalisation). Le symbole (*) indique que toutes les configurations possibles produisent le même résultat.

Le tableau VI.3 présente pour chaque ASA associé à une normalisation et un échantillon de données d'apprentissage, les résultats de classification de l'échantillon de test `SUB-INS-T`. Ces résultats sont obtenus en moyennant sur les trois classificateurs les taux de bonne reconnaissance moyens obtenus pour les 8 classes d'instruments considérées. Notons que l'ensemble de test complet (∞) est utilisé pour l'apprentissage des classificateurs, indépendamment du sous-échantillon utilisé par les ASA (∞, RN ou QV), ce qui permet de mesurer l'influence de l'échantillon spécifiquement sur le comportement des algorithmes de sélection. Par ailleurs, l'effet de la normalisation sur les performances des ASA, en termes de taux de reconnaissance, doit être analysé avec prudence puisque nous utilisons, pour des raisons de simplicité, les mêmes normalisations pour la sélection des attributs et l'apprentissage des classificateurs. La normalisation peut alors avoir un double impact : sur les performances de l'algorithme de sélection et sur les performances de classification.

De plus, les résultats obtenus en utilisant une transformation par PCA vers un espace de même dimension $d = 40$ sont présentés afin de servir de référence.

Nb exemples	5×5000 (RN)			229543 (∞)			8×1024 (QV)		
Normalisation	-	min-max	$\mu\sigma$	-	min-max	$\mu\sigma$	-	min-max	$\mu\sigma$
PCA	43.9	**62.1**	59.7	44.2	**62.1**	60.5	43.8	**63.1**	58.7
Fisher	51.3	62.5	**64.4**	51.2	62.6	**64.7**	49.1	63.4	**63.9**
IRMFSP	45.3	61.4	**61.7**	37.1	62.9	**63.9**	47.5	57.6	**62.4**
MUTINF	61.9	63.2	**64.4**	57.9	61.2	62.2	61.6	63.3	**64.5**
SVM-RFE	-	**61.6**	**61.6**	-	-	-	-	**63.2**	**63.3**

Tab. VI.3 Performances des ASA et de la transformation par PCA en termes de taux de bonne reconnaissance moyens relativement à la normalisation et l'échantillon utilisés. 8 classes d'instruments, 40 attributs sélectionnés à partir de 162 possibles, 229543 exemples d'apprentissage et 270898 exemples de test. Pour chaque ASA, les meilleurs résultats (aux intervalles de confiance à 90% près : rayon $<$ 0.2%) par rapport à la normalisation sont présentés en gras. Les meilleurs résultats, toutes configurations confondues, sont soulignés.

A partir de ces deux tableaux nous observons que :

- dans tous les cas, les performances de classification obtenues sans normalisation sont nettement inférieures à celles obtenues avec l'une des deux normalisations : on constate plus de 20% d'amélioration dans certains cas (pour IRMFSP par exemple). Notons cependant que la normalisation a un impact plus important sur le processus de classification que sur la phase de sélection en soi puisque nous savons que pour les approches Fisher et IRMFSP, les mêmes attributs sont sélectionnés quelle que soit la normalisation (*cf.* section VI-6-A.1). Il apparaît que le critère de séparabilité S ne permet pas de traduire ce comportement de façon systématique puisqu'il privilégie dans tous les cas la sortie des ASA basés sur le sous-échantillon QV. Dans ce cas il semble que la normalisation a un impact plus important sur le processus de "clustering" et nous relevons des valeurs de S élevées avec des données non normalisées. Par contre, le critère d'entropie de représentation H reflète bien l'importance de la normalisation.
- La normalisation "$\mu\sigma$" donne lieu globalement aux meilleures performances avec la plupart des ASA (Fisher, IRMFSP, MUTINF, SVM-RFE), alors que la normalisation "min-max" semble mieux adaptée à la transformation par PCA, et elle est tout aussi efficace que la normalisation "$\mu\sigma$" avec SVM-RFE. En se rappelant que les deux normalisations "$\mu\sigma$" et

"min-max" produisent les mêmes attributs en sortie de Fisher et de IRMFSP, nous déduisons que la normalisation "$\mu\sigma$" est la plus adaptée au fonctionnement des classificateurs considérés (en moyenne). Nous reviendrons dans la suite sur le comportement de chaque classificateur en particulier vis à vis de la normalisation. Notons que le critère H sélectionne systématiquement la solution "$\mu\sigma$" quel que soit l'ASA.

– Les performances obtenues en effectuant la sélection sur les sous-échantillons sont globalement peu dégradées par rapport à celles atteintes en exploitant l'intégralité des données alors même que nous avions noté à la section VI-6-A.1 que les attributs en sortie variaient avec des échantillons différents. Cela indique, eu égard à la redondance des attributs de départ, que les ASA considérés présentent une certaine robustesse car les différents sous-ensembles sélectionnés à partir d'échantillons différents produisent des taux de reconnaissance comparables : il existe en fait différentes solutions d'attributs aux performances équivalentes.
– Par ailleurs, nous relevons que le sous-échantillonnage par QV est une alternative intéressante car elle permet d'atteindre des taux de reconnaissance parfois meilleurs qu'avec l'échantillon complet (avec PCA et MUTINF) tout en allégeant la complexité de la sélection (alors effectuée sur moins d'exemples). D'ailleurs le critère de séparabilité élit dans tous les cas la sortie des ASA basés sur le sous-échantillon QV. Il est raisonnable de penser que cela est dû à un effet de "dé-bruitage", c'est-à-dire de limitation de l'impact des exemples aberrants (*outliers*) sur le résultat d'un ASA, ce qui expliquerait aussi le fait que l'approche MUTINF se comporte mieux en utilisant les sous-échantillons.

B. Comparaison des performances des sélections

Nous nous intéressons maintenant aux performances comparées des ASA et de la transformation par PCA. Dans un premier temps nous évaluons ces performances en considérant les résultats de classification par κ-NN, GMM et SVM en moyenne, et en variant le nombre d'attributs sélectionnés d, ensuite nous les étudierons en rapport avec chaque classificateur.

1) Performances relatives des sélections

Nous observons, à partir des résultats du tableau VI.3, que les meilleures performances moyennes sont obtenues avec les algorithmes Fisher et MUTINF. Un examen des critères heuristiques révèle que ces deux algorithmes réalisent les valeurs de S les plus élevées, mais que les valeurs

d'entropie de représentation sont les plus faibles. Ainsi, de meilleures performances **moyennes** sont obtenues en privilégiant des attributs permettant une bonne séparabilité des classes, même si ceux-ci sont redondants entre eux. Nous verrons dans la suite que ce comportement varie en fonction des classificateurs.

Le tableau VI.4 donne les temps CPU relatifs au déroulement des différents ASA. L'approche Fisher s'avère nettement avantageuse car elle réalise un excellent compromis performances-complexité. Soulignons que l'algorithme SVM-RFE présente une complexité largement supérieure à celles des autres approches alors même qu'il n'exploite qu'un sous-échantillon des données d'apprentissage. De plus, sur les mêmes sous-échantillons, cette approche (la plus élaborée) ne fournit pas ici de meilleurs résultats que les approches les plus simples.

ASA	**Temps CPU**
Fisher	4.4s
IRMFSP	6mn 27s
MUTINF	9mn 51s
SVM-RFE	5j 7h 31mn 30s

Tab. VI.4 Complexité des ASA. Les algorithmes sont implémentés en Matlab (MUTINF et SVM-RFE sont disponibles dans la toolbox Spider [Spider,] qui reprend une implémentation en C des SVM [LibSVM,]). Les calculs ont été effectués sur des machines ayant 2.5GHz de CPU et 2Go de RAM. "j" : jour, "h" : heure, "mn" : minute, "s" : seconde. Sous-échantillon 8×5000 (RN) pour SVM-RFE, et échantillon complet pour les autres ASA.

Enfin, il est intéressant de noter que, de façon générale, de meilleurs résultats sont obtenus avec un ASA plutôt qu'avec une transformation par PCA. Comme nous l'avons signalé, la PCA exprime les attributs dans une base efficace pour la représentation des données et non pour la séparabilité des données de classes différentes.

2) Performances en relation avec la dimension cible

De nombreuses expériences préliminaires ont été menées pour déterminer un choix convenable de d. Nous avons observé que des améliorations significatives, en termes de taux de reconnaissance, sont obtenues en augmentant la valeur de d à partir de 20. Au delà de 40, le gain en performances devient peu significatif par rapport à la complexité. Nous retenons donc les valeurs $d = 20$ et $d = 40$ comme valeurs extrêmales. Il est évident qu'un réglage plus fin peut s'avérer

utile pour réaliser un bon compromis performances/complexité.

Nous donnons dans le tableau VI.5 les performances obtenues pour $d = 20$ attributs sélectionnés en comparaison avec celles correspondant aux sélections précédentes de $d = 40$ attributs (à partir de 162), toujours en moyenne sur les 3 classificateurs κ-NN, GMM et SVM avec les mêmes réglages.

Nous observons d'abord une dégradation générale des performances avec tous les ASA. Cela traduit le fait que le choix d=40 est un choix plus convenable pour notre schéma de classification. Au-delà de ce fait, nous remarquons, pour l'approche MUTINF une dégradation beaucoup plus nette des résultats (8% de baisse pour MUTINF contre moins de 2% de baisse en moyenne pour Fisher, IRMFSP et SVM-RFE). MUTINF s'avère beaucoup moins efficace pour une sélection avec un plus petit rapport $\frac{d}{D}$. Les 20 attributs classés en premier par MUTINF sont donc moins performants que ceux classés par les autres méthodes.

Au contraire, SVM-RFE exhibe la moins forte baisse de performances : en réduisant le nombre d'attributs sélectionnés de moitié, le taux de reconnaissance moyen chute de seulement 0.6%.

	d=**40**	d=**20**
PCA	60.5	58.5
Fisher	64.7	62.9
IRMFSP	63.9	61.9
MUTINF	62.2	56.5
SVM-RFE	61.6	61.0

Tab. VI.5 Taux de reconnaissance moyens (κ-NN,GMM et SVM) relatifs aux différentes sélections pour d=20. Normalisation $\mu\sigma$; sous-échantillon 8×5000 (RN) pour SVM-RFE, et échantillon complet pour les autres ASA.

3) Performances en relation avec les classificateurs

Le tableau VI.6 présente les résultats de classification obtenus pour chaque ASA (avec d=40), classificateur par classificateur.

D'abord, nous remarquons la supériorité du classificateur SVM indépendamment de l'ASA utilisé, ainsi que des performances optimales assez proches avec les κ-NN et les GMM (respectivement 63.5% et 63.2% en utilisant MUTINF).

Ensuite, nous notons clairement la mise en valeur des sélections IRMFSP et SVM-RFE par

Classificateur	κ-NN (κ=489)	GMM (M=8)	SVM (lin)
PCA	∞, mn-mx 62.1	QV, mn-mx 62.6	QV, mn-mx **64.5**
Fisher	∞, $\mu\sigma$ 62.7	∞, $\mu\sigma$ 62.5	∞, $\mu\sigma$ **68.8**
IRMFSP	∞, $\mu\sigma$ 63.1	∞, mn-mx 59.7	∞, $\mu\sigma$ **<u>69.2</u>**
MUTINF	QV, $\mu\sigma$ 63.5	RN, $\mu\sigma$ 63.2	QV, $\mu\sigma$ **66.8**
SVM-RFE	QV, $\mu\sigma$ 62.8	QV, mn-mx 61.1	RN, $\mu\sigma$ **67.4**

Tab. VI.6 Performances des différentes sélections en relation avec les classificateurs en utilisant la normalisation et l'échantillon donnant les meilleures performances (indiqués dans la première ligne de chaque cellule) et d=40. En gras : meilleur classificateur pour chaque ASA.

la classification SVM. En effet, les meilleurs résultats de classification sont obtenus avec l'ASA IRMFSP (69.2%) suivi par les ASA Fisher (68.8%) et SVM-RFE (67.4%) en association avec les SVM. En revanche, associées à la classification par GMM, les approches IRMFSP et SVM-RFE donnent les résultats les moins satisfaisants, alors qu'elles sont des plus performantes dans un schéma de classification par SVM. Nous mettons ici en évidence un lien entre la méthode de sélection et le classificateur utilisé.

En examinant les critères heuristiques (*cf.* dernières colonnes du tableau VI.7), on peut réaliser que IRMFSP et SVM-RFE présentent les valeurs d'entropie de représentation H parmi les plus élevées (significatives, nous le rappelons, d'un sous-ensemble d'attributs moins redondant). L'approche IRMFSP produit une sélection d'attributs présentant une même valeur de séparabilité S que l'approche Fisher (S=0.045) mais la première réalise une valeur de H plus grande (grâce à la phase d'orthogonalisation intervenant dans l'algorithme). Il en est de même pour les ASA MUTINF et SVM-RFE : SVM-RFE réalise un meilleur compromis séparabilité-entropie. La classification par SVM semble la mieux à même d'exploiter un tel compromis, si bien que les approches IRMFSP et SVM-RFE se retrouvent dans le "trio de tête" (avec l'approche Fisher) dans un schéma de classification par SVM.

VI-7. Variations sur les Algorithmes de Sélection des Attributs

A. Un nouvel algorithme de sélection : Fisher-based Selection of Feature Clusters (FSFC)

Eu égard aux bonnes performances de l'approche Fisher, nous nous sommes attachés à l'améliorer de manière à prendre en compte la contrainte de non-redondance des attributs sélectionnés. L'idée du nouvel algorithme s'inspire du fonctionnement des algorithmes décrits dans [Campedel et Moulines, 2005] et [Mitra *et al.*, 2002].

Dans sa version la plus simple, FSFC se déroule en deux temps :

- dans un premier temps, nous effectuons un "clustering" des différents attributs (toutes classes confondues) afin de composer une organisation de ces attributs dans laquelle ceux qui présentent des distributions de valeurs similaires se retrouvent dans les mêmes clusters ;
- ensuite, nous effectuons dans chaque cluster une sélection de d_c attributs à l'aide de l'ASA Fisher.

Du fait que le sous-ensemble d'attributs résultant se compose d'éléments issus de différents clusters, on peut s'attendre à ce que la contrainte de non-redondance soit mieux respectée, puisque les attributs regroupés dans les mêmes clusters sont potentiellement redondants.

En l'absence d'attributs "bruités" et pour une répartition uniforme de la redondance par groupes de variables, on peut se contenter de représenter les attributs par d clusters pour ensuite sélectionner (au moyen de l'algorithme Fisher) $d_c = 1$ attribut dans chaque cluster, obtenant ainsi les d attributs recherchés. Cependant, dans nombre de situations, en l'occurrence dans le contexte de descripteurs audio, il est nécessaire de prendre en compte les deux faits suivants :

- un certain nombre d'attributs ne sont pas toujours pertinents et peuvent être considérés comme bruités, par suite si d_c attributs sont systématiquement sélectionnés dans chaque cluster, il suffit qu'il y ait des clusters de bruit pour que des attributs bruités soient prélevés ;
- il peut y avoir, parmi les attributs pertinents, des clusters d'attributs redondants de tailles très différentes : typiquement on peut retrouver quelques clusters comprenant une dizaine d'attributs et d'autres en regroupant plus d'une cinquantaine. Il en résulte que pour ne pas négliger la contrainte de séparabilité, nous avons intérêt à prélever un plus grand nombre d'attributs à partir des clusters de plus grande taille.

Par conséquent, une version plus élaborée de l'algorithme qui essaie de prendre en compte ces

deux points est proposée :

- d'abord, nous effectuons le clustering en nous fixant un nombre de clusters M_c supérieur au nombre d d'attributs attendus en sortie ;
- ensuite, nous sélectionnons, dans chaque cluster C_i (par Fisher), un nombre d_{c_i} d'attributs dépendant de la taille du cluster, pour obtenir $D_c = \sum_{i=1}^{M_c} d_{c_i}$ attributs sélectionnés ($d < D_c < D$) ;
- enfin nous gardons d attributs parmi les D_c ainsi sélectionnés en employant encore l'approche Fisher.

En fait, le choix du nombre de clusters M_c permet de contrôler les valeurs de S et H du sous-ensemble d'attributs sélectionné. Une valeur M_c élevée (à la limite $M_c = D$) permet de préserver une valeur de S élevée (obtenue par Fisher simple), par contre un faible nombre de clusters fait augmenter l'entropie de représentation H puisqu'on diminue la redondance, mais au détriment de la séparabilité (surtout pour un mauvais choix des d_{c_i}).

En pratique, nous choisissons le nombre de clusters de manière à obtenir le meilleur compromis S et H. Le choix $M_c = \frac{3}{2}d$ s'est avéré fournir des résultats satisfaisants, dans des tests préliminaires réalisés sur le corpus de développement. Pour ce qui est du choix de d_{c_i}, nous choisissons $d_{c_i} = \left\lceil \frac{\mathrm{card}(\mathrm{C}_i)}{d} \right\rceil$ attributs par cluster.

Pour le clustering, nous exploitons l'algorithme agglomératif décrit dans la section V-3. Deux critères de proximité ont été envisagés : la distance de Bhattacharryya et la divergence (*cf.* section V-3-B), en faisant ici l'hypothèse de "gaussianité" des densités de probabilité régissant les distributions des attributs. Cette hypothèse de "gaussianité" permet de simplifier les calculs (autrement assez complexes, eu égard au nombre élevé d'attributs de départ : 162 dans ces expériences, et 543 au total). De plus, en vertu du théorème de la limite centrale [Duda *et al.*, 2001], elle est peu pénalisante étant donné le nombre élevé d'observations pour chaque attribut (provenant de toutes les classes d'instruments). D'ailleurs nous exploitons directement la moyenne et la covariance empiriques des attributs dans le calcul des distances.

Nous avons ainsi effectué le clustering des 162 attributs considérés en visant 60 clusters et en utilisant la distance de Bhattacharryya et la divergence. Un coefficient de corrélation cophénétique plus élevé a été trouvé avec la distance de Bhattacharryya (0.81), donc un meilleur clustering. Ensuite, un attribut a été sélectionné par cluster en utilisant l'ASA Fisher. Enfin, 40 attributs parmi les 60 ainsi trouvés ont été retenus (par Fisher également).

Algorithme 5 FSFC

Entrées: $\mathbf{X} \leftarrow [\mathbf{x}_1^T, \dots \mathbf{x}_o^T, \dots \mathbf{x}_l^T]^T$ //Exemples d'apprentissage

d, M_c //Nombre d'attributs à sélectionner et nombre de clusters

//Initialisation

$\mathcal{S} \leftarrow \{\}$ //Sous-ensemble des attributs sélectionnés

C_i //Clusters obtenus par clustering des attributs en M_c clusters

$i \leftarrow 1$

$\mathcal{S}' = \{\ \}$

tant que $i < M_c$ **faire**

$\mathcal{S}_i \leftarrow$ Sélection de $d_{c_i} = \left\lceil \frac{\mathrm{card}(\mathrm{C}_i)}{d} \right\rceil$ attributs à partir du cluster C_i par Fisher

$\mathcal{S}' \leftarrow \mathcal{S}' \bigcup \mathcal{S}_i$

$i \leftarrow i + 1$

fin tant que

$\mathcal{S} \leftarrow$ Sélection de d attributs à partir de $\mathcal{S}'$ par Fisher

Sorties: $\mathcal{S}$ //Sélection de d attributs.

Méthode	κ-NN (κ=489)	GMM (M=8)	SVM (lin)	Moy	S	H
FSFC	64.0	63.6	69.1	65.6	0.044	2.7
Fisher	62.7	62.5	68.8	64.7	0.045	2.5
IRMFSP	63.1	59.4	69.2	63.9	0.045	2.9
MUTINF	61.2	60.3	65.2	62.2	0.040	2.5
SVM-RFE	60.3	57.3	67.4	61.7	0.040	2.8

Tab. VI.7 Performances des différentes sélections comparées à celles de FSFC.

Nous présentons dans le tableau VI.7 les résultats pour la même tâche de classification que précédemment, obtenus avec la nouvelle approche FSFC en comparaison avec ceux obtenus en utilisant les autres ASA. Nous indiquons également les valeurs de séparabilité et d'entropie de représentation.

L'approche proposée réalise les meilleures performances moyennes en terme de taux de reconnaissance. Avec le classificateur SVM, FSFC est au même niveau que IRMFSP (aux intervalles de confiances près). FSFC est supérieur à tous les autres ASA avec les deux autres classificateurs κ-NN et GMM (cela n'est pas reflété par les valeurs de H et S correspondant à FSFC et IRMFSP). Par rapport à l'approche Fisher, nous observons un meilleur compromis séparabilité et entropie de représentation avec FSFC.

B. Sélection binaire

L'approche que nous développons consiste à effectuer une *sélection binaire* des attributs en ce sens que nous recherchons un sous-ensemble "optimal" d'attributs différent pour la discrimination de chaque paire de classes possible, dans la perspective d'un schéma de classification "un contre un". En d'autres termes, nous sélectionnons $\mathrm{C}_Q^2 = \frac{Q(Q-1)}{2}$ sous-ensembles d'attributs[6] $\{\mathcal{S}_{p,q}\}_{1\leq p<q\leq Q}$ (pour les Q classes considérées), avec $\mathcal{S}_{p,q}$ le sous-ensemble d'attributs optimal pour la discrimination de la paire de classes $\{\Omega_p,\Omega_q\}$. La figure VI.1 présente une vue d'ensemble du processus de sélection binaire.

Nous allons mettre en évidence que cette approche est non seulement plus efficace que l'approche classique en termes de résultats de classification, mais qu'elle présente en plus des avantages d'un point de vue analytique. En effet, elle permet de dégager de façon plus aisée et plus systématique des voies d'amélioration du schéma de classification, par l'utilisation de la matrice de confusions entre classes dans l'élaboration de systèmes plus performants. Par exemple, si de faibles taux de reconnaissance sont obtenus pour une classe Ω_p à cause de nombreuses confusions avec la classe Ω_q, il devient possible de focaliser l'attention uniquement sur la paire $\{\Omega_p, \Omega_q\}$ pour produire un sous-ensemble d'attributs mieux à même de discriminer spécifiquement ces deux classes.

[6] C_p^n dénote le nombre de combinaisons de n parmi p.

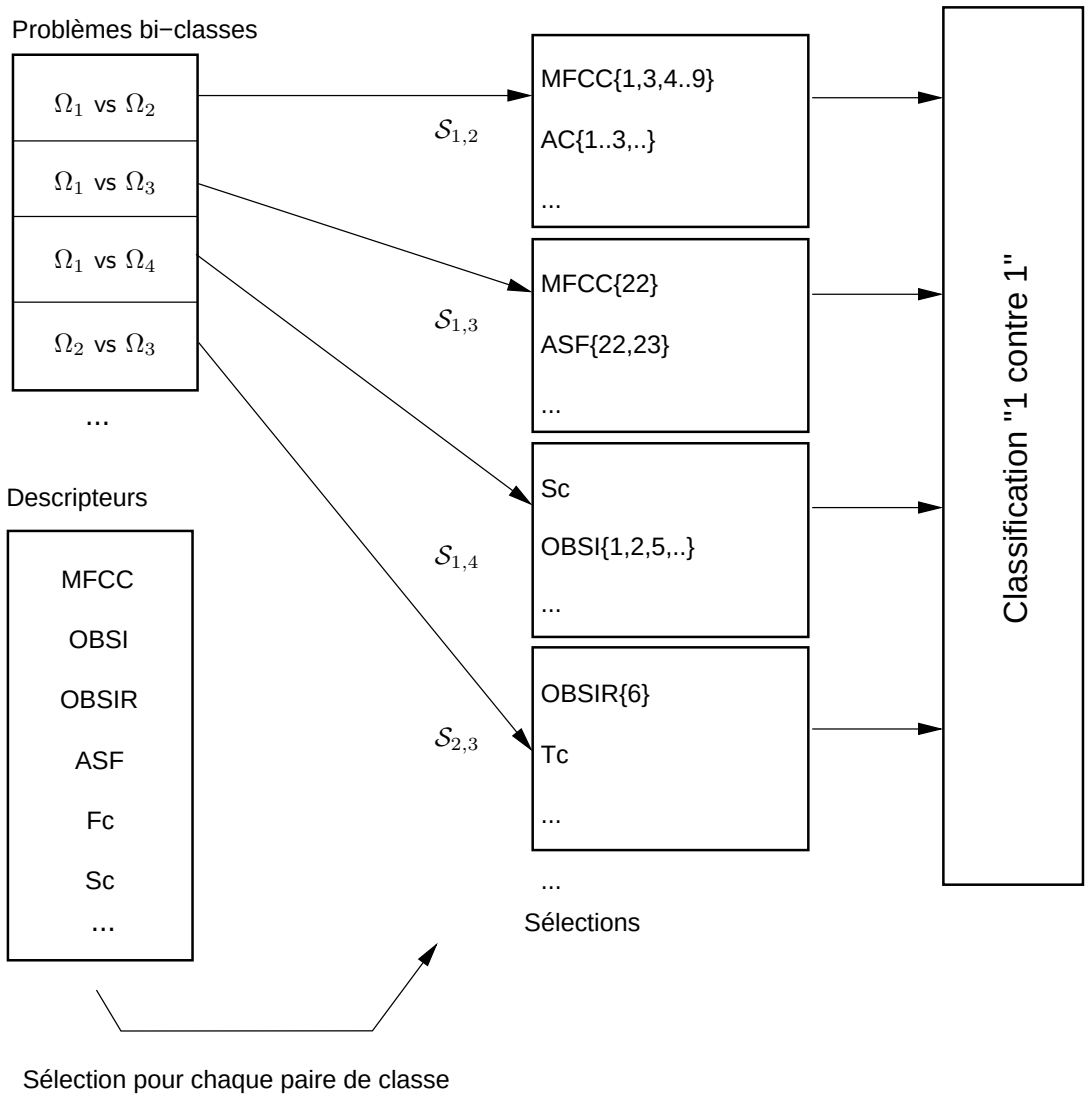

Fig. VI.1 Principe de sélection binaire des attributs.

De plus, la sélection binaire permet d'acquérir une meilleure connaissance du problème de classification et d'en dégager de l'information de haut-niveau. Dans le cas des instruments de musique, une meilleure compréhension des différences de timbre est ainsi gagnée sous forme d'interprétations du type "*l'instrument* Ω_p *ne présente pas les mêmes caractéristiques A et B que l'instrument* Ω_q", où les caractéristiques "*A et B*" sont déduites à partir du sous-ensemble d'attributs spécifiquement sélectionné pour la paire $\{\Omega_p, \Omega_q\}$. Par exemple, le fait de retrouver des caractéristiques de modulation d'amplitude dans les attributs sélectionnés pour les deux instruments indique que le trémolo est une caractéristique permettant de les différencier.

Dans la suite nous notons "C_Q^2-X" tout schéma de sélection binaire exploitant l'ASA "X", et "1-X" le schéma de sélection standard (exploitant toutes les classes à la fois et produisant une même sélection d'attributs).

Nous allons étudier l'apport de la sélection binaire en rapport avec le nombre d'attributs à sélectionner d, nous verrons que le choix de d a un impact important sur les performances de cette méthode. Nous commençons par considérer le cas d=20 puis nous traiterons le cas $d = 40$.

Sélection binaire visant d=20 attributs en sortie Nous présentons dans le tableau VI.8 les résultats obtenus pour la même tâche de classification, en sélectionnant les attributs de façon binaire, comparés à ceux obtenus avec une sélection "standard", avec d=20 attributs dans les deux cas. Nous remarquons que :

- l'approche binaire est globalement plus performante quel que soit l'ASA utilisé, nous obtenons en moyenne de meilleurs taux de reconnaissance indépendamment du classificateur utilisé qu'il soit par essence un classificateur binaire (SVM) ou non-binaire (κ-NN et GMM) ;
- dans de nombreux cas, le gain est assez important : +8% en moyenne avec MUTINF, +4% avec la PCA et +3% avec Fisher ;
- les rares cas où la sélection 1-X génère de meilleurs résultats que la sélection C_8^2-X concernent les configurations associant les κ-NN aux ASA IRMFSP et SVM-RFE, sans doute à cause d'un réglage assez grossier de la valeur de κ qui a été fixée à la valeur moyenne de $\sqrt{l_p + l_q}$, $1 \leq p < q \leq 8$, avec l_p le nombre d'exemples d'apprentissage de la classe Ω_p ;
- C_8^2-SVM-RFE, en association avec le classificateur SVM et avec d=20, produit des résultats (68.1%) se rapprochant des meilleurs résultats obtenus précédemment avec les sélections 1-X ciblant d=40.

Méthode	κ-NN ($\kappa = 489$)	κ-NN ($\kappa = 223$)	GMM (M=8)		SVM (lin)		Moyenne	
	8-class	binaire	8-class	binaire	8-class	binaire	8-class	binaire
PCA	58.9	61.7	55.7	61.2	60.9	64.9	58.5	62.6
Fisher	62.3	64.1	61.2	64.8	65.2	66.6	62.9	65.2
FSFC	61.7	63.0	59.8	63.6	65.7	65.7	62.4	64.1
IRMFSP	60.8	59.4	60.4	61.8	64.5	65.2	61.9	62.1
MUTINF	56.8	62.7	57.4	65.1	55.3	65.2	56.5	64.3
SVM-RFE	60.7	57.4	58.1	59.5	64.4	68.1	61.0	61.7

Tab. VI.8 Résultats de classification avec l'approche de sélection binaire, comparés à ceux obtenus avec l'approche classique avec $d = 20$.

Ainsi avec une dimension réduite de moitié, nous parvenons grâce à la configuration binaire à atteindre des performances comparables. Cela paraît très avantageux du point de vue de la réduction de la complexité de la classification.

Cependant, la complexité d'extraction des attributs devient plus élevée à l'étape de test. En effet, dans le cas binaire, le nombre total d'attributs devant être extraits à partir des signaux de test, correspond à la réunion des ensembles $\{\mathcal{S}_{p,q}\}$, dont le cardinal est généralement supérieur à d, comme on peut le voir dans le tableau VI.9. Cela peut être contraignant si l'on ne tolère pas l'extraction de plus de d attributs.

Méthode	$\text{card}\left(\bigcup \mathcal{S}_{p,q}\right)$
Fisher	82
FSFC	77
IRMFSP	99
MUTINF	92
SVM-RFE	73

Tab. VI.9 Nombre total d'attributs devant être extraits pour toutes les paires de classe avec la sélection binaire, dans le cas d=20.

Le tableau VI.10 présente le détail des taux de reconnaissance en sélectionnant $d = 20$ attributs avec 1-SVM-RFE et C_8^2-SVM-RFE, et en utilisant les SVM. L'amélioration moyenne des résultats est de +4%. De plus, nous observons d'importantes améliorations dans la reconnaissance de certaines classes : +11.7% pour Cl et +10.7% pour Co, par exemple.

% correcte	8-class	binaire
Pn	77.6	79.6
Gt	52.1	52.2
Ob	81.6	82.2
Cl	42.9	54.6
Fh	62.2	65.8
Tr	72.2	71.1
Co	61.0	71.7
Vl	65.3	68.0
Moyenne	64.4	68.1
Ecart-type	12.9	10.6

Tab. VI.10 Résultats de classification SVM avec 1-SVM-RFE et C_8^2-SVM-RFE, d=20.

Sélection binaire visant d=40 attributs en sortie Nous refaisons maintenant les mêmes expériences en sélectionnant $d = 40$ attributs pour chaque problème bi-classes. Les résultats correspondant sont présentés dans le tableau VI.11. Nous ne retrouvons pas toujours les mêmes améliorations que précédemment avec l'approche binaire. Si celle-ci reste plus performante dans la plupart des cas en utilisant le classificateur GMM, notamment avec la PCA et l'ASA Fisher, nous relevons de nombreuses configurations où la sélection 1-X est plus performante.

Méthode	κ-NN (489)		κ-NN (223)		GMM (M=8)		SVM (lin)		Moyenne	
	8-class		binaire		8-class	binaire	8-class	binaire	8-class	binaire
PCA	60.5		61.6		57.0	59.2	64.0	65.9	60.5	62.2
Fisher	62.7		63.6		62.5	64.8	68.8	66.8	64.7	65.1
FSFC	64.0		63.2		63.6	64.7	69.1	68.2	65.6	65.4
IRMFSP	63.1		60.1		59.4	59.2	69.2	65.8	63.9	61.7
MUTINF	61.2		63.7		60.3	63.6	65.2	66.1	62.2	64.5
SVM-RFE	60.3		58.5		57.3	58.5	67.4	67.4	61.6	61.5

Tab. VI.11 Résultats de classification avec l'approche de sélection binaire comparés à ceux obtenus avec l'approche classique avec $d = 40$.

Ainsi, la sélection binaire nécessite qu'un soin particulier soit apporté au choix du nombre d'attributs à sélectionner pour chaque paire de classes. En effet, certains problèmes bi-classes demandent l'utilisation d'un nombre d'attributs plus petit que celui nécessaire à la discrimination

de toutes les classes globalement. Le fait d'utiliser un nombre d'attributs élevé pour ces paires de classes a pour effet d'introduire des variables inutiles ou bruitées qui ne servent pas leur discrimination. Par conséquent, la valeur de d doit être adaptée pour chaque problème bi-classes pour atteindre des performances optimales.

L'approche binaire présente l'avantage de permettre l'adaptation des attributs à un contexte particulier de classification, c'est à dire qu'elle permet d'optimiser des problèmes de classification complexes en concentrant les efforts sur les sous-problèmes bi-classes demandant le plus d'attention. Concrètement, cela peut se faire en concevant des descripteurs dédiés spécifiquement à la discrimination de deux classes particulières (ce qui est plus simple que de concevoir des attributs permettant la séparation de toutes les classes en même temps). Nous pensons que la sélection binaire possède, de ce fait, un grand potentiel.

Afin d'illustrer cette aptitude de l'approche binaire, nous réalisons l'expérience suivante. Soient $\mathcal{E}_1^{40}$ l'ensemble de $d = 40$ attributs sélectionnés par l'algorithme 1-IRMFSP, et $\mathcal{E}_{p,q}^{20}$ les ensembles de $d = 20$ attributs sélectionnés par C_8^2-SVM-RFE pour les 28 paires de classes $\{\Omega_p, \Omega_q\}$, $1 \leq p < q \leq 8$. Nous voulons obtenir des sélections d'attributs $\bar{\mathcal{E}}_{p,q}$ plus performantes que $\mathcal{E}_{p,q}^{20}$. Nous mesurons alors, pour chaque paire, la séparabilité $S_{p,q}$ obtenue avec $\mathcal{E}_{p,q}^{20}$ (*cf.* section VI-5-A). Ensuite,

- si $S_{p,q} < 0.02$, nous prenons $\bar{\mathcal{E}}_{p,q}$=$\mathcal{E}_1^{40}$;
- sinon, nous prenons $\bar{\mathcal{E}}_{p,q}$=$\mathcal{E}_{p,q}^{20}$.

Douze ensembles $\mathcal{E}_{p,q}^{20}$ (sur les 28 possibles) ont présenté des valeurs de séparabilité $S_{p,q} < 0.02$ et ils ont donc été remplacés par $\mathcal{E}_1^{40}$. La dimension moyenne des problèmes bi-classes (Ω_p vs Ω_q) est alors

$$\bar{d} = \frac{1}{28} \sum_{1 \leq p < q \leq 8} d_{p,q} = 29,$$

avec $d_{p,q}$ le nombre d'attributs correspondant à la paire $\{\Omega_p, \Omega_q\}$ (parmi les deux valeurs possibles $d = 20$ ou $d = 40$).

Les résultats de classification par SVM, en se basant sur ces nouvelles sélections, sont donnés dans le tableau VI.12, comparés à ceux trouvés avec 1-IRMFSP(d=40) et C_8^2-SVM-RFE(d=20). Les sélections $\bar{\mathcal{E}}_{p,q}$ (colonne "OPT") produisent les mêmes performances que la sélection 1-IRMFSP(d=40) à la différence que dans le premier cas, 16 classificateurs SVM sur 28 opèrent dans des espaces de dimension 20, alors que dans le deuxième cas tous les classificateurs SVM

opèrent en dimension 40, d'où une importante réduction de la complexité. Nous nous gardons de plus la possibilité de concevoir des descripteurs ciblant spécifiquement la discrimination de certaines paires de classes pour lesquelles les confusions sont importantes.

Méthode	IRMFSP(40)	C_8^2-SVM-RFE(20)	OPT
Pn	85.3	79.6	**85.5**
Gt	61.7	52.2	58.4
Ob	82.8	82.2	**83.5**
Cl	63.9	54.6	62.6
Fh	58.8	65.8	**62.2**
Tr	71.3	71.1	**71.9**
Co	53.9	71.7	**54.7**
Vl	76.0	68.0	**75.8**
Moyenne	69.2	68.1	**69.3**
Ecart type	11.5	10.6	11.6
Dim. moy.	40	20	28

Tab. VI.12 Optimisation de la sélection C_8^2-SVM-RFE(d=20) par "hybridation" avec la sélection 1-IRMFSP(d=40).

VI-8. Conclusions sur la sélection des attributs

Nous avons étudié un certain nombre d'algorithmes de sélection automatique des attributs pour la tâche de la reconnaissance des instruments de musique.

Dans un premier temps, nous nous sommes intéressés à l'influence de la normalisation des données et de leur échantillonnage sur la sortie de ces algorithmes et nous avons observé que :

- la normalisation est une étape importante dans la construction du schéma de classification : même si elle ne modifie pas dans tous les cas la sortie des ASA, elle est nécessaire au bon fonctionnement des classificateurs ;
- la normalisation la mieux appropriée pour la sélection des attributs est celle qui produit des données centrées et de variance unitaire (nous l'avons désignée par $\mu\sigma$) ; celle-ci n'est en revanche pas adaptée à la transformation par PCA, la normalisation "min-max" qui ramène la dynamique des données dans l'intervalle [-1, +1] est dans ce cas-ci plus adaptée) ;

– il peut être intéressant de réaliser un sous-échantillonnage des données préalablement à l'étape de sélection pour limiter l'effet d'observations aberrantes et réduire la complexité du traitement.

Nous nous sommes ensuite attachés à comparer les performances des algorithmes de sélection étudiés en relation avec les pré-traitements réalisés sur les données, le nombre d'attributs sélectionnés et les classificateurs utilisés. Nous retenons que :

– l'utilisation d'un ASA permet d'atteindre de meilleurs résultats de classification que ceux obtenus en ayant recours à une transformation par PCA des données ;
– les approches de sélection les plus simples (Fisher) permettent d'atteindre des performances comparables, sinon supérieures à celles permises par les approches les plus élaborées, notamment l'approche SVM-RFE qui pêche par une complexité très élevée, rendant très difficile son application à des bases d'apprentissage de dimensions typiques de l'indexation audio ;
– certains algorithmes sont moins sensibles que d'autres à l'effet de la réduction du nombre d'attributs sélectionnés, en particulier SVM-RFE, ce qui indique qu'ils réalisent un classement plus fiable des attributs les plus utiles en positionnant les plus efficaces aux premiers rangs ;
– la prise en compte de la contrainte de non-redondance des attributs sélectionnés permet une amélioration significative des taux de reconnaissance en association avec le classificateur SVM qui est plus à même de tirer profit de cette qualité, si bien que les approches qui intègrent la contrainte de non-redondance (ici IRMFSP) réalisent les meilleurs résultats de classification.

Dans un deuxième temps, nous avons proposé de nouveaux schémas de sélection des attributs qui s'avèrent intéressants du point de vue des performances mais également d'un point de vue analytique. En effet :

– notre algorithme de sélection FSFC donne lieu des résultats de classification en moyenne supérieurs à ceux obtenus avec les autres ASA considérés, tout en produisant une taxonomie des attributs, dans laquelle ceux qui présentent des distributions de valeurs similaires sont regroupés dans les mêmes clusters (nous reviendrons sur cette organisation des attributs dans le chapitre VIII) ;
– notre approche de sélection binaire permet d'atteindre de bonnes performances de classi-

fication avec des sélections d'attributs de tailles plus petite en moyenne, mais elle permet surtout de comprendre les différences de caractéristiques des classes prises par paires et offre la possibilité de concentrer l'effort de conception sur des sous-problèmes (de nature plus simple) qui méritent le plus d'attention. Cette approche nécessite cependant l'adaptation du nombre d'attributs sélectionnés d à chaque problème bi-classes. De plus, elle suppose, qu'à l'étape de test, il soit toléré d'extraire un nombre total d'attributs plus important que dans la configuration de sélection standard, puisque c'est le sous-ensemble d'attributs qui correspond à la réunion de tous les sous-ensembles $\mathcal{S}_{p,q}$, $1 \leq p < q \leq Q$, qui doit être extrait à partir du signal de test.

Nous retenons l'approche FSFC pour la suite du développement. Nous ne ferons appel à l'approche de sélection binaire que pour le système final. Pour des raisons de simplicité, FSFC sera donc utilisé dans les études à suivre dans une configuration standard (non-binaire) et nous sélectionnerons $d = 40$ attributs. Nous ferons à nouveau varier d dans la mise en œuvre du système final.

VII. Etude expérimentale préliminaire de la classification par SVM

Nous nous intéressons dans ce chapitre à l'optimisation de la classification par SVM. Nous examinons différents noyaux et des critères permettant un réglage optimal des paramètres des SVM à partir de l'ensemble d'apprentissage (sans recours à une étape de test).

Par ailleurs, nous introduisons l'utilisation de fenêtres de décision en temps plus longues, permettant d'obtenir de meilleurs taux de reconnaissance.

VII-1. Introduction

Nous avons pu mettre en évidence grâce aux expériences menées sur la sélection d'attributs, la supériorité du classificateur SVM (dans sa configuration linéaire, la plus simple) par rapport aux autres classificateurs considérés (GMM et κ-NN). Des expériences complémentaires montrent que même en optimisant le nombre de composantes du mélange gaussien (en faisant varier M dans l'ensemble $\{8, 16, 32, 64, 128, 256, 512\}$), les performances de classification par GMM restent en-dessous de celles des SVM linéaires sur la tâche de classification des instruments de musique.

Par conséquent, notre choix s'est porté sur le classificateur SVM dans la construction de notre système de classification, car en plus de ces performances, il nous paraît plus prometteur et en tout cas mieux justifié d'un point de vue théorique puisqu'il ne fait pas d'hypothèses approximatives sur la forme des densités de probabilité des données.

Un système de classification par SVM nécessite le réglage de paramètres tels que le paramètre de pénalisation C ou le choix d'un type de noyau et de sa paramétrisation. Ces paramètres sont typiquement réglés au moyen d'une procédure de validation croisée. Cette procédure nécessite

l'exécution de plusieurs instances d'apprentissage et de test en explorant un ensemble de valeurs possibles des différents paramètres, pour retenir celles donnant les meilleurs résultats de classification (en moyenne sur les différentes instances de test).

Une alternative moins coûteuse et plus avantageuse dans les situations où les données à disposition sont limitées, consiste à trouver les paramètres optimaux à partir du même et seul ensemble d'apprentissage en exploitant des critères qui tentent de prédire le comportement du classificateur en généralisation (c'est-à-dire sur de nouveaux exemples de test). Deux critères de ce type ont été introduits dans la section V-2-E : l'estimation $\xi\alpha$ du risque réel (l'erreur en généralisation) [Joachims, 2000] donnée par la formule (V.65) et la borne VC donnée par (V.17).

Ces deux caratéristiques ont été présentées dans des travaux antérieurs comme efficaces pour l'obtention de bons réglages des classificateurs SVM. Joachims propose de minimiser l'erreur $\xi\alpha$ (sur un ensemble de valeurs possibles des paramètres) pour déterminer les valeurs adéquates du paramètre C et le choix du noyau [Joachims, 2000]. Schölkopf *et al.* préconisent d'élire pour cela le jeu de paramètres réalisant la plus petite dimension VC (notée h) afin d'obtenir la borne la plus petite sur le risque [Schölkopf *et al.*, 1995].

Nous envisageons dans la suite l'utilisation de ces deux critères pour le réglage des SVM en comparant leur efficacité.

VII-2. Paramètres d'optimisation du calcul des SVM

Comme décrit dans la section V-2-C, l'apprentissage des SVM sur des ensembles de taille importante impose le recours à des techniques de décomposition du problème d'optimisation en sous-problèmes de taille plus petite. La taille θ des sous-ensembles de travail dans ce processus de décomposition doit être réglée (*cf.* section V-2-C).

Les tests que nous avons effectués (en variant θ dans l'ensemble $\{2, 10, 20, 40, 200\}$) indiquent que le choix de θ a un impact pratiquement nul sur les performances, mais qu'il influence fortement la durée de l'optimisation. Les valeurs trop grandes (θ=200) ou trop petites (θ=2) causent des délais de calculs plus importants. Le choix $\theta = 20$ s'est avéré convenable du point de vue du temps de calcul.

VII-3. Choix du paramètre C

Nous commençons par étudier le comportement des SVM vis à vis du paramètre de pénalisation C (*cf.* section V-2-B). De nombreux tests préliminaires ont été effectués en utilisant les trois noyaux, linéaire, gaussien et polynômial, et des valeurs de C que nous avons, dans un premier temps, fait varier par puissances de 10, en prenant $C = \{1, 10, 100, 1000, 10000\}$. Cela nous a conduit à restreindre l'étude au noyau linéaire et à des valeurs de C dans l'ensemble $\{1, 10, 20\}$. Les mêmes comportements que ceux qui sont décrits ici sont retrouvés pour des valeurs plus grandes de C et d'autres noyaux.

Une valeur de C, notée C_{dat}, fixée de façon adaptive à partir des exemples d'apprentissage a également été envisagée : elle est obtenue comme l'inverse de la longueur moyenne des l exemples d'apprentissage transformés $\Phi(\mathbf{x}_i)$ [Joachims,], en prenant

$$C_{dat} = \frac{1}{\frac{1}{l}\sum_{i=1}^{l} k(\mathbf{x}_i, \mathbf{x}_i)}. \tag{VII.1}$$

Des expériences préliminaires sont effectuées sur les trois classes Pn, Gt et Ob à partir des données `SUB-INS` (*cf.* chapitre II). Trois machines à vecteurs supports sont ainsi apprises pour les paires Pn/Gt, Pn/Ob et Gt/Ob, pour chaque valeur de C considérée et nous obtenons dans chaque cas :

- une estimation de la dimension VC : h (*cf.* section V-2-E.1) ;
- une estimation de l'erreur en généralisation : l'erreur $\xi\alpha$ (*cf.* section V-2-E.2) ;
- le nombre de vecteurs supports bornés (BSV) et le nombre total de vecteurs supports (SV) ;
- le nombre d'erreurs sur l'ensemble d'apprentissage (nb err. app.) ;
- le temps CPU (nous utilisons une implémentation en C des SVM [Joachims,], et nous exécutons les calculs sur des machines ayant 3GHz de CPU et 3Go de RAM).

Le tableau VII.1 présente les valeurs moyennes de ces paramètres pour les machines Pn/Gt, Pn/Ob et Gt/Ob. La valeur moyenne de C_{dat} trouvée est de 0.032 ($\pm$0.004).

Nous observons alors les tendances suivantes, concernant les critères mesurés : en prenant des valeurs de C plus grandes,

1) l'erreur $\xi\alpha$ diminue ;
2) la dimension VC augmente ;
3) le nombre d'erreurs sur l'ensemble d'apprentissage diminue ;
4) le nombre de vecteurs supports diminue ;
5) le temps CPU augmente.

Paramètre	$C = C_{dat}$	$C = \mathbf{1}$	$C = \mathbf{10}$	$C = \mathbf{20}$
h	122360	1351333	3471416	3690315
Erreur $\xi\alpha$	13.07	12.41	12.22	12.20
nb err. app.	3493	3392	3347	3347
nb BSV	8749	8286	8156	8144
nb SV	8783	8331	8202	8190
CPU(s)	33	298	422	711

Tab. VII.1 Valeurs moyennes des caractéristiques des SVM linéaires apprises (Pn/Gt, Pn/Ob, Gt/Ob) pour différentes valeur de C.

Les observations 2) à 6) sont en fait prévisibles. En effet, choisir C plus grand revient à pénaliser de façon plus importante les outliers, ce qui conduit à un rétrecissement de la marge afin d'en limiter le nombre. C'est ce qui explique que pour les faibles valeurs de C, le nombre de vecteurs supports dans la marge (les BSV, *cf.* section V-2) ainsi que le nombre d'erreurs sur l'ensemble d'apprentissage sont plus élevés (à cause d'une pénalisation plus lâche). A une marge plus petite correspond un classificateur aux propriétés de généralisation moins prévisibles donc une dimension VC plus élevée. Cette dernière propriété peut être retrouvée de façon plus directe, étant donné que l'estimation de la dimension VC est obtenue en considérant $h \approx r^2||\mathbf{w}||^2$ (*cf.* section V-2-E) et que la largeur de la marge est définie par $\frac{2}{||\mathbf{w}||}$. Ce comportement indique que le critère de dimension VC minimale ne peut être utilisé pour sélectionner le paramètre C (puisque h est croissant en C)[1].

La tendance de l'erreur $\xi\alpha$ est plus inattendue (observation 1). Joachims présente cette erreur comme un critère possible pour sélectionner la valeur de C [Joachims, 2000], or il s'avère que dans notre cas, elle est décroissante en fonction de C (la tendance est confirmée par des expériences complémentaires). En d'autres termes, elle traduit d'avantage la tendance du risque empirique que celle du risque fonctionnel puisqu'elle sélectionne des valeurs de C les plus grandes (donnant des marges les plus petites et un plus petit nombre d'erreurs sur l'ensemble d'apprentissage).

Pour confirmer ces intuitions, nous réalisons un test sur des données de réglage `SUB-INS-D` (*cf.* chapitre II). Les résultats de ce test sont donnés dans le tableau VII.2 pour les trois classes Pn, Gt et Ob, et les valeurs de C considérées.

[1] nous verrons dans la suite que ce critère reste néanmoins utile pour sélectionner la paramétrisation du noyau.

Etant donnée la grande variabilité des données audio[2], les meilleurs résultats sont obtenus, comme prévu, en gardant un maximum de marge, c'est-à-dire les valeurs de C les plus petites. Le choix $C = C_{dat}$ fournit les meilleurs résultats, suivi de $C = 1$. Cependant, le premier choix présente certains inconvénients. Si l'on envisage de régler la paramétrisation du noyau en exploitant les critères considérés (h et erreur $\xi\alpha$), le paramètre C doit être fixé (à la lumière de la discussion précédente), ce qui n'est pas réalisé en utilisant les valeurs C_{dat} qui dépend du noyau utilisé (à cause de (VII.1)). En conséquence, nous sélectionnons la valeur $C = 1$ qui correspond à une valeur moyenne de C_{dat} mesurée sur un ensemble plus large d'instruments. Ce choix sera validé dans la suite.

Paramètre	C=C_{dat}	C=1	C=10	C=20
Pn	80.3	70.4	62.6	62.0
Gt	82.3	87.8	90.3	90.5
Ob	97.4	96.7	96.5	96.5
Moyenne	**86.7**	85.0	83.1	83.0
Ecart-type	9.4	13.4	18.0	18.4

Tab. VII.2 Résultats de classification avec SVM linéaires pour différentes valeurs de C.

VII-4. Choix et paramétrisation du noyau

Nous souhaitons maintenant, en nous fixant une valeur de C, sélectionner le noyau et sa paramétrisation les mieux appropriés, en nous basant sur les critères de dimension VC (empirique) et/ou d'erreur $\xi\alpha$. Rappelons qu'à *priori* les meilleures configurations réalisent une dimension VC minimale et une erreur $\xi\alpha$ minimale.

Nous considèrons les noyaux, polynômial :

$$k(\mathbf{x}, \mathbf{y}) = \left(\frac{\mathbf{x}.\mathbf{y}}{d}\right)^{\delta}, \tag{VII.2}$$

et RBF gaussien :

$$k(\mathbf{x}, \mathbf{y}) = \exp\left(-\frac{||\mathbf{x} - \mathbf{y}||^2}{d\sigma^2}\right). \tag{VII.3}$$

[2]les données de test sont issues de sources (albums) différentes de celles utilisées pour l'apprentissage, *cf.* chapitre II.

Notons qu'une mise à l'échelle a été effectuée (en divisant par la dimension des vecteurs d) afin de limiter l'intervalle de variation de $k(\mathbf{x}, \mathbf{y})$ [Schölkopf *et al.*, 1995]. Nous rappelons que les valeurs élevées des paramètres δ et σ correspondent à des surfaces de décision plus complexes alors que les petites valeurs produisent des surfaces de décision plus "planes". Nous suivons [Schölkopf *et al.*, 1995] pour le choix des paramètres à tester. Les choix intéressants de σ^2 se situent, avec la mise à l'échelle, dans l'intervalle [0,1]. Nous présentons les cas $\sigma^2 \in \{0.2, 0.5, 1\}$ qui sont assez représentatifs.

Nous réalisons l'apprentissage des trois SVM (Pn/Gt, Pn/Ob, Gt/Ob) sur les mêmes données (*cf.* section VII-3) pour les différents choix des paramètres δ, pour le noyau polynômial, et du paramètre σ pour le noyau gaussien, en mesurant les critères considérés. Les résultats sont résumés dans le tableau VII.3 en valeurs moyennes pour les trois machines.

C=1	**Noyau polynômial**				**Noyau gaussien**		
Paramètre	δ=2	δ=3	δ=4	δ=5	σ^2=1	σ^2=0.5	σ^2=0.2
h	78041	508428	12217270	128484305	3282	3324	5018
$\xi\alpha$ **err.**	10.16	9.02	13.40	16.01	6.16	5.74	4.40
nb err. app.	1498	1001	1122	1554	837	565	245
nb SV	6849	6078	9041	10808	4395	4103	6907
CPU	386	275	778	717	260	233	418

Tab. VII.3 Valeurs moyennes des caractéristiques des SVM apprises (Pn/Gt, Pn/Ob, Gt/Ob) pour différents noyaux. Les valeurs optimales des critères sont encadrées.

Nous observons que les deux critères considérés privilégient le noyau gaussien mais qu'ils ne sélectionnent pas les mêmes paramètres pour chaque noyau. En outre, nous voyons que les dimensions VC présentent, dans certains cas, des valeurs assez élevées (> 5000), notamment pour le noyau polynômial. En fait, de telles valeurs ne sont pas très "informatives" puisqu'elles donnent des bornes sur le risque (*cf* expression (V.17)) trop grossières. A titre d'exemple, une dimension VC de 5000 donne une borne de 60% (approximativement) sur le risque réel (avec 60000 exemples d'apprentissage), ce qui n'est pas satisfaisant.

Pour mieux cerner le comportement des critères considérés, nous réalisons un test sur l'ensemble de réglage `SUB-INS-D`. Les résultats sont présentés dans le tableau VII.4 et mis en parallèle avec les valeurs des critères.

Nous remarquons que pour des petites valeurs du critère h (c'est le cas pour le noyau gaussien)

C=1	**Noyau polynômial**				**Noyau gaussien**		
Paramètre	δ=2	$\boxed{\delta=3}$	δ=4	δ=5	$\boxed{\sigma^2=1}$	σ^2=0.5	$\boxed{\sigma^2=0.2}$
h	78041	508428	12217270	128484305	$\boxed{3282}$	3324	5018
$\xi\alpha$ **err.**	10.16	$\boxed{9.02}$	13.40	16.01	6.16	5.74	$\boxed{4.40}$
Pn	85.1	85.2	84.4	82.2	87.9	88.0	87.0
Gt	76.3	85.5	75.8	84.7	81.6	80.4	78.6
Ob	92.8	98.6	95.7	98.8	99.1	99.1	99.0
Moyenne	84.8	**89.8**	85.3	88.6	**89.5**	89.2	88.2
Ecart-type	8.2	7.7	10.0	8.9	8.8	9.4	10.2

Tab. VII.4 Taux de reconnaissance sur les données de l'ensemble SUB-INS-D pour différents noyaux. Les valeurs des paramètres préconisées par les deux critères h et $\xi\alpha$ sont encadrées. Les meilleurs taux de reconnaissance sont donnés en gras.

la tendance de celui-ci suit celles des performances. De meilleurs taux de reconnaissance moyens (avec le noyau gaussien) sont bien obtenus pour les valeurs moyennes de h les plus petites (le maximum 89.5% est obtenu pour $h = 3282$).

Le critère $\xi\alpha$, par contre, ne suit pas la tendance des résultats de classification. Celui-ci sélectionne les valeurs de σ^2 les plus petites qui réalisent le minimum d'erreur sur l'ensemble d'apprentissage, mais pas sur l'ensemble de test : nous sommes ici face à un problème de *sur-apprentissage*.

Cependant, nous observons que dans les situations où le critère h ne peut être exploité (lorsque h est trop grand, c'est le cas pour le noyau polynômial) le critère $\xi\alpha$ sélectionne la valeur du paramètre δ qui correspond au meilleur taux de reconnaissance (89.8% pour $\delta = 3$).

Nous décidons donc de considérer en priorité le critère de dimension VC, lorsque celui-ci est pertinent (c'est-à-dire inférieur à 5000) tout en gardant la possibilité de recourir au critère $\xi\alpha$ dans les situations où h est trop grand. Nous validerons cette procédure dans la section VII-5.

Il est intéressant de remarquer que les deux noyaux, polynômial et gaussien, avec leurs réglages optimaux donnent des performances similaires (aux intervalles de confiance près). Ce résultat est en fait assez connu : le choix du type de noyau est moins important que le contrôle de la capacité (dépendant du réglage des paramètres) dans le type de structure choisie [Schölkopf *et al.*, 1995].

Dans la suite, nous gardons le noyau gaussien (qui assure une dimension VC plus petite et

permet ainsi d'obtenir une meilleure estimation du risque) et nous validons la procédure de réglage des paramètres proposée sur l'ensemble de test SUB-INS-T.

VII-5. Validation de la procédure de réglage des paramètres des SVM

Le tableau VII.5 résume les valeurs moyennes des paramètres relatifs aux SVM apprises pour toutes les paires de classes du corpus SUB-INS, en considérant un noyau gaussien dont nous faisons varier le paramètre σ^2 dans l'ensemble $\{1, 0.5, 0.2\}$ pour chacune des valeurs de C parmi 1, 10 et 20.

Paramètre	σ^2	1	0.5	0.2
$C = \mathbf{1}$	h	2142	2154	3630
	$\xi\alpha$ **err.**	3.32	2.64	2.50
	nb err. app.	277	162	60
	nb SV	2304	2462	5891
	CPU	101	109	309
$C = \mathbf{10}$	h	10341	8184	6313
	$\xi\alpha$ **err.**	1.99	1.90	2.41
	nb err. app.	110	42	4
	nb SV	1362	1797	5659
	CPU	189	203	385
$C = \mathbf{20}$	h	16517	11713	6928
	$\xi\alpha$ **err.**	1.79	1.83	2.40
	nb err. app.	80	25	1
	nb SV	1227	1736	5643
	CPU	227	246	352

Tab. VII.5 Valeurs moyennes des caractéristiques des 28 SVM apprises pour les 8 classes du corpus SUB-INS, avec différentes valeurs de C et différentes valeurs de σ du noyau gaussien. Les valeurs des critères h et $\xi\alpha$ sélectionnées sont encadrées.

Rappelons que la valeur de C que nous préconisons à la lumière de la discussion précédente est $C = 1$. Pour chaque valeur de C, nous indiquons le critère sélectionné pour choisir la valeur optimale de σ^2. Ainsi, nos hypothèses sont que :

1) pour C=1, σ^2=1 est le meilleur choix : c'est celui qui correspond à la plus petite dimension VC, h=2141 (< 5000) ;
2) pour $C = 10$, les dimensions VC étant "trop" grandes, nous choisissons σ^2=0.5 qui réalise l'erreur $\xi\alpha$ la plus petite (1.90%) ;
3) enfin, pour C=20, nous retenons σ^2=1 pour une erreur $\xi\alpha$ minimale à 1.79% ;

et le meilleur choix parmi ceux-ci est : $\underline{C=1 \text{ et } \sigma^2=1}$.

Nous vérifions la validité de ces hypothèses en considérant les résultats de classification obtenus sur l'ensemble de test `SUB-INS-T` et présentés dans le tableau VII.6.

Comme prévu, les meilleurs résultats sont obtenus pour C=1 et σ^2=1 (71.4% en moyenne). Notons que ces résultats sont similaires à ceux obtenus avec $C = C_{dat}$, ce qui confirme que le fait de fixer C à 1 n'est pas pénalisant par rapport à un choix de C "adaptatif". L'hypothèse 3) est aussi vérifiée : le meilleur choix de σ^2 pour C=20 est bien 1. L'hypothèse 2) n'est en revanche pas vérifiée mais la valeur de σ^2 prédite reste la plus proche de la valeur produisant les meilleurs résultats.

Ainsi, la procédure de réglage des paramètres des SVM proposée peut être considérée comme efficace puisqu'elle a permis de prédire la meilleure paramétrisation du noyau pour une valeur de C optimale fixée à priori.

Dans la suite nous "figeons" le choix du paramètre C et le type de noyau en retenant $C = 1$ et le noyau gaussien (qui donne des dimensions VC plus petites et par suite des bornes plus fines sur le risque). Nous utiliserons la procédure précédemment décrite pour sélectionner le paramètre σ du noyau.

Nous cherchons maintenant à savoir s'il est plus approprié de choisir les valeurs $\sigma_{p,q}$ les mieux adaptées à chaque SVM $\mathcal{C}_{p,q}$ relative à la paire de classe $\{\Omega_p, \Omega_q\}$ (plutôt qu'une même valeur obtenue à partir de critères moyens sur toutes les paires). En fait, nous obtenons alors des résultats similaires (aux intervalles de confiance près) à ceux obtenus avec une même valeur de σ (optimale en moyenne), comme nous pouvons l'observer dans la deuxième colonne du tableau VII.7.

Une amélioration possible concerne la réduction de la complexité. L'idée est de garder une structure de noyau simple (linéaire) pour les paires de classes facilement différenciées. Afin de

	C=C_{dat}			$\boxed{C=1}$			C=10			C=20		
σ^2	1	0.5	0.2	**1**	0.5	0.2	1	0.5	0.2	1	0.5	0.2
Pn	87.8	87.5	85.6	87.8	87.8	86.0	88.1	87.7	87.7	87.7	87.7	88.2
Gt	73.7	73.7	71.1	73.7	73.8	71.4	74.3	72.7	70.4	73.9	72.3	69.6
Ob	78.6	77.2	73.1	78.8	77.2	72.8	77.7	76.3	75.1	77.3	76.2	75.4
Cl	64.0	64.6	68.1	64.0	65.0	68.5	64.9	64.3	67.0	64.5	63.4	66.9
Fh	56.4	56.4	56.0	56.4	55.3	53.8	51.6	51.3	48.8	49.9	49.3	48.7
Tr	69.8	70.0	68.2	69.9	69.6	68.6	70.0	70.0	69.7	70.0	70.7	69.5
Co	58.5	57.0	54.1	58.4	56.4	53.1	55.2	53.8	52.0	54.7	53.6	52.0
Vl	82.6	81.9	80.0	82.5	81.5	79.9	79.9	78.9	80.5	79.5	78.7	80.5
Moyenne	**71.4**	71.0	69.5	**71.4**	70.8	69.2	**70.2**	69.4	68.9	**69.7**	69.0	68.9
Ecart-type	11.3	11.3	10.7	11.4	11.6	11.4	12.5	12.4	13.2	12.8	12.7	13.4

Tab. VII.6 Taux de reconnaissance sur l'ensemble de test SUB-INS-T pour différents noyaux. Les valeurs des paramètres préconisées par les deux critères h et $\xi\alpha$ sont encadrées. Les meilleurs taux de reconnaissance sont donnés en gras.

réaliser cela :

- nous calculons des SVM linéaires pour toutes les paires en mesurant à chaque fois l'erreur $\xi\alpha$;
- si la valeur de l'erreur $\xi\alpha$ est proche de 0 ($< 1\%$) pour la paire de classe $\{\Omega_p, \Omega_q\}$, nous gardons pour celle-ci la SVM linéaire, sinon nous utilisons un noyau gaussien dont nous réglons le paramètre σ.

Des SVM linéaires ont ainsi été retenues pour sept paires de classes parmi les 28 possibles. Les résultats de test obtenus avec ce système sont donnés dans la troisième colonne du tableau VII.7. Nous observons que nous obtenons à moindre coût des performances similaires à celles obtenues avec un système utilisant un noyau gaussien pour toutes les paires.

$\sigma^2_{p,q}$	Meilleur	Plus simple
Pn	87.9	87.9
Gt	73.7	73.7
Ob	77.5	77.3
Cl	64.9	65.2
Fh	56.5	56.6
Tr	69.4	69.6
Co	57.4	57.3
Vl	82.4	82.6
Moyenne	71.2	71.3
Ecart-type	11.4	11.3

Tab. VII.7 Résultats de classification sur `SUB-INS-T` en utilisant, dans la première (respectivement la deuxième) colonne, la meilleure valeur de σ pour chaque paire (respectivement un noyau linéaire plutôt qu'un noyau gaussien, si le noyau linéaire réalise une erreur $\xi\alpha < 1$). C est fixé à 1.

VII-6. Décision en temps

A ce stade du développement nous ne nous sommes pas encore intéressés au choix d'une fenêtre de décision temporelle adéquate. En effet, les taux de reconnaissance précédents ont été calculés en prenant une décision par observation, correspondant à une fenêtre temporelle de 32ms. Or, il est possible d'effectuer la prise de décision sur des fenêtres temporelles plus longues, regroupant N_t observations (qui correspondent à N_t fenêtres d'analyse temporelles recouvrantes sur une durée $T = (Nt-1)\frac{H}{f_s} + \frac{N}{f_s}$ secondes).

La longueur de la fenêtre de décision dépend de l'application envisagée. Par exemple, pour un système de reconnaissance automatique des instruments de musique en temps réel cette longueur doit rester assez petite, typiquement de 1 à 4 secondes[3]. Pour d'autres applications telles que l'archivage automatique d'enregistrements en solo par exemple, les décisions peuvent être prises sur toute la longueur de l'enregistrement, typiquement de quelques minutes, etc. Notons que les fenêtres de décision peuvent elles-même être recouvrantes pour permettre une segmentation plus fine du signal.

[3]Notons que dans ce cas, des fenêtres de décision trop courtes sont à éviter puisque l'on doit laisser à l'utilisateur le temps nécessaire à l'interprétation de l'information de sortie, typiquement un affichage...

Nous présentons dans le tableau VII.8 les taux de reconnaissance obtenus, sur le même ensemble de test que précédemment (`SUB-INS-T`), en variant la taille de la fenêtre de décision de T=32ms (N_t=1 observation) à T=4s (N_t=249 observations) et en gardant un recouvrement de $N_t - 1$ observations entre deux fenêtres successives.

Décision	32ms	1s	2s	4s
Pn	87.8	99.4	99.9	100.0
Gt	73.7	91.5	93.8	96.4
Ob	78.8	89.1	91.5	94.0
Cl	64.0	89.1	94.8	97.8
Fh	56.4	75.0	79.8	83.0
Tr	69.9	81.4	83.5	86.5
Co	58.4	65.2	67.3	70.8
Vl	82.5	92.9	94.0	96.1
Moyenne	71.4	85.5	88.1	90.6
Ecart-type	11.4	11.0	10.6	9.9

Tab. VII.8 Résultats de classification en utilisant des fenêtres de décision temporelles de plus en plus longues (de gauche à droite).

Nous remarquons une nette amélioration des performances en utilisant des fenêtres de décision plus longues. Des décisions prises sur toute la longueur des fichiers conduisent sur ces données de test à des taux de reconnaissance de 100%.

VII-7. Conclusions

Nous avons étudié deux critères permettant de régler les paramètres des SVM (paramètre C et noyau) à partir des données d'apprentissage, sans passer par une étape de test : un critère visant à minimiser une estimation empirique de la dimension VC et un deuxième critère, visant à minimiser une estimation $\xi\alpha$ de l'erreur en généralisation.

Nous avons montré que ces critères ne permettent pas de choisir le paramètre C. Mais ce paramètre doit être fixé afin de pouvoir exploiter les deux critères pour le réglage du noyau. La valeur $C = 1$ a été retenue. Celle-ci permet de favoriser une grande marge (donc une dimension VC petite), ce qui est mieux approprié à notre problème, eu égard à la grande variabilité des données (issues d'enregistrements différents).

Le critère de dimension VC s'avère plus fiable que le critère d'erreur $\xi\alpha$, mais le premier n'est pas toujours exploitable car il arrive qu'il prenne des valeurs trop élevées, et perd ainsi de sa pertinence. C'est dans ces situations que nous faisons appel au critère d'erreur $\xi\alpha$. Certes, ce dernier ne garantit pas systématiquement l'obtention des meilleurs réglages, mais il élit des solutions de paramètres qui restent raisonnables.

Par ailleurs, nous avons retrouvé que des noyaux différents, lorsqu'ils sont correctement paramétrés, donnent lieu à des résultats de classification similaires et nous avons retenu le noyau gaussien (qui permet des dimensions VC plus petites que celles réalisées par le noyau polynômial. Nous) avons également proposé de se contenter d'un noyau linéaire lorsque cela est suffisant à une bonne discrimination d'une paire de classes particulières, pour un allègement de la charge de calcul globale.

Nous avons ensuite introduit l'utilisation de fenêtres de décision temporelle plus longues, regroupant les décisions prises sur une sucession de fenêtres d'analyse courtes. Nous avons observé que les taux de reconnaissance étaient systématiquement plus élevés en utilisant des fenêtres de plus en plus longues. Nous avons arrêté notre choix sur des fenêtres de décision de taille 4s pour le système de reconnaissance final. Ce choix permet de garder en vue la possibilité de réaliser la reconnaissance des instruments en temps réel.

TROISIEME PARTIE

Application à la classification des instruments de musique

Introduction de la troisième partie

Nous nous intéressons maintenant spécifiquement au problème de la reconnaissance des instruments de musique. Rappelons qu'à la lumière des expériences préliminaires présentées précédemment, nous savons que :

- parmi les approches de sélection d'attributs envisagées, l'approche FSFC[4] est la plus convenable pour notre problème, et il peut être avantageux de réaliser une sélection binaire des attributs ;
- parmi les classificateurs considérés nous avons intérêt à utiliser les SVM dont nous savons régler les paramètres, et particulièrement les SVM munis d'un noyau gaussien.

Au chapitre VIII nous analysons la sortie de l'algorithme de sélection retenu (FSFC), utilisé dans une configuration multi-classes (non-binaire). Nous allons voir que cette approche permet de produire une organisation des attributs dans laquelle ceux qui sont "similaires" sont regroupés dans les mêmes clusters tout en étant rangés par ordre d'efficacité pour la tâche envisagée. Cela permet de se faire une idée de l'utilité des différents descripteurs expérimentés pour la reconnaissance des instruments. Ensuite, nous étudions l'apport d'un traitement différencié des attaques de notes de musique, connues pour être des éléments importants de distinction des instruments.

Au chapitre IX nous abordons la classification hiérarchique des instruments, qui constitue la solution que nous retenons pour le système de reconnaissance final. Cette approche suppose l'utilisation d'une taxonomie hiérarchique des instruments. Nous envisageons deux possibilités :

[4]Fisher-based Selection of Feature Clusters

- l'utilisation d'une taxonomie "naturelle" inspirée des familles d'instruments ;
- l'utilisation d'une taxonomie inférée automatiquement à partir des exemples, et qui vise à maximiser les taux de reconnaissance.

Les performances des schémas de classification basés sur ces deux taxonomies sont comparées aux performances d'un système de classification de référence non-hiérarchique.

L'approche de sélection binaire est ensuite mise à contribution et nous montrons l'amélioration des performances apportée par l'adoption de cette stratégie de sélection.

Enfin, au chapitre X, nous montrons que la classification hiérarchique constitue une solution appropriée pour la reconnaissance des instruments en présence dans des extraits musicaux multi-instrumentaux. Notre approche consiste à construire des classes à partir de toutes les combinaisons d'instruments pouvant être joués simultanément, en exploitant le fait que le nombre de classes possibles se trouve réduit à un niveau donné de la taxonomie. L'avantage de cette approche est qu'aucune séparation préalable des sources musicales, ni aucune étape d'estimation de fréquences fondamentales multiples n'est requise.

VIII. Caractérisation spécifique à la classification des instruments de musique

Dans ce chapitre, nous proposons une organisation des attributs pertinents pour la reconnaissance des instruments. Celle-ci est obtenue grâce à notre approche de sélection FSFC appliquée à l'ensemble des attributs explorés (résumés dans le tableau IV.1). Nous étudions par ailleurs, l'apport d'un traitement différencié entre les segments correspondants aux attaques (*onsets*) des notes et ceux correspondants aux parties tenues des sons. C'est en effet une propriété psychoacoustique reconnue que les attaques jouent un rôle important dans notre perception du timbre instrumental. Cette étude est le fruit d'une étroite collaboration avec Pierre Leveau, qui a donné lieu à une publication commune [Essid *et al.*, 2005a].

VIII-1. Organisation des attributs pour la reconnaissance des instruments

Grâce à notre approche de sélection FSFC (*cf.* section VI-7-A) nous sommes en mesure d'organiser les attributs par catégories (clusters), triées par ordre décroissant d'efficacité. Au sein de chaque cluster (un cluster regroupant un sous-ensemble d'attributs "proches" les uns des autres, considérés comme redondants), les attributs sont également triés par ordre décroissant d'efficacité. Ce tri découle de la sortie des différentes instances de l'algorithme de sélection Fisher, appliqué dans chaque cluster puis sur les représentants des différents clusters (comme décrit dans la section VI-7-A).

L'organisation obtenue à partir des données d'apprentissage `INST-A` relatives aux 19 instruments considérés (*cf.* section II) est présentée dans le tableau VIII.1 où les 40 meilleurs clusters ont été retenus. Les $d = 40$ attributs sélectionnés par FSFC pour la classification multi-classes

des instruments sont simplement les premiers éléments apparaissant dans chaque cluster. Ce choix de d résulte des expériences préliminaires sur la sélection d'attributs (*cf.* chapitre VI).

1 : ***Cp2****, Ld15, δ^2lTw, OBSI7, Sk, DWCH28*
2 : ***OBSIR1****, DWCH11, qCq3, Si5, SMR22, OBSI8, dCq2, Cc4, ASF9, AC8, AC3, AC47, ASF11, AC11, SMR30, AC28*
3 : ***Cp3****, Cc7, SMR13, Cp10, Si13, Ld1, dCq8, Si17, AC33*
4 : ***Cp7****, Cp4, Ld5, uCq5, Cc10, OBSIR2, AC39, W2* — *5 :* ***OBSI5****, So* — *6 :* ***Ld14****, Ld23* — *7 :* ***Sh***
8 : ***tCq2****, dCq9, SMR9, dCq4, DWCH12, SMR3, SMR7, SMR27, Ld9, SCF17, SMR39* — *9 :* ***SCF5***
10 : ***Sp****, lZ* — *11 :* ***AR2****, SMR4, dCq5, uCq4, SCF19, SCF21, Sd, lTk, W5, SCF15, AC1, AC43, δlTw*
12 : ***OBSI2****, Ld3, Ld6, SMR19, SCF12, ASF17, Ld2, ASF6, tCq8* — *13 :* ***AR1*** — *14 :* ***Cc2****, Sc*
15 : ***ASF14****, Ld19* — *16 :* ***Cc5****, (ampl. AM)×(freq. AM) 10-40Hz, DWCH10, DWCH23, Cc9*
17 : ***W1****, Ld16, Ld22* — *18 :* ***qCq2****, Cc3, OBSIR5, qCq5, Cp6, DWCH24, SMR14, Sa, SMR18, dCq1, AC42*
19 : ***SCF13****, SCF6, Ld11, SCF8, ampl. AM heurist. 4-8Hz, Ld17* — *20 :* ***Ld8*** — *21 :* ***DWCH25***
22 : ***OBSI3****, SCF2, ASF3, SCF3, Ld20* — *23 :* ***ASF15*** — *24 :* ***SCF9****, SCF22*
25 : ***(ampl. AM)×(freq. AM) 4-8Hz*** — *26 :* ***SCF16****, tCq3* — *27 :* ***W4***
28 : ***ASF10****, DWCH14, ASF19, Si11, SMR11, SMR20, Si2, AC22, SMR25, SMR43, Si9, SMR6, DWCH15, AC23, DWCH16, SMR40, AC40, AC10, SMR15, AC26, SMR35, SMR12, AC41, AC12, SMR21, AC37, δSw, AC45, δ^2lTa, δ^2Cc0, DWCH17, δ^2dCq1, δtCq3, δ^2Cp6*
29 : ***ASF16****, dCq3, OBSIR6* — *30 :* ***uCq3****, OBSI4, Cc1*
31 : ***Ss****, Cp5, ASF20, Z, DWCH13, OBSI1, SMR36, Si3, AC7, AC44, Si4, SMR23, Si14, AC49, Si20, δlTk, SMR34, SMR29, δ^2lTa, δ^2Ta*
32 : ***Ld7****, Cp1* — *33 :* ***SCF11*** — *34 :* ***Ld10*** — *35 :* ***ampl. AM 4-8Hz***
36 : ***Si1****, Si7, OBSIR4, SMR5, AC25, AC18, SMR45, qCq9, uCq8, SMR17, DWCH20, SMR16, SMR44, AC30, Sw, SMR50, AC31, δTa, δLd1, δdCq5, δ^2dCq2, δtCq5, δLd21, δqCq3, δCp4, δ^2 Cp10, δLd22, δ^2Sc, δ^2qCq3, δ^2uCq1, δ^2Sw, δuCq3, δ^2Ld19, δqCq1, δ^2Ld22, δLd19, δLd5, δ^2Ld10*
37 : ***tCq1*** — *38 :* ***ASF23*** — *39 :* ***DWCH26****, Cc8, ASF13, tCq4, AC2*
40 : ***ampl. AM heurist. 10-40Hz***.

Tab. VIII.1 Organisation des attributs. Les 40 clusters les plus efficaces par ordre (décroissant) d'efficacité.

Les observations suivantes peuvent être faites concernant les clusters d'attributs :

– les 40 clusters les plus performants (parmi les 60 considérés pour le clustering) ne couvrent que 43% des attributs initialement considérés (233/543 attributs), pourtant tous les des-

cripteurs (paquets d'attributs) considérés sont représentés dans ces 40 clusters (au travers d'un sous-ensemble de leurs composantes, par exemple 13 coefficients sur 23 pour le descripteur ASF) ;

– des attributs extraits dans des domaines différents (temporel, spectral, cepstral et perceptuel) se retrouvent dans des mêmes clusters : le premier cluster, par exemple, regroupe un coefficient cepstral ($Cp2$), un coefficient de Loudness ($LD15$), deux coefficients issus d'une représentation spectrale ($OBSI7$ et Sk) et deux coefficients issus d'une représentation temps-fréquence ($\delta^2 lTw$ et $DWCH28$). De plus, pour les descripteurs spectraux, des attributs mesurés dans des régions fréquentielles éloignées sont parfois assignés aux mêmes clusters. Il apparaît ainsi que la volonté de concevoir des descripteurs caractérisant des propriétés différentes des classes d'instruments ne soit pas reflétée dans les attributs extraits, qui présentent souvent des distributions de valeurs assez proches.

Intéressons-nous maintenant aux attributs sélectionnés (ceux qui ont le rang 1 dans chaque cluster, ils sont présentés en gras et listés dans le tableau VIII.2). Nous observons que :

– les descripteurs les plus fréquemment sélectionnés sont des descripteurs spectraux. 18/40 des descripteurs sélectionnés sont des descripteurs spectraux, parmi lesquels on retrouve la pente spectrale Ss, le coefficient d'irrégularité spectrale $Si1$, les 2 coefficients AR, 5 coefficients ASF, 5 coefficients SCF, 3 $OBSI$ et $OBSIR1$. Notons que ces 4 derniers coefficients sont classés dans les attributs les plus efficaces (deux d'entre eux sont classés dans les cinq premiers attributs), ce qui indique que ce nouveau descripteur est efficace pour notre tâche. En outre, nous remarquons, concernant les attributs calculés sur plusieurs sous-bandes fréquentielles, que la majorité de ceux qui sont sélectionnés est associée à des régions de moyennes fréquences (autour de celle du La4 à 440Hz) ;

– 9 coefficients cepstraux se trouvent parmi les attributs sélectionnés. Ils comprennent des coefficients issus de représentations cepstrales différentes (Cc, Cp, uCq, tCq et qCq). Les coefficients cepstraux font partie des attributs les mieux classés par l'algorithme de sélection, particulièrement les attributs Cp qui se positionnent à trois reprises parmi les 5 meilleurs attributs ;

– 3 paramètres perceptuels sont classés parmi les 10 premiers attributs sélectionnés : la sharpness Sh, l'étendue perceptuelle Sp et le coefficient de loudness $Ld14$. Au total on retrouve 6 paramètres perceptuels parmi les attributs sélectionnés ;

– les attributs obtenus à partir de la transformée en ondelettes s'avèrent également utiles à

la classification des instruments, 4 de ces attributs ont été retenus ($W1$, $DWCH25$, $W4$, $DWCH26$) ;

– enfin des attributs temporels considérés, seuls des paramètres de modulation d'amplitude ont été sélectionnés : 2 attributs décrivant le trémolo, le produit de la fréquence AM et de l'amplitude AM, ainsi que l'amplitude AM dans l'intervalle 4-8Hz, et un attribut décrivant la rugosité des sons, *i.e.* l'amplitude AM heuristique dans l'intervalle 10-40Hz.

Les rapports signal à masque (SMR) n'ont pas été retenus par l'algorithme FSFC dans ce contexte, même s'ils sont largement représentés dans les 40 clusters sélectionnés. Nous verrons qu'ils seront utiles dans un contexte multi-instruments.

VIII-2. Utilité d'un traitement différencié des attaques de notes

Des études en cognition et acoustique musicale indiquent que les transitoires d'attaque et de fin de notes musicales intègrent une part importante de l'information utile à l'identification des instruments (voir par exemple [Clark *et al.*, 1964, McAdams *et al.*, 1995]). L'information sur le mode de production des sons est essentiellement localisée au début et à la fin des notes, typiquement les impulsions de souffle pour les intruments à vent, les coups d'archet pour les cordes frottées ou les pincements et coups de marteau pour les cordes pincées et frappés (par exemple, le piano et la guitare). Des expériences de cognition musicale ont ainsi montré que des descripteurs caractérisant le début des notes de musique participent à la discrimination des instruments par l'Homme.

Dans le contexte de la reconnaissance automatique des instruments à partir de notes isolées, des descripteurs acoustiques extraits à partir des transitoires d'attaque (par exemple la durée de l'attaque, le facteur de crête, etc.) se sont montré efficaces, et ce particulièrement pour la discrimination de familles d'instruments [Eronen, 2001a, Peeters, 2003]. Cependant, l'extraction de tels descripteurs à partir de phrases musicales dans des conditions de jeu réalistes, impliquant des transitions plus ou moins rapides entre notes, n'est pas aisée. Comme nous l'avons vu, les signaux de musique sont dans ce cas analysés sur une succession de fenêtres temporelles de taille fixe, sans qu'aucune distinction ne soit faite entre segments transitoires et segments non-transitoires. Du fait que les segments non-transitoires sont généralement de durée beaucoup plus courte que les segments transitoires, l'information véhiculée par ces derniers se retrouve diluée dans l'étendue du signal et son impact sur les performances finales de classification devient

faible.

Nous cherchons à savoir s'il est possible d'exploiter efficacement les propriétés des transitoires d'attaque au sein d'un système de reconnaissance des instruments à partir d'extraits mono-instrumentaux. Cela suppose que nous puissions détecter les segments comprenant les transitoires (d'attaques), pour effectuer un traitement différencié de ces derniers et des segments non-transitoires (le reste des segments). Nous ferons pour cela appel à la technique de segmentation décrite dans la section III-3-B. Notre approche consiste à produire des sélections d'attributs particulières pour chaque type de segments (transitoires[1] et non-transitoires), sélections qui sont utilisées pour construire des classificateurs différents pour chaque type de segment.

Nous rappelons que la segmentation retenue se base sur un détecteur d'attaques : lorsqu'une attaque est détectée, N_t fenêtres d'analyse (courtes), comprenant et suivant l'attaque, sont considérées comme faisant partie d'un segment transitoire. Deux "longueurs de transitoires" sont expérimentées : $N_t = 2$ ($\approx$ 50ms) et $N_t = 4$ ($\approx$ 80ms). Ces choix découlent de la nécessité de réaliser un compromis qui englobe des transitoires de durées variables (ces durées peuvent être inférieures à la longueur de la fenêtre d'analyse ou au contraire correspondre à celles de plusieurs fenêtres d'analyse).

Nous exploitons dans les expériences suivantes le corpus `SUB-INS`. Après segmentation, chaque fenêtre d'analyse de 32ms est affectée à l'une des deux catégories que nous nous sommes données : "transitoires" ou "non-transitoires".

Deux sous-ensembles de données sonores sont ainsi constitués : un sous-ensemble d'observations de fenêtres transitoires et le sous-ensemble complémentaire formé des observations de fenêtres non-transitoires.

A. Attributs sélectionnés sur les différents segments

Notre algorithme de sélection d'attributs FSFC (ciblant $d = 40$ attributs en sortie) a été exécuté sur les ensembles d'apprentissage suivants (issus de `SUB-INS-A`) :

– deux ensembles comprenant les observations issues de segments transitoires pour les deux variantes $N_t = 2$ et $N_t = 4$; les sous-ensembles d'attributs sélectionnés correspondant seront désignés respectivement par AS-T2 et AS-T4 ;

[1] Nous utilisons le terme "transitoires" pour désigner les transitoires d'attaque uniquement.

– deux ensembles regroupant les observations des segments étiquetés comme non-transitoires, par la même méthode de segmentation ; les sous-ensembles d'attributs sélectionnés correspondant seront désignés respectivement par AS-S2 et AS-S4.

Nous comparons les différentes sélections d'attributs obtenues à celle qui résulte de l'application de FSFC à toutes les données d'apprentissage, indépendamment du type de segment et qui est désignée par FS-R. Les attributs sélectionnés pour les différents types de segment et à partir des deux variantes de segmentation sont présentés dans le tableau VIII.2. Dans chaque cas (AS-T2, AS-S2, AS-T4 et AS-S4), nous présentons en gras les attributs qui n'ont pas été choisis dans la sélection de référence AS-R.

Nous constatons que les attributs sélectionnés pour les segments non-transitoires (AS-S2 et AS-S4) sont quasi les mêmes que ceux qui ont été retenus dans la sélection de référence. Ce résultat est prévisible étant donnée la faible proportion des segments transitoires (environ 10% des observations) qui rend difficile la prise en compte de leurs caractéristiques par l'algorithme de sélection. Nous en déduisons que la sélection AS-R n'intègre en fait que des caractéristiques des segments non-transitoires.

De nouvelles variables apparaissent dans les sélections spécifiques aux segments transitoires : 14 nouveaux attributs dans AS-T2 et 13 dans AS-T4. Ce sont encore, pour la plupart, des descripteurs spectraux. Notons la présence d'un coefficient de variation temporelle du cepstre ($\delta Cp1$) dans AS-T2.

B. Pouvoir de discrimination des différents segments

Nous comparons le pouvoir de discrimination des différents segments à l'aide de mesures de séparabilité S (*cf.* section VI-5-A) obtenues pour les 4 ensembles de données décrits précédemment et les attributs qui leur sont associés par l'algorithme de sélection. Ces mesures sont représentées dans la figure VIII.1 à partir de laquelle, il peut être déduit que :

– les valeurs de séparabilité obtenues avec les données issues des segments transitoires (AS-T2, AS-T4) sont supérieures à celles obtenues sur les données des segments non-transitoires mais également supérieures à celles obtenues avec les attributs des données de référence ;
– des deux possibilités de segmentation, celle correspondant à AS-T4 donne les meilleures performances de séparation ;
– les attributs relatifs aux segments non-transitoires donnent lieu aux pires performances de séparation (légèrement inférieures à celles de la configuration de référence).

AS-R
Cp2, OBSIR1, Cp3, Cp7, OBSI5, Ld14, Sh , tCq2, SCF5, Sp , AR2, OBSI2, AR1, Cc2, ASF14, Cc5, W1, qCq2, SCF13, Ld8, DWCH25, OBSI3, ASF15, SCF9, (freq. AM)×(ampl. AM) 4-8Hz , SCF16, W4, ASF10, ASF16, uCq3, Ss, Ld7, SCF11, Ld10, ampl. AM 4-8Hz , Si1, tCq1, ASF23, DWCH26, ampl. AM heurist. 10-40Hz.
AS-T2
*Cp2, OBSIR1, OBSI5, Cp7, (freq. AM)×(ampl. AM) 4-8Hz , Ld14, Sh , tCq2, Sp , **OBSI6**, Ld8, **Sf**, OBSI2, OBSI3, AR1, Cc2, ASF15, **dCq3**, Ld7, **Ld19**, W4, **SCF17**, Ld10, DWCH25, **Si6**, **Cp11**, ampl. AM 4-8Hz, qCq2, SCF13, **OBSI4**, **SCF18**, **SCF22**, **OBSI1**, tCq1, SCF16, ampl. AM heurist. 10-40Hz, **qCq1**, ASF23, δ**Cp1**,**Cc9**.*
AS-S2
*Cp2, OBSIR1, Cp3, Cp7, W1, Ld14, Sh , tCq2, SCF5, Sp , AR2, OBSI2, AR1, Cc2, ASF14, Cc5, OBSI5, SCF13, qCq2, Ld8, DWCH25, OBSI3, ASF15, SCF9, (freq. AM)×(ampl. AM) 4-8Hz , SCF16, SCF11, W4, ASF16, uCq3, Ss, Ld7, Ld10, **uCq1**, ASF10, ampl. AM 4-8Hz , Si1, tCq1, ASF23, DWCH26.*
AS-T4
*Cp2, Cp3, Cp7, (freq. AM)×(ampl. AM) 4-8Hz, Ld14, OBSIR1, Sh, tCq2, Sp , **SCF22**, Ld8, **OBSI6**, Cc5, OBSI2, OBSI3, AR1, Cc2, ASF15, **dCq3**, Ld7, **Ld19**, qCq2, W4, **SCF17**, Ld10, OBSI5, **Sf**, **Cp11**, **SCF18**, ampl. AM 4-8Hz, SCF13, **Si6**, **OBSI1**, tCq1, **OBSI4**, **qCq1**, DWCH25, ampl. AM heurist. 10-40Hz, ASF23, **SCF15**.*
AS-S4
*Cp2, OBSIR1, Cp3, Cp7, W1, Ld14, Sh , tCq2, Sp , AR2, OBSI2, AR1, Cc2, ASF14, SCF9, OBSI5, SCF13, Cc5, qCq2, SCF5, DWCH25, OBSI3, ASF15, (freq. AM)×(ampl. AM) 4-8Hz, SCF16, SCF11, Ld8, W4, ASF16, uCq3, Ld7, Ld10, **uCq1**, ampl. AM heursit. 10-40Hz , Ss, ampl. AM 4-8Hz , ASF23, tCq1, ASF10, **freq. AM 10-40Hz**.*

Tab. VIII.2 Attributs sélectionnés pour les différents segments du signal dans l'odre donné par l'algorithme de sélection.

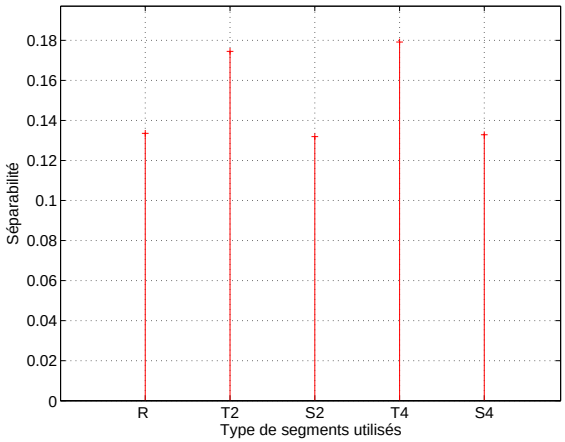

Fig. VIII.1 Mesures de séparabilité obtenues pour les attributs sélectionnés pour les données issues de segments différents.

Nous obtenons ainsi une confirmation objective du fait que les transitoires d'attaques sont particulièrement utiles à la discrimination des timbres d'instruments. Il s'agit maintenant de renforcer ces mesures par une expérience de classification.

C. Classification sur les différents segments

En nous basant sur les ensembles d'attributs sélectionnés spécifiquement pour les données issues des différentes segmentations, nous réalisons trois expériences de classification des instruments : la première exploitant uniquement les segments transitoires, la seconde uniquement les segments non-transitoires et la troisième toutes les fenêtres du signal (sans recourir à une segmentation). Nous utilisons, pour des raisons de simplicité, un classificateur SVM linéaire avec $C = 1$.

Les décisions sont prises sur N_t fenêtres courtes successives (comme décrit à la section VII-6), nous utiliserons les notations $N_t(T)$ et $N_t(S)$ pour distinguer les décisions prises par les classificateurs associés aux segments transitoires et ceux associés aux segments non transitoires. La figure VIII.2 illustre les fenêtres de décision.

Les résultats de classification obtenus sur les différents segments sont présentés dans le tableau VIII.3. Les longueurs de décision N_t sont choisies de manière à permettre une comparaison équitable des taux de reconnaissance. Elles sont imposées par les longueurs des segments transitoires, de telle sorte que $N_t = N_t(T) = N_t(S)$.

En moyenne, de meilleures performances de classification sont obtenues sur les segments transitoires. Les meilleurs résultats sont atteints en considérant des segments transitoires de

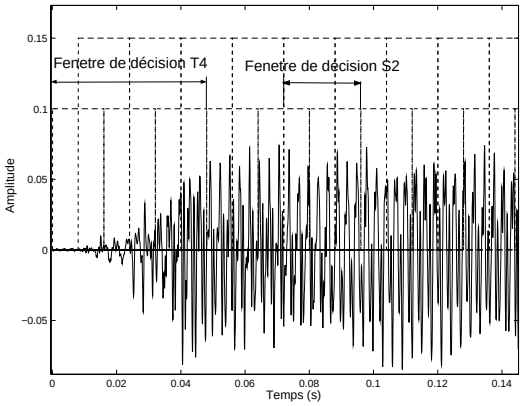

Fig. VIII.2 Exemples de fenêtres de décision. Les rectangles en trait interrompu représentent les fenêtres d'analyses courtes recouvrantes.

longueur $N_t = 4$. Nous remarquons également une dégradation des performances moyennes lorsque les segments transitoires ne sont pas pris en compte (de 71.7% à 70.4% pour $N_t = 2$ et de 73.0% à 70.4% pour $N_t = 4$).

Nous analysons les résutats plus en détail en considérant les matrices de confusions relatives à la configuration de référence "R" et à la classification sur les segments transitoires avec N_t=4. Elles sont données dans les tableaux VIII.4 et VIII.5.

La classification sur les segments transitoires résoud certaines confusions de façon assez nette. Par exemple, nous relevons l'amélioration de +11% du taux de reconnaissance du cor (Fh) qui est moins souvent confondu avec la clarinette, la guitare et le piano. En général, les instruments les plus fréquemment confondus par le système de référence profitent de la classification sur les segments transitoires. Par exemple, la guitare est confondue avec le piano dans 12.2% des cas dans la configuration standard et cette confusion n'a lieu que dans 4.2% des cas sur les segments transitoires. Cependant, de nouvelles confusions apparaissent, notamment pour la paire (guitare vs violoncelle) : de 13.1% on passe à 20.6%.

Il est important de noter que la classification sur les transitoires n'est pas avantageuse pour tous les instruments. Nous remarquons, en particulier, une dégradation des taux de reconnaissance de la clarinette sur les segments transitoires (-7%). Celle-ci est plus souvent confondue avec la trompette (de 11.3 à 16.4%) et le violon (de 5.5 à 9.0%). La clarinette est en fait mieux reconnue en ignorant les segments transitoires : les taux de reconnaissance passent de 57.7% (système de référence) à 60.0% dans la configuration "S4". Cela indique qu'il serait avantageux d'utiliser un

% correcte	R, N_t=2	T2, N_t=2	S2, N_t=2	R, N_t=4	T4, N_t=4	S4, N_t=4
Pn	82.9	83.0	86.1	84.3	83.5	87.1
Gt	69.9	68.9	64.7	71.8	74.4	68.0
Ob	79.8	84.4	77.9	81.3	86.9	74.3
Cl	56.3	49.0	58.4	57.7	50.0	60.0
Fh	72.6	87.6	70.4	74.0	85.1	70.2
Tr	70.6	76.9	70.4	71.7	77.9	71.4
Co	62.6	65.5	55.9	63.3	68.2	53.9
Vl	78.9	73.0	79.6	80.2	77.1	78.3
Moyenne	71.7	**73.6**	70.4	73.0	**75.4**	70.4
Ecart-type	9.0	12.6	10.5	9.1	11.9	10.3

Tab. VIII.3 Résultats de classification sur les deux types de segments : transitoires "T" et non transitoires "S" avec $N_t = 2$ et $N_t = 4$, comparés aux résultats obtenus pour un système sans segmentation "R". Des différences de scores de 0.2% (respectivement 2%) sont significatives pour la configuration "R" et "S" (respectivement "T"), en considérant des intervalles de confiance à 95%. L'ensemble de test `SUB-INST-T` est utilisé.

R	Pn	Gt	Ob	Cl	Fh	Tr	Co	Vl
Pn	84.3	14.2	0.7	0.1	0.6	0.0	0.1	0.0
Gt	12.2	71.8	0.0	2.6	0.1	0.0	13.1	0.2
Ob	0.0	0.0	81.3	7.1	0.2	10.8	0.0	0.6
Cl	1.0	1.3	4.1	57.7	15.2	11.3	3.9	5.5
Fh	3.7	2.7	0.4	10.3	74.0	7.1	0.8	1.0
Tr	0.5	0.0	9.4	4.9	2.5	71.7	0.4	10.5
Co	2.6	5.2	0.4	6.8	0.4	0.2	63.3	21.0
Vl	0.0	0.1	2.1	5.4	0.6	7.7	3.8	80.2

Tab. VIII.4 Matrice de confusions relative à la classification sans segmentation avec N_t=4. Lire "ligne" confondue avec "colonne" dans x% des tests.

T4	Pn	Gt	Ob	Cl	Fh	Tr	Co	Vl
Pn	83.5	15.1	0.4	0.2	0.8	0.1	0.0	0.0
Gt	4.2	74.4	0.0	0.4	0.1	0.0	20.6	0.1
Ob	0.0	0.0	86.9	3.1	0.0	9.3	0.0	0.6
Cl	0.3	2.7	4.2	50.0	14.2	16.4	2.5	9.0
Fh	0.8	0.0	0.0	2.8	85.1	10.0	0.0	0.9
Tr	0.0	0.0	8.2	4.3	2.1	77.9	0.0	7.6
Co	1.8	4.8	0.4	6.3	0.6	0.1	68.2	17.6
Vl	0.0	0.0	1.2	3.3	0.4	15.9	1.8	77.1

Tab. VIII.5 Matrice de confusion relative à la classification sur les segments transitoires "T4" avec N_t=4.

traitement particulier des transitoires uniquement pour les problèmes bi-classes qui en tirent partie, typiquement (piano vs guitare), (cor vs clarinette), etc.

La question qui se pose à présent concerne l'utilité du traitement précédent au sein d'un système de reconnaissance profitant de longueurs de décision réalistes de 2 à 4s. Le tableau VIII.6 présente les résultats de classification obtenus avec le système de référence "R" en utilisant des fenêtres de décision de longueur $N_t = 124$ (2s). Les taux de reconnaissance sont nettement supérieurs à ceux obtenus avec des systèmes effectuant la prise de décision uniquement sur les segments de transitoires dont la longueur ne dépasse pas $N_t = 4$.

Nous poursuivons actuellement le travail sur cette problématique en explorant des stratégies qui permettraient de fusionner les décisions prises par des classificateurs "spécialisés dans les transitoires" avec d'autres spécialisés dans les non-transitoires pour parvenir à une meilleure décision sur des fenêtres de décision nominales, dans lesquelles se retrouvent des segments transitoires et non-transitoires.

VIII-3. Conclusions

Nous avons analysé, dans ce chapitre, la sortie produite par notre algorithme de sélection des attributs FSFC. Cet algorithme nous a permis de produire une organisation des attributs dans laquelle ceux qui présentent des distributions de valeurs similaires sont regroupés, et triés par

Pn	92.2
Gt	86.8
Ob	89.8
Cl	76.4
Fh	92.1
Tr	79.3
Co	71.3
Vl	88.2
Moyenne	**84.5**
Ecart-type	7.8

Tab. VIII.6 Résultats obtenus avec un système sans segmentation pouvant exploiter des fenêtres de décisions de tailles $N_t = 124$ (2s).

ordre d'efficacité pour la discrimination des instruments.

Nous avons observé que des attributs calculés dans des domaines différents, qui sont sensés caractériser des proporiétés acoustiques ou perceptuelles distinctes, se retrouvent dans les mêmes clusters. Les clusters d'attributs sont de tailles très variables. La conception de nouveaux attributs gagnerait à viser l'obtention de paramètres qui ne se placeraient pas dans les clusters de grandes tailles pour une diversité accrue de la description.

Les descripteurs spectraux sont largement représentés dans le sous-ensemble des attributs les plus efficaces. Notre nouveau descripteur d'intensité et de rapport d'intensités dans les sous-bandes en octaves s'avère utile : 3 attributs $OBSI$ et 1 attribut $OBSIR$ se retrouvent dans les premiers rangs du classement. Les attributs cepstraux sont particulièrement bien classés par l'algorithme de sélection, ils occupent les tous premiers rangs.

Nous nous sommes ensuite intéressés aux transitoires d'attaque et à l'utilité de réaliser un traitement différencié de ces éléments importants du son. Nous avons mis en évidence par des critères objectifs que les segments d'attaques, décrits de façon spécifique, permettent une meilleure discrimination de la plupart des classes d'instruments.

Pourtant il n'est pas évident que cette propriété puisse être exploitée efficacement au sein d'un système de reconnaissance automatique des instruments pouvant se permettre des prises de décision sur des fenêtres temprorelles de durées largement supérieures aux durées des attaques.

En effet, l'information portée par l'attaque se retrouve dans ce cas diluée dans l'étendue de la fenêtre de décision. En attendant de parvenir à la réalisation de stratégies permettant de "relever" l'information d'attaque, nous effectuons un traitemant non-différencié des segments.

IX. Classification hiérarchique des instruments de musique, cas mono-instrumental

Nous présentons dans ce chapitre notre système de reconnaissance des instruments à partir d'un contenu musical mono-instrumental. Nous adoptons une stratégie de classification hiérarchique basée sur une taxonomie automatique des instruments. Cette taxonomie est inférée au moyen d'algorithmes de clustering agglomératif hiérarchique exploitant des distances probabilistes (divergence et Bhattacharryyaa). Une attention particulière est portée au calcul de ces distances. Nous faisons pour cela appel à une méthode à noyau. Nous comparons les résultats de classification obtenus avec la taxonomie automatique à ceux obtenus avec la taxonomie naturelle des familles d'instruments. L'approche de sélection binaire des attributs est ensuite mise à profit pour aboutir à un schéma de classification plus performant.

IX-1. Introduction

La classification hiérarchique a été récemment utilisée avec succès pour de nombreuses tâches de classification audio, particulièrement la classification des instruments de musique [Martin, 1999, Eronen, 2001a, Peeters, 2003] et la classification du genre musical [Pachet et Cazaly, 2000, McKay et Fujinaga, 2004, Li et Ogihara, 2005]. En premier lieu, le recours à la classification hiérarchique a pour but d'améliorer les taux de reconnaissance par rapport à ceux obtenus avec les systèmes dits "plats", dans lesquels toutes les classes sont considérées sur un seul niveau, sans organisation particulière. Par ailleurs, cela permet de réaliser une scalabilité de classification par l'introduction d'étiquettes de classes plus vagues à des niveaux supérieurs de la taxonomie hiérarchique considérée par le système de classification.

Dans la plupart des travaux, des taxonomies évidentes, empruntées à d'autres domaines d'acti-

vité ont été exploitées. Les taxonomies utilisées pour la classification des instruments s'inspirent de l'organisation des familles d'instruments dûes à l'acoustique musicale et la musicologie, alors que celles qui ont été exploitées pour la classification du genre musical prennent origine dans les usages de l'industrie musicale.

De telles taxonomies présentent l'avantage d'être habituelles et intuitives, permettant ainsi une facilité d'appréhension par l'utilisateur. D'un autre côté, elles souffrent de deux inconvénients majeurs. D'abord, sur la base de l'intuition, un nombre élevé de taxonomies possibles peut être retenu, menant à des systèmes hétérogènes et à des classifications contradictoires. Ensuite, ces taxonomies ne sont pas forcément destinées à maximiser les performances de classification.

Des tentatives de réponses à ces deux problèmes ont été proposées dans des travaux précédents. Pachet & Cazaly ont proposé des directives à suivre dans la construction d'une taxonomie des genres musicaux [Pachet et Cazaly, 2000]. L'application de l'analyse MDS (Multi-Dimensional Scaling) pour observer les dissimilarités entre instruments de musique [McAdams *et al.*, 1995, Peeters, 2003] peut également être vue comme une étape importante vers la réalisation d'organisation plus "naturelles" des classes (au sens de la similarité des propriétés acoustiques de ces dernières). Plus récemment, il a été fait appel au clustering hiérarchique pour obtenir une taxonomie hiérarchique des instruments destinée à être utilisée pour la classification d'instruments "non-enregistrés", c'est-à-dire non connus à l'étape d'apprentissage [Kitahara *et al.*, 2004]. La distance de Mahalanobis a été utilisée comme critère de proximité des classes dans le processus de clustering, en faisant l'hypothèse de gaussianité des données. Dans [Li et Ogihara, 2005], une taxonomie automatique des genres musicaux a été inférée en regroupant récursivement les genres qui sont fréquemment confondus par un classificateur plat.

Nous proposons une méthode pour l'inférence de taxonomies automatiques, destinée à une utilisation au sein d'un schéma de classification hiérarchique. La méthode est proche de celle décrite dans [Kitahara *et al.*, 2004] mais elle a été développée parallèlement à ce travail et nous faisons appel à un critère de proximité des classes plus élaboré. Nous comparons les performances de classification hiérarchique basée sur cette méthode à celles réalisées par un système exploitant une taxonomie "naturelle" en familles d'instruments pour la tâche de reconnaissance automatique des instruments sur des enregistrements de solo.

Cette étude est menée sur le corpus complet `INS` (*cf.* chapitre II) comprenant 19 instruments.

IX-2. Principe de classification hiérarchique

Il s'agit d'exploiter dans le processus de classification une *taxonomie hiérarchique* des instruments [Martin, 1999]. Cette taxonomie prend la forme d'un arbre tel que celui présenté dans la figure IX.1. Les nœuds de cet arbre se composent de classes ou d'unions de classes que nous désignons par *super-classes*, regroupées selon un critère choisi (*cf.* section IX-3). Les super-classes (entourées d'une ellipse dans la figure IX.1) peuvent être subdivisées en classes (*feuilles de l'arbre* ou *nœuds de décision*), ou en d'autres super-classes (*nœuds intermédiaires*). La classification se fait de façon hiérarchique en ce sens qu'un exemple de test est d'abord classé parmi les classes des niveaux supérieurs (en partant du niveau 1), avant d'être plus précisément déterminé en traversant les noeuds de l'arbre de haut en bas jusqu'à aboutir à un nœud de décision. A chaque niveau de l'arbre, le nœud "le plus probable" est sélectionné pour être traversé, et ce en utilisant la règle de décision MAP (*cf.* section V-1-A).

Les décisions intermédiaires (aux nœuds intermédiaires) sont prises sur les fenêtres d'observation ($N_t = 1$). Ensuite, les décisions finales sont prises sur des fenêtres de décision plus longues, de 4s, regroupant la suite temporelle des décisions obtenues à la sortie de la classification hiérarchique (sur $N_t = 249$ fenêtres d'observation successives).

Notons qu'il peut être avantageux de faire appel à des techniques plus élaborées de parcours de l'arbre telles que la "recherche par faisceau" (*beam search*) dont le but est de minimiser la répercution des erreurs de classification commises aux niveaux supérieurs sur la décision finale. Cela est rendu possible par l'exploration des n_p nœuds les plus probables ($n_p > 1$) à chaque niveau de la taxonomie pour éviter que les décisions prises à un niveau $\mathcal{N}$ ne soient pondérées par le produit des probabilités de succès de la classification parmi les nœuds des niveaux précédents $\mathcal{N} - i$, $1 \leq i \leq \mathcal{N} - 1$.

La question qui se pose est : "quelle taxonomie utiliser ?" ; nous y répondons dans ce qui suit.

IX-3. Taxonomies hiérarchiques des instruments de musique

A. Taxonomie "naturelle" des instruments de musique : familles d'instruments

Différentes taxonomies ont été proposées pour la tâche de classification des instruments de musique à partir de notes isolées [Martin, 1999, Eronen, 2001a, Peeters, 2003]. Ces taxonomies suivent d'assez près l'organisation des *familles d'instruments*, essentiellement dûe à l'acoustique

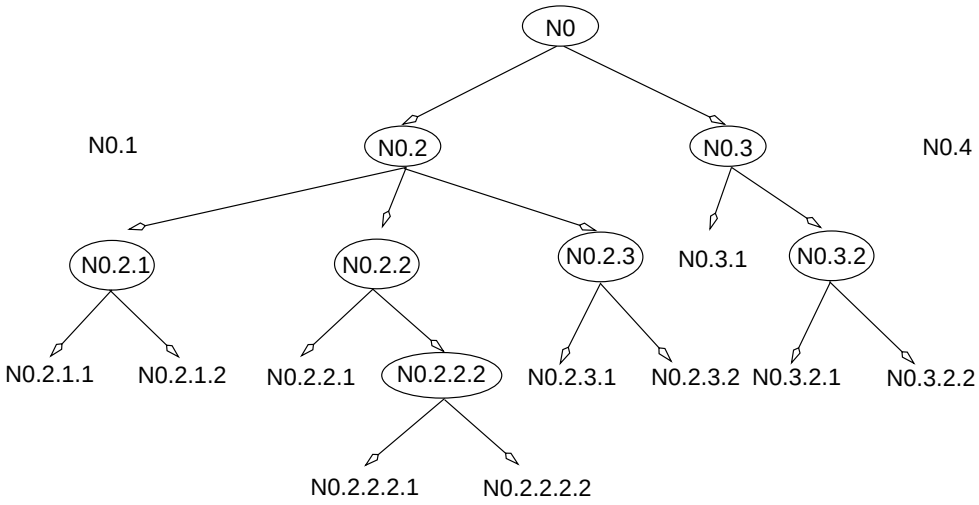

Fig. IX.1 Exemple de taxonomie hiérarchique.

instrumentale et la musicologie. Elles se déclinent de différentes façons, généralement en relation avec le mode de production des sons. Un exemple est donné dans la figure IX.2. Les déclinaisons dans cet exemple ne sont pas unanimement partagées. La famille des claviers, par exemple, ne reflète pas réellement le mode de production, si bien que le piano est généralement associé à la famille des *cordes*, précisément les *cordes frappées*, l'orgue associé à la famille des *vents* et le clavecin à la famille des *cordes pincées*. Les instruments de la famille des cordes frottées sont souvent jouées en *pizzicato*[1] et peuvent par conséquent être associés à la famille des cordes pincées.

Les taxonomies qui ont été utilisées pour la reconnaissance automatique des instruments de musique ne s'accordent pas non plus sur une organisation particulière. Si elles ont en commun d'adopter une première division des instruments en *instruments entretenus* et *instruments non entretenus*, elles font des choix différents pour d'autres regroupements, particulièrement dans la famille des instruments à vents. Nous retenons la taxonomie proposée par Peeters [Peeters, 2003] pour la construction d'un premier système de classification hiérarchique des instruments. Celle-ci est représentée dans la figure IX.3. Elle organise les instruments en rapport avec le mode de production des sons.

[1] en pinçant les cordes avec les doigts

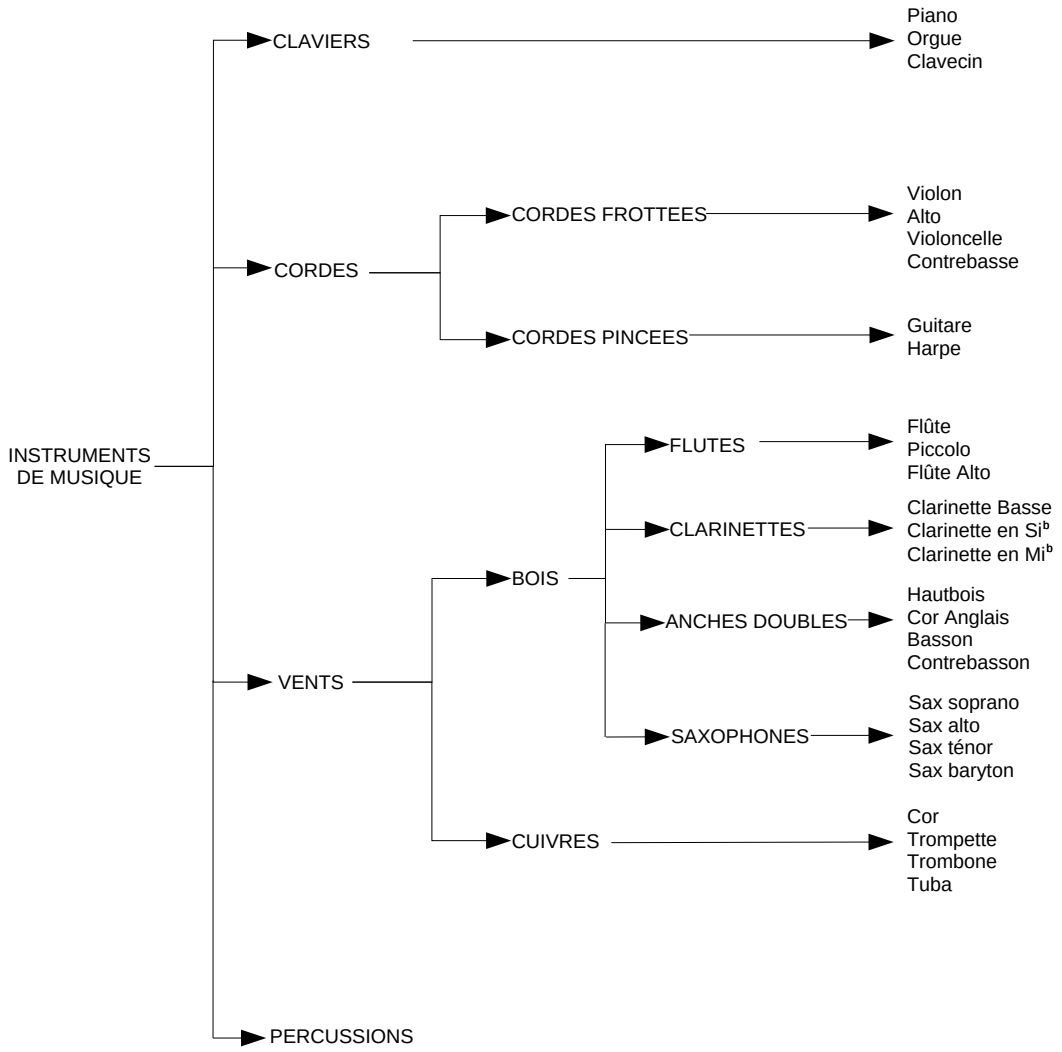

Fig. IX.2 Exemple de taxonomie hiérarchique en familles d'instruments.

Signalons que Peeters a entrepris une analyse MDS (Multi Dimensional Scaling [Duda *et al.*, 2001]) se basant sur les descripteurs utilisés afin de vérifier la pertinence de la taxonomie proposée. Cela a permis de justifier certains choix effectués mais n'a pas été utilisé pour inférer une taxonomie automatique.

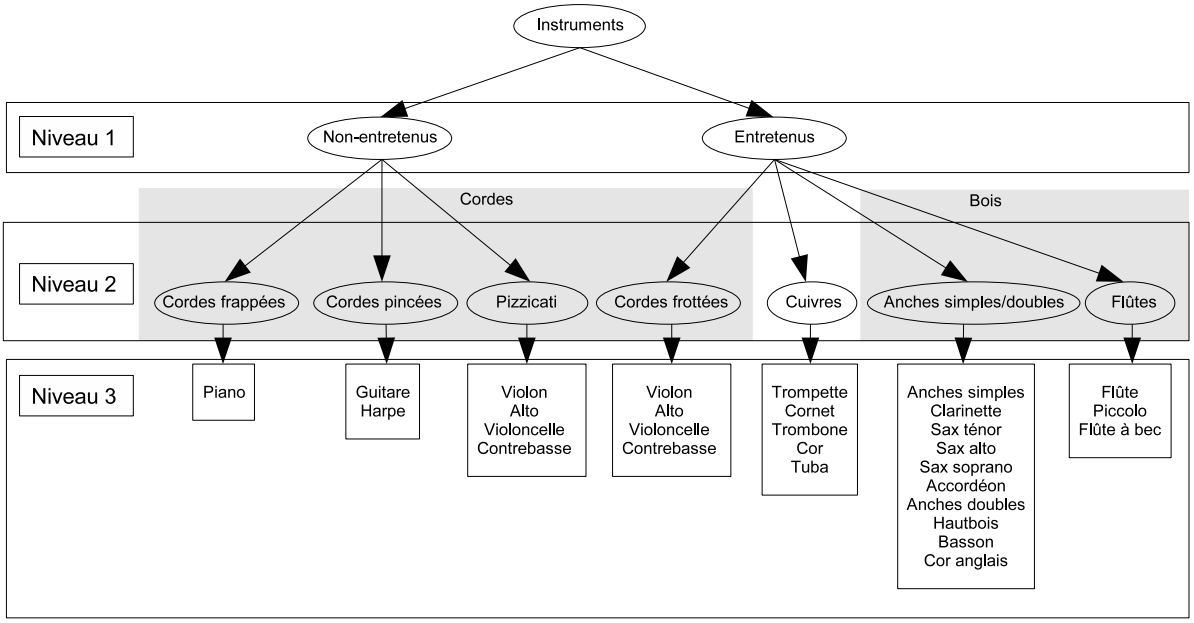

Fig. IX.3 Taxonomie hiérarchique utilisée par Peeters pour la reconnaissance des instruments à partir de notes de musique isolées [Peeters, 2003].

B. Inférence de taxonomies automatiques

Le but est d'obtenir une taxonomie hiérarchique des classes d'instruments[2] qui soit adaptée au schéma de classification envisagé. A cette fin, nous organisons les classes à l'aide de l'algorithme de clustering hiérarchique présenté dans la section V-3-A. Nous cherchons ainsi à regrouper les classes dont les observations d'attributs sont proches, au sens d'un critère adéquat, pour que les super-classes résultantes soient plus facilement discriminées.

Le choix des attributs à utiliser pour la construction de la taxonomie est fondamental. Nous utilisons la sélection de 40 attributs obtenue avec l'algorithme FSFC dans la configuration

[2]La méthode proposée ne se limite pas au problème de classification des instruments, elle peut être utilisée pour des tâches de classification très variées.

standard (non-binaire). Son utilisation pour le clustering se justifie par le fait que celle-ci est sensée être globalement efficace pour la discrimination de tous les instruments considérés. Elle participe donc de l'adaptation de la taxonomie au schéma de classification utilisé. En effet, si l'on veut que la taxonomie soit optimale pour le schéma de classification, celle-ci doit organiser les instruments dans l'espace des attributs sur lequel agissent les classificateurs. Par conséquent, elle doit dépendre fortement de la sélection d'attributs utilisée.

Comme nous l'avons précédemment signalé, le choix du critère de proximité des classes est aussi important. Il est nécessaire de recourir à des distances robustes qui permettent de limiter l'effet des observations aberrantes sur les performances de clustering. De plus, ces distances doivent être en cohérence avec l'approche de classification utilisée. Nous exploitons pour le clustering des distances probabilistes en examinant deux alternatives : la distance de Bhattacharryya et la divergence, pour retenir la distance qui produit le meilleur clustering. Cela peut être vu comme une opération de clustering des densités de probabilités des observations relatives aux différentes classes. Les données que nous traitons sont connues pour être mal approximées par des modèles mono-gaussiens, d'où le recours à une approche par noyau pour le calcul de ces distances (*cf.* section V-3-B).

Dans ce processus de calcul, il est nécessaire d'effectuer une décomposition en valeurs propres de matrices de Gram (*cf.* section V-2-D.2) de tailles $l_q \times l_q$, où l_q est le nombre d'exemples d'apprentissage de la classe Ω_q. Cette opération est coûteuse ($O(l_q^3)$) car l_q est assez grand (il peut dépasser 40,000 pour certaines classes). Par conséquent, les ensembles d'apprentissage sont divisés en sous-ensembles de 2000 observations et les distances requises sont approximées par la moyenne des distances calculées entre ces sous-ensembles. Le nombre de valeurs propres à préserver a été étudié dans des expériences préliminaires, et nous sommes restés sur deux valeurs intéressantes, à tester : $r_i = r_j = 10$ et $r_i = r_j = 20$.

Le noyau exploité est le noyau RBF gaussien. Nous utilisons la mise à l'échelle décrite dans la section VII-4 et deux valeurs de σ^2 sont testées : σ^2=0.5 et σ^2=1.

Ainsi, nous calculons les distances entre toutes les paires de classes d'instruments considérées et nous les utilisons dans le déroulement de l'algorithme de clustering agglomératif.

Il s'agit dans un premier temps de sélectionner le meilleur clustering parmi les différentes possibilités résultant de l'expérimentation de la distance de Bhattacharryyaa et de la divergence, mais aussi des différentes valeurs de σ et r_i. Nous effectuons notre choix en nous basant sur les

valeurs des coefficients cophénétiques, à maximiser, (*cf.* section V-3-A) associés à chacun des clusterings obtenus. Le tableau IX.1 résume les mesures de ces coefficients.

$r_i = r_j =$	10	20	
σ^2	0.5	0.5	1
Bhattacharryya	0.47	0.56	0.54
Divergence	0.71	**0.73**	0.69

Tab. IX.1 Coefficients cophénétiques des clusterings effectués en fonction des distances utilisées et des paramètres σ du noyau et r_i, r_j.

L'utilisation de la divergence avec $r_i = r_j = 20$ et $\sigma^2 = 0.5$ réalise le clustering le plus pertinent au regard du coefficient cophénétique (maximum dans ce cas). Nous représentons dans la figure IX.4 le dendrogramme associé à cette solution. Cette représentation sous forme d'arbre peut déjà être considérée comme une taxonomie des instruments. Néanmoins, il est plus judicieux d'élaguer l'arbre pour une meilleure efficacité de la classification. D'une part, nous avons intérêt à ne garder que les regroupements les plus pertinents (dans lesquels les classes restent "proches" les unes des autres), d'autre part, il est plus intéressant de limiter le nombre de niveaux de la taxonomie essentiellement pour limiter la complexité de la classification, mais également pour obtenir des taxonomies plus lisibles, qui soient faciles à manipuler par un utilisateur.

Nous choisissons de limiter la taxonomie à quatre niveaux (racine de l'arbre non comprise), ce qui demande trois coupes du dendrogramme puisque le dernier niveau est déduit automatiquement en développant les nœuds du niveau 3 jusqu'aux feuilles (ce qui revient à une coupe par la droite $y = 0$). Ces coupes sont visibles sur la figure IX.4. Elles donnent naissance à la taxonomie représentée dans la figure IX.5.

La taxonomie trouvée est significativement différente de celle des familles d'instruments. A l'exception de quelques regroupements habituels, par exemple l'association, au sein d'un même cluster, du piano et de la guitare, ou encore de l'alto et du violon, la plupart des autres regroupements ne correspondent pas à l'organisation en familles d'instruments.

Au premier niveau, la contrebasse jouée *con arco*[3] et la contrebasse jouée en *pizzicato* sont associées dans un même cluster avec le tuba, et le piano et la guitare sont regroupés avec

[3] avec l'archet

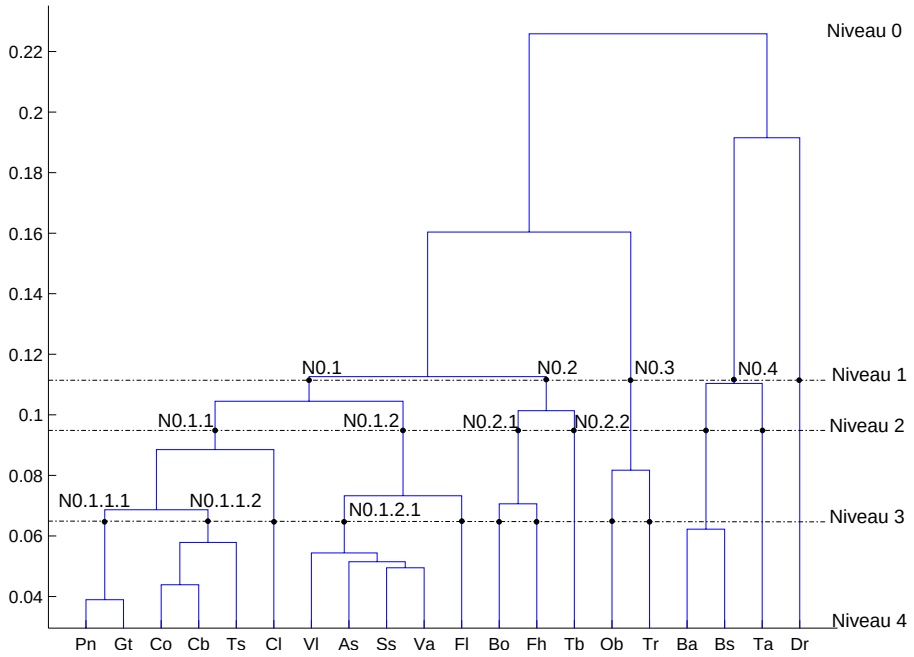

Fig. IX.4 Dendrogramme obtenu avec la divergence, σ^2=0.5 et $r_i = r_j = 20$.

la majorité des bois et des cordes frottées. Ainsi, la distinction "instruments entretenus/non-entretenus" n'a pas été considérée comme pertinente. En effet, comme cette propriété n'est pas capturée par les attributs sélectionnés, elle ne peut être "vue" par le schéma de classification, et ce n'est donc pas optimal de la prendre en compte dans la taxonomie.

La plupart des bois se retrouvent à ce niveau au sein du même cluster (noeud N0.1). Ce n'est pas le cas des cuivres qui sont dispersés dans des groupes différents. Le tuba est associé à la contrebasse, la trompette au hautbois et le cor et le trombone associés au basson.

Aux niveaux inférieurs, nous observons que le trombone se détache de la paire (basson, cor), la flûte se détache du regroupement (violon, alto, saxophone alto et saxophone soprano), et la clarinette se détache des clusters (piano, guitare) et (violoncelle, clarinette basse, saxophone ténor).

Ainsi, il apparaît, à partir des descripteurs sélectionnés, que l'information de tessiture soit dominante dans le regroupement des classes, puisque les instruments dont les tessitures se recouvrent fortement (dans la partie centrale *cf.* figure IX.6) sont assignés aux mêmes clusters. Ces regroupements ne nous surprennent pas car ils traduisent les confusions que nous observons dans les expériences de classification (nous les retrouverons dans la section IX-4) : les instruments qui sont fréquemment confondus par le système de classification se retrouvent souvent au sein

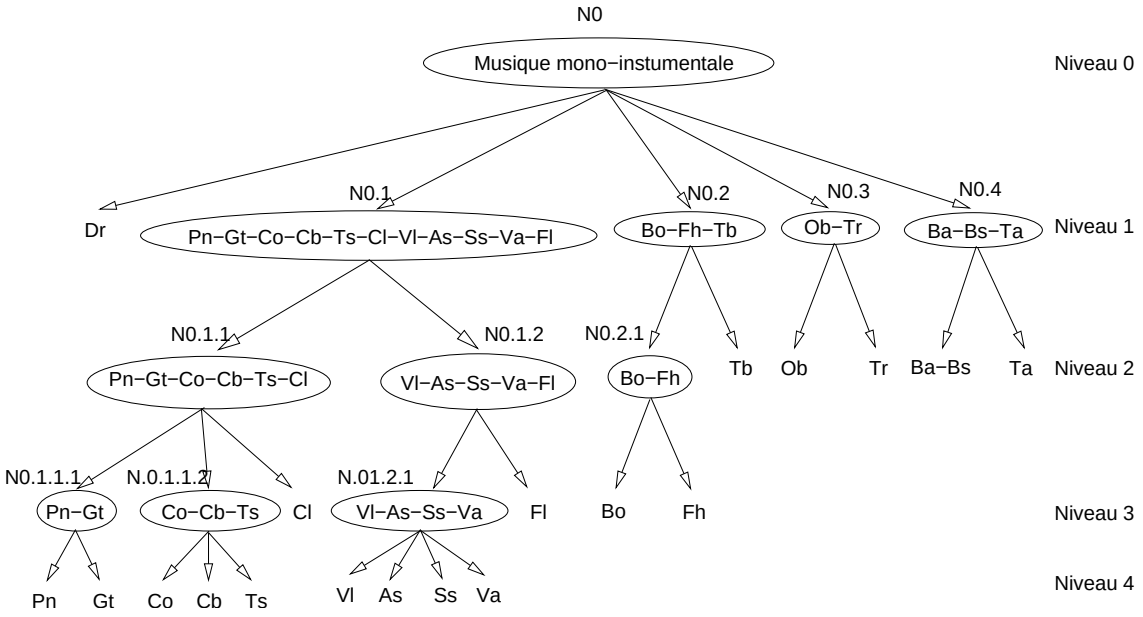

Fig. IX.5 Taxonomie générée automatiquement.

des mêmes groupes de la taxonomie.

IX-4. Système de classification non-hiérarchique de référence

Nous étudions dans un premier temps les performances d'un système de classification de référence, non-hiérarchique. Ce système exploite des modèles GMM dans une configuration standard (non binaire)[4]. Le nombre de composantes du mélange gaussien a été fixé à $M = 8$, nos tests ont en effet montré que des valeurs plus grandes de M ne permettaient pas d'améliorer les performances, au contraire celles-ci ont tendance à s'altérer.

La matrice de confusions à l'issue du test effectué sur l'ensemble `INS-T` en utilisant des fenêtres de décision de longueur 4s ($N_t = 249$) est présentée dans le tableau IX.3. Le taux de reconnaissance moyen est de 61.3%[5]. Ces taux varient de façon significative d'un instrument

[4]Un schéma de classification bi-classe ciblant les 20 classes d'instruments considérées implique l'apprentissage de 190 classificateurs binaires et l'utilisation d'autant de classificateurs dans la phase de test (sur chaque fenêtre d'observation de 32ms), ce qui représente une charge de calcul importante que nous cherchons à éviter.

[5]Les confusions entre contrebasse *con arco* et *pizzicato* ne sont pas considérées comme telles : l'instrument étant le même, nous calculons son taux de reconnaissance à partir de ceux obtenus pour Ba et Bs.

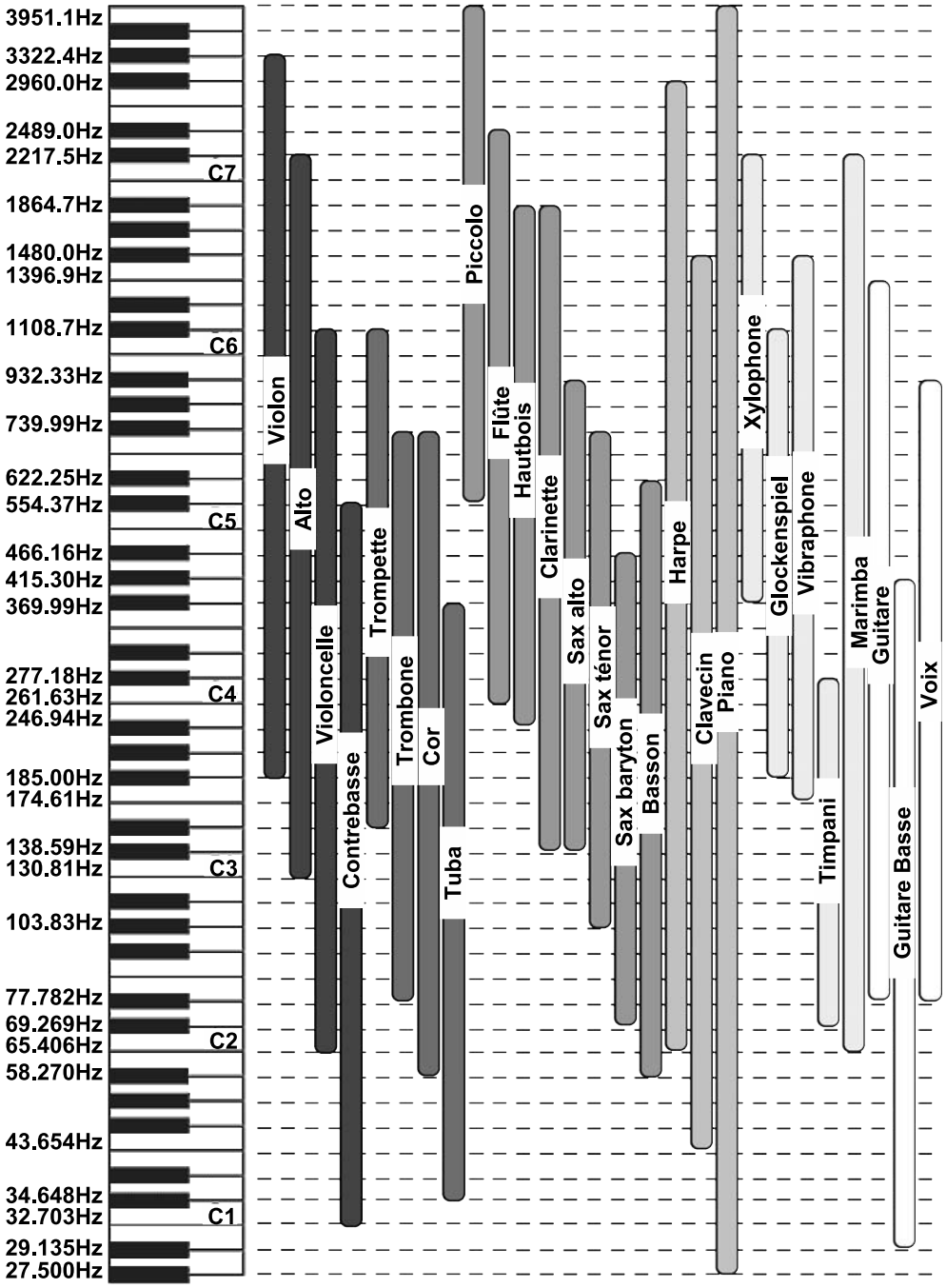

Fig. IX.6 Tessitures des instruments.

à l'autre : si des performances acceptables sont atteintes pour certains instruments (batterie : 100.0%, piano : 88.5%, cor : 88.8%), les résultats ne sont pas satisfaisants pour d'autres, par exemple les saxophones ténor et soprano (22.1% et 1.3%), la flûte (55.8%), l'alto (49.6%) et la clarinette (23.8%). Nous relevons deux types de confusions :

- des confusions au sein d'une même famille d'instruments, par exemple la clarinette est confondue avec le saxophone alto dans 29.4% des cas, l'alto avec le violon dans 38.9% des cas, le tuba avec le trombone dans 22.1% cas. De telles confusions sont prévisibles eu égard au mode de production des sons ; elles ont été rapportées dans les études précédentes sur la reconnaissance des instruments à partir de notes musicales isolées (voir [Eronen, 2001a] par exemple) ;
- des confusions entre instruments qui semblent intuitivement "éloignés" et qui n'ont pas été notées dans les études sur les notes isolées. Nous relevons par exemple que le basson est confondu avec le cor dans 24.3% des cas, le hautbois avec la trompette dans 11.2% des cas et le saxophone ténor avec le violon dans 24.9% des cas. Il est raisonnable de penser que ces confusions ont lieu du fait que ces instruments ont des tessitures qui se recouvrent fortement dans leur partie centrale.

Nous obtenons donc la confirmation que la taxonomie générée automatiquement a tendance à regrouper au sein d'une même super-classe les instruments qui sont fréquemment confondus les uns avec les autres.

IX-5. Systèmes de classification hiérarchique

A. Classification à partir d'une taxonomie naturelle

La figure IX.7 montre la restriction de la taxonomie présentée dans la section IX-3-A aux instruments considérés dans notre étude. La batterie a été insérée seule au premier niveau de la hiérarchie.

Des classificateurs SVM ont été appris pour la discrimination des classes de chaque nœud de la taxonomie, à partir de la sélection de $d = 40$ attributs obtenue par FSFC. Le paramètre C des SVM est fixé à 1 et le noyau réglé, pour chaque paire de classes possible, suivant la procédure décrite dans la section VII-4. Le nombre total de SVM apprises sur la totalité des nœuds intermédiaires (les 6 nœuds, entourés par des ellipses) est 45 : 3 à chacun des nœuds N0 et N0.1, 6 à chacun des nœuds N0.2, N0.2.1 et N0.2.2 et 21 au nœud N0.2.3.

Le noyau linéaire a été sélectionné (par les critères utilisés) uniquement au dernier niveau, pour les problèmes bi-classes : (Tr vs Ta), (Bo vs Ob) et (Vl vs Bs). Pour le noyau gaussien, la valeur de $\sigma^2 = 0.5$ a été sélectionnée la plupart du temps. Ce système de classification sera désigné par CHF[6].

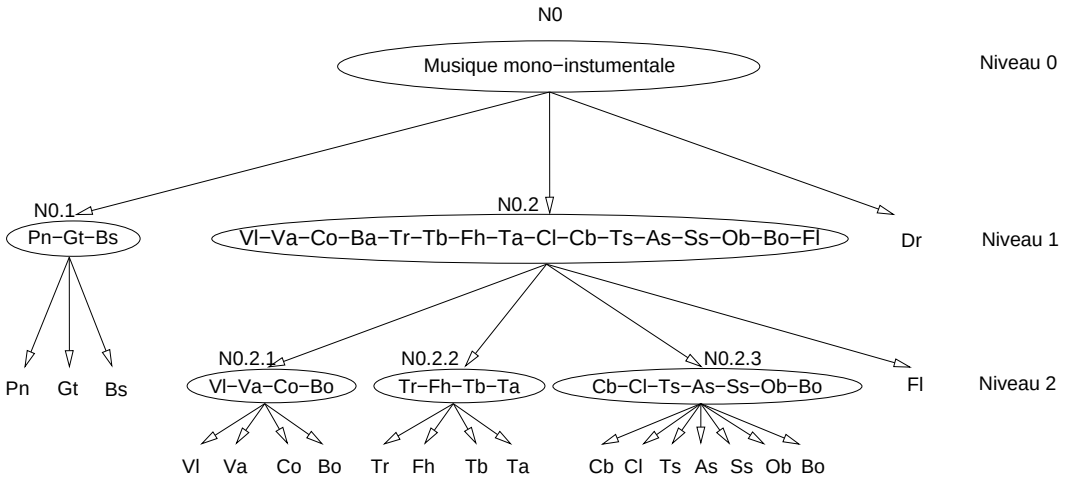

Fig. IX.7 Taxonomie hiérarchique en familles d'instruments.

Intéressons-nous aux résultats de classification finale[7]. La matrice de confusions relative à la sortie du système de classification hiérarchique de tous les instruments considérés est présenté dans le tableau IX.4. Les décisions sont prises comme pour le système de référence toutes les 4s ($N_t = 249$). Le taux de reconnaissance moyen est de 63.8%, soit une amélioration de 2.5% par rapport au système de référence. 10 instruments sur 18 sont mieux reconnus par le nouveau système. Les performances sont significativement améliorées pour le saxophone alto (+24%), la clarinette (+18.8%), la contrebasse (+18.5%), et l'alto (+11.4%). La reconnaissance des 8 instruments restant est desservie par la classification hiérarchique : nous relevons en particulier le cas du cor qui perd 33.5%. En fait, nous assistons à une redistribution des confusions qui n'est pas toujours favorable. Nous pouvons dire que le rapprochement d'instruments de la même famille ne bénéficie pas toujours à tous ces instruments. Pour les cuivres par exemple, le cor est plus fréquemment confondu avec le trombone en comparaison avec le système de référence (de 7.3% on passe à 30.6% de cas de confusion). De même le rapprochement de la guitare et

[6]Classification Hiérarchique basée sur la taxonomie des Familles d'instruments.

[7]*cf.* annexe B pour les résultats de classification aux nœuds intermédiaires.

du piano sert la reconnaissance du piano (la confusion avec la guitare passe de 9.9% des cas à 5.4%), mais il n'est pas profitable à la reconnaissance de la guitare (confondue 8.0% du temps avec le piano contre 3.4% du temps dans le système de référence).

En revanche, les confusions inter-familles se trouvent diminuées par le système hiérarchique. Ainsi, le basson est confondu maintenant avec le cor dans 10.8% (contre 24.3% pour la référence), le hautbois n'est plus confondu avec la trompette que 5.6% du temps (-5.6%) et le saxophone ténor est plus facilement discriminé du violon (-23.1% de cas de confusion).

B. Classification à partir d'une taxonomie automatique

Nous testons dans cette partie la taxonomie générée automatiquement pour la reconnaissance des instruments, dans les mêmes conditions que celles utilisées par le système de référence et le système hiérarchique précédent.

Des classificateurs SVM ont été appris pour la discrimination des classes de chaque nœud de la taxonomie, à partir de la sélection de $d = 40$ attributs obtenue par FSFC. Le paramètre C des SVM est fixé à 1 et le noyau réglé, pour chaque paire de classe possible, suivant la procédure décrite dans la section VII-4. Le nombre total de SVM apprises sur la totalité des nœuds intermédiaires (les 11 nœuds, entourés par des ellipses) est 36. La complexité est donc ici plus faible que celle du système CHF, pour lequel 45 SVM ont été nécessaires. Là aussi, la valeur de $\sigma^2 = 0.5$ a été sélectionnée la plupart du temps pour le noyau gaussien.

Ce système de classification sera désigné par CHA[8].

La matrice de confusions correspondant aux résultats de classification finale[9] est donnée dans le tableau IX.5. Les décisions sont prises comme pour les deux systèmes précédents toutes les 4s ($N_t = 249$). Le taux de reconnaissance moyen est de 64.6%, soit 3.3% de mieux que le système de référence mais uniquement 0.8% de mieux que le système hiérarchique précédent. 11 instruments sur 18 sont mieux reconnus par le système hiérarchique CHA en comparaison avec le système de référence. On retrouve globalement les mêmes cas d'amélioration qu'avec le système CHF mais ces améliorations ne se chiffrent pas aux mêmes taux : elles sont plus

[8] Classification Hiérarchique basée sur la taxonomie Automatique.

[9] *cf.* annexe B pour les résultats de classification aux nœuds intermédiaires.

importantes pour six des classes d'instruments (Pn, Gt, Va, Tr, Ss et Fl) et moins importantes dans trois cas (Ob, As et Tb). Le nouveau système hiérarchique permet par exemple de réduire plus efficacement les confusions de la flûte avec le hautbois et la clarinette : elles ne se chiffrent plus qu'à 4.1% et 9.6%, respectivement, contre 11.3% et 18.5% pour CHF et 15.0% et 16.8% pour la référence. Nous assistons là aussi à une redistribution des confusions qui n'est pas toujours favorable. Des cas de confusions résolues par le système CHF sont moins bien traitées par CHA, par exemple (hautbois vs trompette), (trombone vs saxophone ténor), etc. Nous remarquons que les confusions entre instruments qui se retrouvent dans les mêmes clusters de la taxonomie ne sont pas toujours atténuées : par exemple le basson est fréquemment confondu avec le cor (24.8% du temps contre 10.8% avec CHF) et le saxophone ténor est plus fréquemment assigné à la classe violoncelle (10.1% du temps contre 2.5% pour le système de référence et 7.7% avec CHF).

C. Récapitulation des performances des différents systèmes

Le tableau IX.2 récapitule les taux de reconnaissance obtenus avec les deux systèmes hiérarchiques testés, en comparaison avec ceux du système de référence. Les deux systèmes hiérarchiques permettent d'atteindre des performances moyennes supérieures à celles du système de référence. La taxonomie automatique donne lieu à des résultats de classification en moyenne supérieurs à ceux atteints par la taxonomie des familles d'instruments. Le système CHA permet d'identifier avec plus de succès 11 instruments sur 18, en comparaison avec le système de référence, contre 10 sur 18 pour le système CHF.

Cependant, la différence entre les performances moyennes des deux systèmes hiérarchiques reste faible. Comme nous l'avons vu, la classification hiérarchique ne permet pas dans tous les cas d'atténuer les confusions entre instruments regroupés au sein d'un même cluster.

Dans les systèmes hiérarchiques précédents nous avons utilisé la même sélection d'attributs que pour le système de référence et ce à tous les niveaux de la taxonomie. Nous allons maintenant mettre en évidence l'apport de la sélection binaire des attributs.

IX-6. Utilisation de l'approche de sélection binaire des attributs

Nous effectuons une sélection binaire des attributs à chaque niveau de l'arbre de classification CHA : un sous-ensemble optimal d'attributs est obtenu pour chaque paire de classes d'un niveau

	Ref.	Familles	Automatique
Pn	88.5	93.9	**95.2**
Gt	77.0	74.5	**77.3**
Bo	50.3	**57.9**	43.6
Ob	84.3	**91.3**	88.2
Cl	23.8	**42.6**	39.4
Fh	**88.8**	55.3	64.4
Tr	73.8	71.0	**74.1**
Co	**64.3**	58.0	59.0
Vl	**71.7**	66.6	70.2
Ba-Bs	75.2	**93.7**	**93.5**
As	71.0	**95.0**	93.9
Ts	**22.1**	18.7	18.1
Ss	1.3	8.4	**9.4**
Fl	55.8	65.5	**77.9**
Tb	66.4	**69.5**	67.7
Ta	**38.7**	34.6	37.9
Va	49.6	61.0	**61.6**
Dr	**100.0**	91.2	90.7
Moyenne	61.3	63.8	**64.6**

Tab. IX.2 Récapitulation des performances des différents systèmes.

donné, en faisant appel à l'approche FSFC. Par exemple, au nœud N0.1.1, 3 sous-ensembles d'attributs optimaux sont recherchés pour les trois problèmes bi-classes (Pn-Gt vs Co-Cb-Ts), (Pn-Gt vs Cl) et (Co-Cb-Ts vs Cl). Pour chaque paire de classe $\{\Omega_p, \Omega_q\}$ d'un nœud N_x donné, nous envisageons en fait trois sélections d'attributs possibles :

- une sélection de $d = 40$ attributs, $\mathcal{E}_{p,q}^{40}$;
- une sélection de $d = 20$ attributs, $\mathcal{E}_{p,q}^{20}$;
- la sélection globale (non-binaire) de $d = 40$ utilisée dans les systèmes précédents, $\mathcal{E}_1^{40}$.

La meilleure sélection parmi les trois précédentes est utilisée pour la construction de la SVM relative à la paire $\{\Omega_p, \Omega_q\}$ du nœud N_x, en exploitant les mêmes critères que ceux qui sont utilisés pour régler les paramètres des SVM, à savoir la dimension VC ou l'erreur $\xi\alpha$ (*cf.* chapitre V). En d'autres termes, pour chaque paire $\{\Omega_p, \Omega_q\}$ du nœud N_x, nous procédons à l'apprentissage de 12 SVM correspondant aux variations possibles du noyau (linéaire et gaussien avec $\sigma^2 \in \{0.2, 0.5, 1\}$) et de la sélection d'attributs ($\mathcal{E}_{p,q}^{40}$, $\mathcal{E}_{p,q}^{20}$ et $\mathcal{E}_1^{40}$) pour retenir la meilleure configuration (meilleure sélection d'attributs et meilleur noyau) au sens des critères considérés.

Le détail des sous-ensembles d'attributs sélectionnés spécifiquement pour chaque paire de classes possible peut être consulté sur Internet [Essid, a].

Nous effectuons alors un nouveau test dans les mêmes conditions que précédemment. La matrice de confusions finale[10] est présentée dans le tableau IX.6. Une amélioration des performances moyennes a été obtenue par rapport au sytème CHA non muni de la sélection binaire des attributs. 11 instruments sur 18 tirent partie de l'approche de sélection binaire. Le taux de reconnaissance moyen est de 66.4%, soit une amélioration de 2% par rapport au système CHA non muni de la sélection binaire des attributs et une amélioration de 5.1% par rapport au système de référence.

Quatre instruments gagnent plus de 5% en taux de reconnaissance, il s'agit du basson (+16.6%), du violon (+10.2%), de la flûte (+9.3%) et de la trompette (+5%). La flûte est moins fréquemment confondue avec pratiquement tous les instruments. Le basson est deux fois moins fréquemment confondu avec le cor (de 24.8%, on passe à 10.4% du temps). La confusion du violon avec l'alto ne se produit plus que dans 1.3% des tests. Le violon reste plus fréquemment confondu avec le saxophone alto qu'avec l'alto (dans 6.8% des tests contre 11.9% avec le système CHA non binaire).

[10] *cf.* annexe B pour les résultats de classification aux nœuds intermédiaires.

La reconnaissance du saxophone ténor et du saxophone soprano reste problématique (seulement 19.6% et 6.8% de taux de reconnaissance, respectivement). Cela est dû en partie au fait qu'il est difficile de distinguer les différents saxophones. La plupart des études sur la reconnaissance des instruments, particulièrement à partir de phrases musicales, ne cherchent pas en fait à distinguer les différents saxophones. Dans notre cas le taux de reconnaissance moyen de l'instrument saxophone (indépendamment de la tessiture) est de 55.2%, et le taux de reconnaissance moyen (pour tous les instruments) devient alors 70.7%.

	Dr	Pn	Gt	Bs	Ba	Co	Va	Vl	Ta	Tb	Fh	Tr	Bo	Ts	As	Ss	Fl	Ob	Cl	Cb
Dr	100.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Pn	0.0	88.5	9.9	0.2	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.3	0.0	0.0
Gt	2.2	3.4	77.0	0.0	1.7	7.1	1.7	0.0	0.0	0.0	0.0	0.0	0.0	6.5	0.0	0.0	0.0	0.0	0.0	0.3
Bs	0.0	0.0	2.3	74.1	19.3	0.0	0.0	0.0	3.5	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.8
Ba	0.1	0.0	3.6	10.9	76.7	5.8	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	2.8	0.0
Co	0.0	0.1	4.1	0.4	1.7	64.3	7.1	9.3	0.0	0.0	0.5	0.0	0.0	2.5	0.9	1.6	0.1	0.3	4.4	2.8
Va	0.0	0.0	0.0	0.0	0.0	0.0	49.6	38.9	0.0	0.0	0.0	0.0	0.0	1.3	0.8	6.7	0.0	0.0	2.6	0.2
Vl	0.0	0.0	0.0	0.0	0.0	0.3	5.0	71.7	0.0	0.0	0.0	9.2	0.0	0.0	4.8	6.0	3.0	0.0	0.0	0.0
Ta	0.0	0.0	2.1	6.3	0.0	0.0	0.0	0.0	38.7	22.1	11.1	0.0	5.2	2.6	0.9	0.0	0.0	0.0	11.0	0.0
Tb	0.0	2.4	0.8	0.0	0.0	0.0	0.0	0.3	0.0	66.4	0.3	2.2	1.1	14.1	12.3	0.0	0.0	0.0	0.0	0.0
Fh	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	7.3	88.8	0.0	2.8	0.0	0.1	0.0	0.0	0.0	0.8	0.0
Tr	0.0	0.0	0.0	0.0	0.0	0.0	0.0	11.9	0.0	4.0	0.0	73.8	0.0	0.0	0.0	0.6	0.0	6.2	3.5	0.0
Bo	0.0	2.5	1.7	0.0	0.0	0.0	0.8	0.0	0.0	0.3	24.3	0.0	50.3	0.2	2.3	0.0	6.0	0.0	11.7	0.0
Ts	0.4	0.9	1.2	0.0	0.0	2.5	6.0	24.9	0.0	0.5	1.5	2.8	0.0	22.6	34.8	0.0	0.8	0.0	0.0	1.0
As	0.0	0.0	1.7	0.0	0.0	0.0	8.2	0.0	0.0	0.0	3.1	0.5	0.8	13.4	71.0	0.7	0.0	0.0	0.5	0.0
Ss	0.0	0.0	0.0	0.0	0.0	0.0	0.0	3.8	0.0	4.8	6.9	7.4	11.8	1.5	26.2	1.3	29.8	0.0	6.5	0.0
Fl	0.0	0.0	0.4	0.0	0.0	0.0	1.6	4.6	0.0	0.0	0.0	0.3	0.0	0.0	1.6	3.8	55.8	15.0	16.8	0.0
Ob	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.2	0.0	11.2	0.0	0.0	0.0	0.4	2.1	84.3	1.8	0.0
Cl	0.0	0.0	0.0	0.0	0.0	0.2	1.7	0.0	0.0	9.2	12.6	1.5	0.0	1.8	29.4	11.9	4.1	1.2	23.8	2.6

Tab. IX.3 Matrice de confusions pour le système de référence. Fenêtre de décision de 4s.

	Dr	Pn	Gt	Bs	Ba	Co	Va	Vl	Ta	Tb	Fh	Tr	Bo	Ts	As	Ss	Fl	Ob	Cl	Cb
Dr	91.2	0.0	0.0	0.0	0.0	8.8	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Pn	0.0	93.9	5.1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0	0.0
Gt	1.4	8.0	74.5	0.4	1.1	12.4	0.8	0.0	0.1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0
Bs	0.0	0.0	2.5	69.7	23.2	0.0	0.0	0.0	1.1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.2	3.1
Ba	0.0	0.1	0.0	13.2	81.8	1.6	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	3.2	0.0
Co	0.0	0.0	1.6	0.7	0.0	58.0	23.1	7.6	0.0	0.0	0.0	0.0	0.0	1.9	0.8	3.0	0.0	0.0	1.6	1.3
Va	0.0	0.0	0.0	0.0	0.0	1.0	61.0	25.3	0.0	0.0	0.0	0.0	0.0	2.1	5.3	4.2	0.0	0.0	0.8	0.0
Vl	0.0	0.0	0.0	0.0	0.0	0.0	7.8	66.6	0.0	0.0	0.0	2.1	0.0	0.0	13.2	9.4	0.4	0.0	0.1	0.0
Ta	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	34.6	27.5	7.4	0.0	0.1	0.0	21.7	0.0	0.0	0.0	8.4	0.0
Tb	0.0	2.4	0.0	0.0	0.0	0.3	0.0	0.2	0.0	69.5	0.0	0.5	0.0	0.0	26.1	0.0	0.0	0.7	0.0	0.0
Fh	0.0	0.0	0.1	0.1	0.0	0.0	0.0	0.1	2.6	30.6	55.3	0.9	0.0	0.0	6.9	2.5	0.0	0.1	0.0	0.5
Tr	0.0	0.0	0.0	0.0	0.0	0.0	0.0	10.7	0.0	2.7	0.0	71.0	0.0	0.2	1.3	0.0	0.0	9.9	4.0	0.0
Bo	0.0	0.7	0.2	0.0	0.0	0.0	0.0	0.0	2.0	5.6	10.8	0.1	57.9	1.1	8.1	3.0	3.4	0.0	6.9	0.0
Ts	0.0	0.0	0.0	0.0	0.0	7.7	7.7	1.6	0.0	0.1	1.9	5.4	0.0	18.7	53.8	0.5	0.0	1.3	1.2	0.0
As	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.3	0.0	0.0	0.0	0.8	0.0	3.7	95.0	0.0	0.0	0.0	0.0	0.0
Ss	0.1	0.1	0.0	0.1	0.0	0.1	0.0	1.3	0.1	7.7	10.6	1.0	20.4	4.5	20.4	8.4	13.8	0.1	11.3	0.1
Fl	0.0	0.0	0.4	0.0	0.0	0.2	0.3	1.6	0.0	0.0	0.3	0.0	0.0	0.5	1.2	0.1	65.5	11.3	18.5	0.0
Ob	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.4	0.0	5.6	0.0	0.0	0.0	0.0	0.0	91.3	2.5	0.0
Cl	0.0	0.0	0.0	0.0	0.0	0.1	1.2	0.0	0.0	1.1	0.0	0.6	0.0	0.3	43.8	0.3	0.1	8.2	42.6	1.4

Tab. IX.4 Matrice de confusions pour le système de classification hiérarchique basé sur la taxonomie des familles d'instruments.

	Dr	Pn	Gt	Bs	Ba	Co	Va	Vl	Ta	Tb	Fh	Tr	Bo	Ts	As	Ss	Fl	Ob	Cl	Cb
Dr	90.7	0.0	0.0	0.0	0.0	9.2	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Pn	0.0	95.2	2.2	0.0	2.2	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.2	0.0	0.0
Gt	0.0	7.3	77.3	0.0	0.9	13.6	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.7	0.0
Bs	0.0	0.0	0.8	0.0	98.8	0.0	0.0	0.0	0.3	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.1
Ba	0.0	1.9	0.0	0.0	93.5	2.5	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.9	0.0
Co	0.0	0.0	1.0	0.0	1.0	59.0	21.1	9.2	0.0	0.0	0.0	0.0	0.0	1.7	0.6	4.3	0.0	0.0	0.6	1.1
Va	0.0	0.0	0.0	0.0	0.0	0.0	61.6	26.7	0.0	0.0	0.0	0.0	0.0	0.9	4.9	5.7	0.0	0.0	0.0	0.0
Vl	0.0	0.0	0.0	0.0	0.0	0.0	6.8	70.2	0.0	0.0	0.0	0.8	0.0	0.0	11.9	9.7	0.2	0.0	0.0	0.0
Ta	0.0	0.0	0.0	0.0	1.1	0.0	0.0	0.0	37.9	25.8	11.3	0.0	0.0	0.0	23.6	0.0	0.0	0.0	0.0	0.0
Tb	0.0	5.1	0.0	0.0	0.0	0.6	0.0	1.5	0.0	67.7	0.0	0.4	0.0	2.3	22.0	0.3	0.0	0.0	0.0	0.0
Fh	0.1	0.1	1.4	0.1	0.1	0.1	0.6	0.0	0.1	21.9	64.4	0.1	0.1	0.0	7.0	2.8	0.7	0.1	0.4	0.1
Tr	0.0	0.0	0.0	0.0	0.0	0.0	0.0	11.7	0.0	2.0	1.1	74.1	0.0	0.3	1.5	0.0	0.0	5.2	3.8	0.0
Bo	0.1	2.6	3.7	0.0	0.1	0.0	0.5	0.0	0.0	3.4	24.8	0.0	43.6	0.0	5.5	1.7	4.5	0.0	9.3	0.0
Ts	0.0	0.0	0.0	0.0	0.0	10.1	5.6	3.8	0.0	0.0	2.1	2.6	0.0	18.1	55.2	0.7	0.1	0.0	1.5	0.0
As	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.2	0.0	5.5	93.9	0.0	0.0	0.0	0.2	0.0
Ss	0.1	0.0	0.1	0.0	0.0	0.0	0.0	2.7	0.0	7.1	14.5	0.0	8.5	4.4	22.0	9.4	20.6	0.0	10.2	0.0
Fl	0.0	0.0	2.9	0.0	0.0	0.0	1.4	1.6	0.0	0.0	0.0	0.0	0.0	0.0	0.6	1.6	77.9	4.1	9.6	0.0
Ob	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.6	0.0	0.0	0.0	6.6	0.0	0.0	0.0	0.7	1.3	88.2	2.4	0.0
Cl	0.0	0.0	0.1	0.0	0.0	0.3	2.6	0.0	0.0	0.0	0.6	0.0	0.0	3.8	43.6	3.9	0.4	0.6	39.4	4.3

Tab. IX.5 Matrice de confusions pour le sytème de classification hiérarchique basé sur la taxonomie automatique.

	Dr	Pn	Gt	Bs	Ba	Co	Va	Vl	Ta	Tb	Fh	Tr	Bo	Ts	As	Ss	Fl	Ob	Cl	Cb
Dr	92.9	0.0	0.0	0.0	0.0	6.9	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Pn	0.0	97.2	0.4	0.0	2.3	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.1	0.0	0.0
Gt	0.2	3.2	82.1	0.0	0.9	11.3	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.2	0.0	0.0	0.0	0.0	1.0	0.0
Bs	0.0	0.0	1.5	0.0	97.7	0.0	0.0	0.0	0.7	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.1
Ba	0.0	2.0	0.4	0.0	94.7	1.5	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.3	0.0
Co	0.0	0.0	1.4	0.0	0.8	57.8	23.3	12.8	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.1	0.0	0.0	0.6	2.0
Va	0.0	0.0	0.0	0.0	0.0	0.0	60.7	35.2	0.0	0.0	0.0	0.0	0.0	0.9	0.8	2.2	0.0	0.0	0.0	0.0
Vl	0.0	0.0	0.0	0.0	0.0	0.0	1.3	80.4	0.0	0.0	0.0	0.9	0.0	0.0	6.8	10.1	0.3	0.0	0.0	0.0
Ta	0.0	0.0	0.0	0.0	5.0	0.0	0.0	0.0	38.1	27.5	4.1	0.0	0.0	0.0	25.1	0.0	0.0	0.0	0.0	0.0
Tb	0.0	5.0	0.1	0.0	0.0	0.0	0.0	1.3	0.0	67.4	0.0	0.5	0.0	2.1	23.1	0.0	0.3	0.0	0.0	0.0
Fh	0.1	0.1	0.2	0.1	0.1	0.1	4.5	0.1	0.1	29.8	54.9	0.1	2.0	0.1	3.9	0.1	1.9	0.1	1.7	0.1
Tr	0.0	0.0	0.0	0.0	0.0	0.0	0.0	8.0	0.0	3.7	0.0	79.1	0.0	0.3	0.0	0.0	0.0	2.7	6.1	0.0
Bo	0.0	3.1	2.5	0.0	0.0	0.0	0.0	0.0	0.0	0.6	10.7	3.2	60.2	0.0	8.2	0.0	4.8	0.0	6.5	0.0
Ts	0.9	0.0	0.1	0.0	0.0	6.0	3.0	35.0	0.0	0.1	1.4	4.2	0.4	19.6	24.9	0.1	1.7	0.1	1.3	1.1
As	0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.6	0.0	0.0	0.0	0.0	0.0	5.4	92.9	0.0	0.0	0.0	0.0	0.0
Ss	0.0	0.1	0.0	0.0	0.0	0.0	0.0	3.4	0.0	4.9	17.9	0.1	8.7	2.4	24.6	6.8	20.2	0.0	10.8	0.0
Fl	0.0	0.0	3.0	0.0	0.0	0.0	0.0	0.7	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.1	87.1	0.1	8.9	0.0
Ob	0.0	0.0	0.0	0.0	0.0	0.0	0.0	2.2	0.0	0.0	0.0	14.9	0.0	0.0	0.0	0.0	0.0	80.2	2.5	0.0
Cl	0.0	0.0	0.0	0.1	0.0	0.0	7.8	0.3	0.0	0.0	0.6	0.0	0.0	0.0	38.3	0.1	5.0	0.5	41.8	5.3

Tab. IX.6 Matrice de confusions du sytème de classification hiérarchique basé sur la taxonomie automatique et la sélection binaire des attributs.

IX-7. Conclusions

Dans ce chapitre, nous avons analysé et comparé les performances de deux taxonomies hiérarchiques pour la tâche de la reconnaissance automatique des instruments de musique sur des enregistrements mono-instrumentaux : la première est inspirée de l'organisation "naturelle" des familles d'instruments, et la seconde a été inferrée automatiquement par clustering hiérarchique.

Ces deux taxonomies cherchent à regrouper les instruments ayant des caractéristiques acoustiques et perceptuelles similaires au sein de mêmes clusters. La taxonomie des familles des instruments réalise cela, de façon intuitive, en se basant sur des propriétés des instruments déduites des études en acoustique musicale et en musicologie. La taxonomie automatique, quant à elle, exploite pour cela des mesures objectives, matérialisées par les attributs sélectionnés pour la classification, et utilise celles-ci dans le calcul des proximités entre classes.

En utilisant des classificateurs SVM et une sélection globale des attributs, nous avons trouvé qu'avec la taxonomie automatique nous obtenons des performances légèrement supérieures à celles permises par la taxonomie naturelle. Cependant, l'analyse des matrices de confusions relatives aux deux systèmes, suggère que ces deux taxonomies peuvent être critiquées. En effet, nous avons observé que lorsque des instruments qui sont difficiles à distinguer sont regroupés par la taxonomie au sein des mêmes **nœuds de décision**, aux premiers niveaux de la taxonomie (c'est le cas, par exemple, pour le basson et le cor dans la taxonomie automatique), ils ne sont pas, dans la plupart des cas, mieux reconnus.

Ce résultat contredit l'hypothèse selon laquelle les classes "ressemblantes" doivent être systématiquement regroupées dans le processus de construction de la taxonomie. Il semble plus avantageux de regrouper ces classes aux niveaux supérieurs de la taxonomie, mais de les "éloigner" (en les positionnant dans des nœuds différents) aux niveaux inférieurs, où les décisions finales sont prises. C'est ce qui est réalisé dans la taxonomie automatique pour la paire (flûte vs clarinette). Ces deux instruments, qui ne sont déterminés qu'au niveau 3, sont regroupés au sein du même nœud N0.2 au niveau 1, mais dispersés dans les nœuds N0.1.2 et N0.1.1 au niveau 2. En conséquence, ils sont beaucoup moins fréquemment confondus.

Ensuite, nous avons étudié l'apport d'une sélection des attributs plus contextuelle, en faisant appel à l'approche binaire. Des améliorations significatives en termes de performances de classification ont ainsi été obtenues. Le taux de reconnaissance moyen de ce système est de 66.4%

en prenant des décisions toutes les 4s. Le taux de reconnaissance atteint 70.7% si les différents saxophones ne sont pas distingués. Nous rappelons, à titre indicatif, que les tests réalisés sur la reconnaissance par l'Homme des instruments, en utilisant des extraits de 10s de musique de 27 instruments [Martin, 1999] rapportent des taux de reconnaissance de 67%.

Nous ne pouvons malheureusement pas nous comparer directement à ces résultats (à cause des longueurs de décision et du nombre d'instruments différents). Nous tâcherons donc de réaliser des tests de perception sur un sous-ensemble d'extraits de notre base de test. Les résultats de ces tests seront prochainement communiqués.

X. Reconnaissance des instruments à partir d'extraits de musique multi-instrumentale

Nous présentons dans ce chapitre notre système de reconnaissance des instruments de musique en contexte multi-instrumental. Au moment où ce système a été développé nous n'avions pas encore expérimenté tous les descripteurs présentés au chapitre IV ni exploré tous les algorithmes de sélection d'attributs comparés au chapitre VI. Par conséquent, ne sont utilisés ici que 355 attributs parmi les 543 examinés au final (ils seront spécifiés dans la section X-2-B). De plus, l'approche de sélection FSFC n'est pas exploitée puisqu'elle n'a été développée que tardivement. Des choix de paramètres effectués à l'époque (tels que le nombre d'attributs sélectionnés et leur normalisation, les paramètres des SVM ou la longueur de décision) ne correspondent pas à ceux que nous préconisons à la lumière des dernières expériences effectuées.

L'architecture proposée a fait l'objet d'un article de revue [Essid *et al.*, 2006a] dont nous proposons ici un résumé. Nous pensons que de meilleures performances que celles qui ont été publiées à l'époque peuvent être atteintes en exploitant les récents développements.

Nous commençons par une description de l'approche adoptée et nous présentons ensuite une synthèse des résultats expérimentaux obtenus avant de proposer des conclusions. Pour plus de détails nous invitons le lecteur à consulter l'article correspondant (*cf.* annexe C).

X-1. Description du système proposé

L'idée de départ est d'identifier tous les mélanges ou combinaisons d'instruments pouvant être joués simultanément à un instant donné de la pièce musicale traitée. Dans ce schéma, les classes peuvent être, par exemple, piano, (piano+contrebasse), (piano+contrebasse+batterie), (batterie+contrebasse), etc. Immédiatement se pose le problème de la combinatoire élevée qui

en résulte. A titre d'exemple, en restreignant l'univers des instruments à seulement 10 possibles, pour des orchestrations variant des solos aux quartets, en théorie le nombre de combinaisons possible atteint déjà $C_{10}^1 + C_{10}^2 + C_{10}^3 + C_{10}^4 = 595$. Évidemment, un système devant tester un nombre aussi élevé de classes (potentiellement encore plus élevé pour un nombre d'instruments et d'orchestrations plus important) avant de parvenir à une décision, peut difficilement être mis en œuvre en pratique. La question qui se pose est alors : Comment un système ciblant la classification de mélanges d'instruments peut-il être viable ?

D'abord, la réduction de la complexité du système doit essentiellement concerner la procédure de test. En effet, des procédures d'apprentissage de complexité élevée peuvent être tolérées puisqu'elles sont sensées être effectuées "une fois pour toutes" dans des laboratoires disposant de ressources importantes de calcul, alors que le test doit rester assez "léger" pour être supporté par l'équipement des utilisateurs finaux.

Ensuite, même si en théorie toutes les combinaisons d'instruments sont possibles, certaines de ces combinaisons sont particulièrement rares en musique. Il est évident que le choix de l'orchestration constitue l'un des degrés de liberté du compositeur. Cependant, si une large variété d'orchestrations est utilisée dans la musique contemporaine (en particulier en classique et jazz), il est clair que la majorité des formations du type trio et quartet utilisent des orchestrations typiques, en rapport avec le genre musical. Par exemple, en jazz, les trios typiques se composent d'une guitare ou d'un piano, d'une contrebasse et d'une batterie ; les quartets font intervenir un piano ou une guitare, une contrebasse, une batterie et un instrument à vent ou une voix chantée... Dans une vaste majorité de genres musicaux, chaque instrument, ou groupe d'instruments, joue un rôle typique, en relation avec le rythme, l'harmonie ou la mélodie. Clairement, les pièces de jazz faisant intervenir le piano, la contrebasse et la batterie sont beaucoup plus probables que des pièces qui mettraient en scène violon et saxophone ténor sans aucun autre accompagnement, ou des duos d'alto et de hautbois... Par conséquent, des mélanges aussi rares peuvent être raisonnablement éliminés de l'ensemble des classes possibles (optionellement) ou inclus dans une classe "divers".

Même si l'on considère que les orchestrations les plus courantes, le nombre de combinaisons possibles reste élevé. L'idée clé est d'exploiter une taxonomie hiérarchique qui regroupe les mélanges d'instruments présentant des caractéristiques acoustiques similaires au sein de super-classes (constituant les niveaux élevés de la taxonomie). Nous définissons ainsi un schéma de classification hiérarchique fonctionnant sur le même principe que celui présenté dans le chapitre

IX, dans lequel le nombre de classes possibles à un niveau donné de la hiérarchie se trouve réduit (par rapport au nombre total de mélanges possibles).

Cette taxonomie doit donner lieu à de bonnes performances de classification et dans la mesure du possible être "lisible" afin qu'un nombre maximum de super-classes présentent des étiquettes qui puissent être facilement formulées par l'utilisateur. De cette façon, une classification "grossière" (s'arrêtant à des niveaux intermédiaires de la taxonomie) demeure utile.

Un diagramme en blocs du système proposé est donné dans la figure X.1.

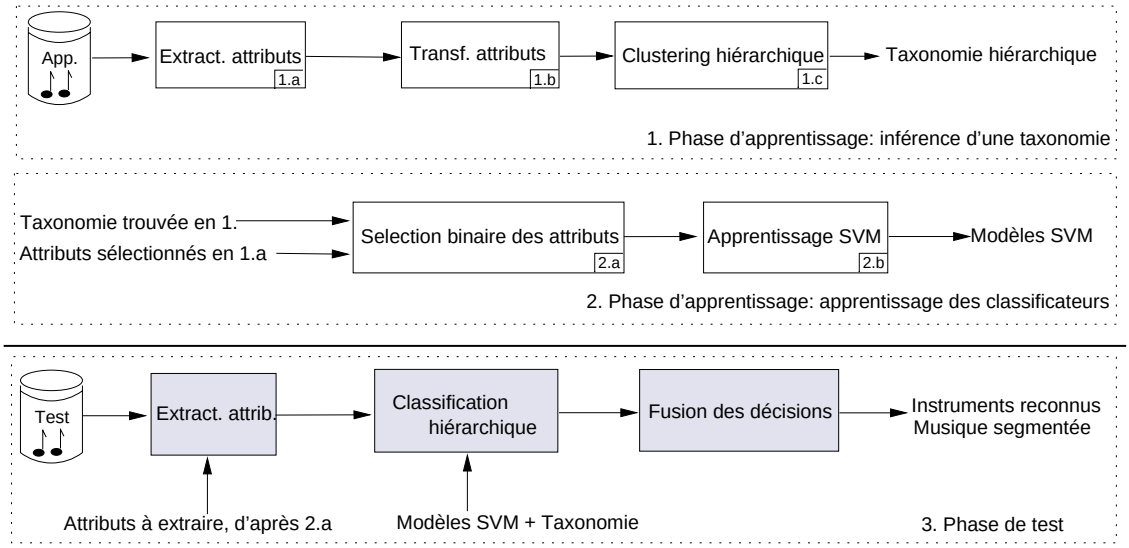

Fig. X.1 Schéma de principe du système de reconnaissance. Les blocs de test sont grisés.

A l'étape d'apprentissage, le système effectue les tâches suivantes :

1) *Construction de la taxonomie* :
   a) les descripteurs (donnés dans le tableau X.1) sont extraits du signal ;
   b) la dimension de l'espace des attributs est réduite par une PCA donnant lieu à un ensemble (plus petit) d'attributs transformés (*cf.* section VI-3) ;
   c) un algorithme de clustering hiérarchique (exploitant des distances probabilistes robustes) est utilisé pour inférer une taxonomie hiérarchique (tel que décrit dans la section IX-3-B) ;
2) *Apprentissage de classificateurs* :
   a) l'ensemble d'attributs original (obtenu à l'étape 1.a) est traité par l'algorithme de sélection IRMFSP utilisé dans une configuration binaire (*cf.* chapitre VI) pour produire un sous-ensemble optimal d'attributs pour chaque paire de classes possible,

à chaque nœud de la taxonomie (obtenue à l'étape 1.) ;

b) des classificateurs SVM (*cf.* section V-2) sont appris à chaque nœud de la taxonomie se basant sur les attributs sélectionnés à l'étape 2.a.

Pour le test (blocs grisés), seuls les attributs sélectionnés sont extraits du signal audio pour être utilisés dans la classification exploitant la taxonomie et les SVM apprises en 2.a.

X-2. Performances du système proposé

Nous testons notre système sur le corpus `MINS` (*cf.* chapitre II), composé d'extraits de jazz. Nous pensons que l'approche proposée peut être facilement suivie pour d'autres genres de musique (en supposant que le timbre des instruments n'ait pas été fortement modifié par des effets d'équalisation ou d'autres retouches par des ingénieurs du son).

A. La taxonomie automatique

Au moment où le présent système a été développé nous avions choisi d'utiliser les attributs transformés par PCA pour la construction de la taxonomie, en réduisant la dimension de l'espace transformé à 30[1]. La motivation en était que, ne connaissant pas à priori le résultat du clustering (les super-classes obtenues aux différents niveaux de la taxonomie), il était préférable d'utiliser l'information de tous les attributs en utilisant la PCA comme un moyen de réduire la dimension du problème. Des expériences ultérieures ont montré que des taxonomies plus pertinentes étaient obtenues en utilisant une sélection d'attributs (obtenue par un algorithme de sélection). C'est l'approche que nous avons suivi pour la construction de la taxonomie des instruments de musique au chapitre IX.

Pour le calcul des distances probabilistes nous avions utilisé un noyau RBF gaussien $k(\mathbf{x}, \mathbf{y}) = \exp\left(-\frac{||\mathbf{x}-\mathbf{y}||^2}{2\sigma^2}\right)$, avec $\sigma = 0.5$.

Comme indiqué dans la section V-3-A, la pertinence du résultat de clustering hiérarchique peut être évaluée à l'aide du coefficient de corrélation cophénétique qui doit être proche de 1. Nos expériences ont montré qu'un coefficient cophénétique plus grand était obtenu si les classes de solo (piano, batterie et contrebasse) n'étaient pas prises en compte dans le processus

[1] 94% de la variance totale est ainsi conservée.

de clustering hiérarchique des ensembles. Par conséquent, nous avons effectué le clustering de toutes les classes à l'exception du piano solo, la batterie solo et la contrebasse solo, en utilisant les deux distances de Bhattacharryya et divergence (calculées avec le noyau gaussien). La valeur du coefficient cophénétique obtenue avec la distance de Bhattacharryya est 0.85 contre 0.97 avec la divergence. Par suite, nous déduisons qu'un clustering efficace des ensemble a été effectué au moyen de la divergence.

Nous avons ensuite réalisé des coupes du dendrogramme pour obtenir différentes possibilités de clustering, avec un nombre de clusters qui a été fait varié de 4 à 16. Les niveaux de la taxonomie hiérarchique sont déduits à partir de ces différents clusterings, de telle sorte que les niveaux supérieurs sont déduits à partir de clusterings "grossiers" (faible nombre de clusters) et les niveaux inférieurs déduits de clusterings plus "fins" (nombre élevé de clusters). Le choix des niveaux à retenir est guidé par des considérations de "lisibilité" pour que les super-classes obtenues soient associées à des étiquettes qui puissent être formulées par l'Homme de façon intuitive. De plus, le nombre maximum de niveaux a été contraint à 4.

En prenant ces considérations en compte, les niveaux déduits des clusterings avec 6, 12 et 16 clusters ont été retenus, donnant lieu à la taxonomie représentée dans la figure X.2, où les solos ont été simplement insérés au sommet de la taxonomie (le plus haut). Des tests préliminaires ont montré que la classe BsDr[2] pouvait être mieux reconnue en l'associant au premier cluster (BsDrPn-BsDrPnM-BsDrW). Cela a été jugé acceptable puisque l'étiquette du nouveau cluster (BsDr-BsDrPn-BsDrPnM-BsDrW) est devenue plus "convenable" car elle peut être facilement décrite par "musique faisant intervenir contrebasse, batterie et autres instruments". Notons que tous les clusters obtenus portent des étiquettes faciles à formuler.

B. Attributs sélectionnés

L'algorithme IRMFSP dans une configuration binaire est utilisé à chaque nœud de la taxonomie, produisant des sous-ensembles d'attributs spécifiquement adaptés au contexte. Notons qu'à chaque nœud, un sous-ensemble différent d'attributs est recherché pour chaque paire de classes. Par exemple, au nœud (BsPn-BsPnM), trois sous-ensembles optimaux sont recherchés pour les trois problèmes bi-classes (BsPn vs BsEgPn), (BsPn vs BsPnVm) et (BsEgPn vs

[2]Bs : contrebasse *pizzicato*, Dr : batterie, etc. *cf.* chapitre II pour les codes des instruments.

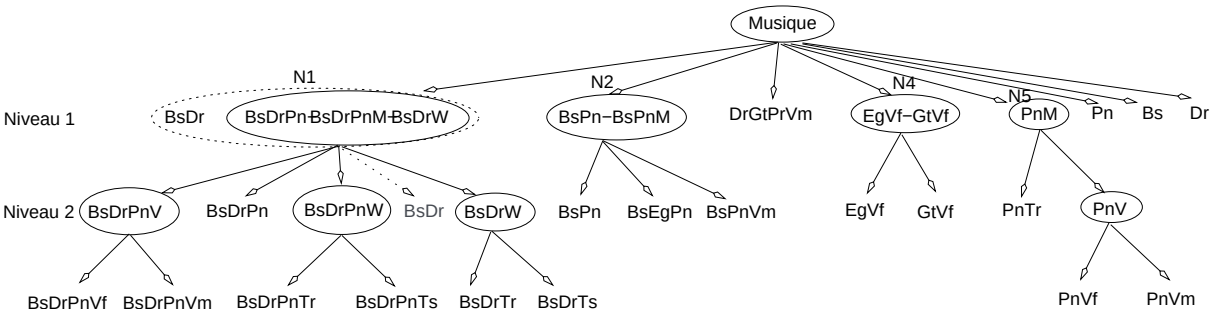

Fig. X.2 Taxonomie obtenue.

BsPnVm). De même, 10 sous-ensembles optimaux sont sélectionnés au nœud (BsDr-BsDrPnV-BsDrPn-BsDrPnW-BsDrW) et 28 sous-ensembles au niveau le plus haut. Le nombre total de sous-ensembles ainsi optimisés vaut 47 pour tous les nœuds de la taxonomie.

Le tableau X.1 liste les attributs examinés. A chaque nœud, $d = 50$ attributs sont sélectionnés par IRMFSP pour chaque paire de classes. La troisième colonne du tableau indique les attributs les plus fréquemment, sélectionnés à partir de chaque paquet, sur les 47 sous-ensembles obtenus.

Les attributs qui sont les plus fréquemment choisis sont les SMR (24 d'entre eux ont été sélectionnés sur les 47 sous-ensembles). Ce descripteur qui n'a pas été retenu pour la classification des instruments dans le contexte mono-instrumental, s'avère particulièrement utile dans le cas multi-instrumental. Même s'il est difficile d'interpréter intuitivement ce résultat, nous pouvons déduire que les effets de masquage inhérents à des sources sonores différentes permettent leur discrimination. Les autres attributs perceptuels efficaces sont la loudness spécifique relative, particulièrement dans les bandes de fréquences Bark élevées, et la sharpness.

Pour ce qui est des descripteurs spectraux, ceux déduits à partir des moments spectraux ainsi que la décroissance spectrale et la platitude spectrale globale ont plus de succès que les autres descripteurs spectraux.

Les moments temporels à court terme et à long terme s'avèrent aussi efficaces. De plus, la variation du kurtosis dans le temps est fréquemment sélectionnée pour décrire la variation des transitoires de la forme d'onde du signal audio, ce qui n'est pas surprenant pour de la musique comprenant des percussions.

Enfin, un nombre réduit de coefficients cepstraux a été sélectionné (en présence des autres attributs), ce qui confirme que ce descripteur n'est pas incontournable pour des tâches de reconnaissance sonore. Les attributs restant sont sélectionnés de façon marginale pour des paires

Paquet d'attributs	Taille	Les plus fréquemment sélectionnés dans le paquet
$AC = [A1, ..., A49]$	49	$AC49$ (4/47)
$Z = [ZCR, lZCR]$	2	ZCR (9/47), $lZCR$ (7/47)
$Tx = [Tc, Tw, Ta, Tk] + \delta + \delta^2$	12	$Tw(24/47), Tk(22/47)$
$lTx = [lTc, lTw, lTa, lTk] + \delta + \delta^2$	12	$lTc(20/47), lTw(27/47), lTk(23/47), \delta lTk(17/47)$
$Ex = [eTc, eTw, eTa, eTk] + \delta + \delta^2$	12	$Tw(24/47), eTk(23/47), \delta^2 eTk(14/47)$
$AM = [AM1, ..., AM8]$	8	$ampl.AM$ 10-40 Hz (8/47)
$Cp = [Cp1, ..., Cp10] + \delta + \delta^2$	30	$Cp1(28/47), Cp3(23/47)$
$AR = [AR1, AR2]$	2	$AR1(15/47)$, $AR2(14/47)$
$Sx = [Sc, Sw, Sa, Sk] + \delta + \delta^2$	12	$Sc(29/47), Sw(24/47), Sa(28/47), Sk(34/47)$
$ASF = [ASF1, ..., ASF23]$	23	$ASF22(13/47)$
$SCF = [SCF1, ..., SCF23]$	23	$SCF22(7/47)$
$[Ss, Sd, Sv, So, Fc]$	5	$Sd(17/47), So(22/47), Fc(14/47)$
$Si = [Si1, ..., Si21]$	21	$Si1(13/47)$
$OBSI = [O1, ..., O8]$	8	$O3(8/47), O8(7/47), O7(6/47)$
$OBSIR = [OR1, ..., OR7]$	7	$OR3(9/47)$
$Ld = [L1, ..., L24] + \delta + \delta^2$	72	$Ld4(31/47)$
$[Sh, Sp] + \delta + \delta^2$	6	$Sh(30/47), Sp(9/47)$
$SMR = [S1, ..., S51]$	51	$S38, S51(31/47), S15, S21(29/47), S1(28/47)$ $S19, S29, S41, S43, S46, (27/47)$

Tab. X.1 Paquets d'attributs utilisés dans l'étude sur la reconnaissance multi-instrumentale et attributs les plus fréquemment sélectionnés dans chaque paquet. Les fractions entre parenthèses indiquent le nombre de paires de classes (parmi toutes les paires possibles) pour lesquelles les attributs donnés ont été sélectionnés.

de classes spécifiques.

Pour le détail des sous-ensembles d'attributs sélectionnés pour chaque paire de classes, le lecteur pourra consulter [Essid, b].

C. Classification

Nous examinons à présent les performances de classification du système proposé. Les fenêtres de décision utilisées sont de taille $N_t = 120$ (approximativement 2s). Cela permet à ce système d'être employé pour la segmentation de la musique d'ensemble. En combinant les décisions prises sur des fenêtres de 2s, il est aisé de définir les segments faisant intervenir chaque instrument ou mélange d'instruments.

	N1	N2	N3	N4	N5	N6	N7	N8
N1:BsDr-BsDrPn-BsDrPnM-BsDrW	**91**	1	0	0.	5	2	0.	0.
N2:BsPn-BsPnM	4	**83**	0	0	1	3	0	10
N3:DrGtPrVm	29	3	***63***	6	0	0	0	0
N4:EgVf-GtVf	19	2	0	**60**	18	1	0	0
N5:PnM	26	1	2	11	**55**	4	0	0
N6:Pn	0	2	0	0	15	**83**	0	0
N7:Dr	**61**	0	0	0	5	0	34	0
N8:Bs	0	44	0	0	0	2	0	***54***

Tab. X.2 Matrice de confusions au premier niveau.

Nous présentons les matrices de confusion obtenues avec notre système dans les tableaux X.2, X.3 et X.4, respectivement pour le premier niveau (au sommet), le deuxième niveau et le troisième niveau (feuilles de l'arbre) de la taxonomie utilisée. Les taux présentés entre parenthèses représentent une estimation des taux de reconnaissance absolus, *i.e.* obtenus en multipliant les taux de reconnaissance correspondant au nœud courant par les taux de reconnaissance des parents de ce nœud qui sont traversés en suivant le chemin de la racine (sommet) au nœud courant.

Certains résultats sont considérés comme préliminaires car nous manquons malheureusement de données pour certaines classes. En conséquence, les résultats correspondant aux classes pour

Noeud N1	N1.1		N1.2		N1.3		N1.4		N1.5	
	MAP	Heurist	MAP	Heurist	MAP	Heurist	MAP	Heurist	MAP	Heurist
N1.1:BsDrPnV	35 (32)	**46 (42)**	**50**	17	10	32	**5**	**5**	**0**	**0**
N1.2:BsDrPn	**0**	**1**	**100 (91)**	**72 (66)**	**0**	27	**0**	**0**	**0**	**0**
N1.3:BsDrPnW	**0**	**0**	**92**	50	8 (7)	**50 (46)**	**0**	**0**	**0**	**0**
N1.4:BsDrW	**0**	**0**	**0**	**0**	**0**	**0**	49 (45)	**79 (72)**	**51**	21
N1.5:BsDr	13	15	**8**	**5**	**0**	**1**	**0**	7	**79 (72)**	**72 (66)**

Noeud N2	N2.1		N2.2		N2.3	
	MAP	Heurist	MAP	Heurist	MAP	Heurist
N2.1:BsPn	**99 (82)**	**94 (78)**	**0**	**5**	**1**	**1**
N2.2:BsEgPn	***57***	43	33 (27)	***48 (40)***	10	10
N2.3:BsPnVm	**0**	**0**	**0**	**0**	***100 (83)***	***100 (83)***

Noeud N4	N4.1	N4.2
N4.1:EgVf	***100 (60)***	**0**
N4.2:GtVf	***100***	0 (0)

Noeud N5	N5.1	N5.2
N5.1:PnTr	**100 (55)**	**0**
N5.2:PnV	**0**	**100 (55)**

Tab. X.3 Matrice de confusions au deuxième niveau, en utilisant deux stratégies de décision alternatives aux nœuds N1 et N2. Taux de reconnaissance absolus entre parenthèses.

lesquelles la taille des données de test est inférieure à 200s sont données en lettres italiques pour prévenir de la limitation de leur validité statistique.

En commençant par le premier niveau, les résultats obtenus peuvent être considérés comme encourageants étant donnée la courte durée des fenêtres de décision et la grande variabilité qui caractérise les enregistrements utilisés. Le taux de reconnaissance moyen est de 65%. Pour la classe N1 (BsDr-BsDrPn-BsDrPnM-BsDrW)[3], 91% de taux de reconnaissance est atteint, alors que la classe N7 (batterie seule) n'est correctement identifiée que dans 34% des tests. La batterie est assignée à la classe N1 61% du temps. De nouveaux descripteurs sont nécessaires à une meilleure discrimination de ces deux classes. Par exemple, des attributs caractérisant l'absence d'harmonicité pourraient être efficaces dans ce cas, puisque les instruments percussifs tels que la batterie ne présentent pas une forte harmonicité. En général, la plupart des classes ont été majoritairement confondues avec N1 à l'exception de la classe N6 (piano). Ce résultat est intéressant : il est facile de discriminer le piano joué en solo et le piano joué avec un

[3]Bs : contrebasse *pizzicato*, Dr : batterie, Pn : Piano, W : trompette ou saxophone, M : trompette ou voix.

accompagnement (83% pour le piano contre 91% pour N1). Le piano a été plus fréquemment confondu avec la classe N5 (PnTr-PnV)- 15% du temps- qu'avec N1.

Au deuxième niveau, les résultats trouvés au nœud N1 en utilisant la règle de décision MAP ne sont pas acceptables. En effet, la classe BsDrPnW n'est correctement identifiée que dans 8% des tests et la classe BsDrPnV, 35% du temps, car ces deux classes sont fréquemment associées à l'étiquette BsDrPn, respectivement dans 92% et 50% des cas. De même, la classe BsDrW est confondue avec BsDr dans 51% des tests. Cela n'est pas surprenant étant données les contraintes d'annotation des signaux, mentionnées dans la section II-3. De fait, plusieurs exemples de la classe BsDrW se sont nécessairement immiscés dans les ensembles d'apprentissage et de test relatifs aux classes BsDrPnV et BsDrPnW. Il en est de même pour les données de la classe BsDrW qui contient sûrement des exemples de BsDr.

Nous adoptons donc une heuristique qui permet de pallier ce problème. Le fait est que pour les paires (BsDr vs BsDrW), (BsDrPn vs BsDrPnW) et (BsDrPn vs BsDrPnV), les surfaces de décision optimales sont biaisées à cause de la présence d'exemples aberrants à la fois dans les ensembles d'apprentissage et de test. Alternativement aux techniques de suppression des observations aberrantes [Dunagan et Vempala, 2001], qui peuvent être inefficaces dans notre cas, eu égard au nombre important d'*outliers*, nous utilisons un seuil de décision biaisé. Chaque fois qu'un segment de test est classé BsDr par le critère MAP, si la deuxième classe la plus probable est BsDrW nous révisons la décision en considérant uniquement la sortie du classificateur (BsDr vs BsDrW). Alors, deux actions sont entreprises :

- d'abord, nous classons les observations dans la catégorie BsDr seulement si $P(\text{BsDr} \mid \text{BsDr ou BsDrW}) > 0.8$, au lieu d'utiliser le seuil bayésien habituel de 0.5 ;
- ensuite, nous comptons le nombre d'observations classés BsDr au sein de la fenêtre d'observation (120 observations consécutives) et nous n'élisons cette classe que si les 2/3 des observations de la fenêtre de décision sont associés à cette étiquette, sinon le segment de 2s courant est classé BsDrW.

La même heuristique est suivie pour toutes les paires impliquant BsDrPn et la paire (BsPn vs BsEgPn) au nœud N2. Il en résulte qu'en moyenne, de meilleurs résultats de classification sont obtenus dans ces contextes, comme on peut le voir dans les colonnes du tableau X.3, étiquetés par "Heurist".

Enfin, de bonnes performances de classification de mélanges de quatre instruments peuvent être obtenues comme indiqué par le tableau X.4.

Noeud N1.1	N1.1.1	N.1.1.2
N1.1.1:BsDrPnVf	**87 (37)**	13
N1.1.1:BsDrPnVm	28	**72 (30)**

Noeud N1.3	N1.3.1	N1.3.2
N1.3.1:BsDrPnTr	**100 (46)**	0
N1.3.2:BsDrPnTs	29	**71 (33)**

Noeud N1.4	N1.4.1	N1.4.2
N1.4.1:BsDrTr	***100 (72)***	0
N1.4.2:BsDrTs	9	***91 (66)***

Noeud N5.2	N5.2.1	N5.2.2
N5.2.1:PnVf	***97 (53)***	3
N5.2.2:PnVm	28	**72 (40)**

Tab. X.4 Matrice de confusions au troisième niveau (feuilles de l'arbre).

Étant donné que les extraits utilisés dans nos expériences traduisent des conditions d'enregistrements variables (des enregistrements en studio et en *Live* ont été utilisés) et qu'une partie de ces extraits est au format mp3 (ce que l'on peut considérer comme des signaux bruités, corrompus par un bruit de quantification et avec une limitation de bande) nous sommes confiants quant à l'applicabilité de notre approche à d'autres genres musicaux. Le système n'est pas sensible à une balance variable dans le mixage des instruments puisqu'il est capable, par exemple, d'identifier correctement le mélange BsDrPn à la fois sur des passages de solo de piano (piano plus fort que la contrebasse et la batterie) et sur des passages de solo de contrebasse (contrebasse plus forte que le piano).

X-3. Conclusion

Nous avons présenté une nouvelle approche pour la reconnaissance des instruments de musique en contexte multi-instrumental. Nous avons montré que la stratégie qui consiste à reconnaître les mélanges d'instruments (joués simultanément) est réalisable, en utilisant un système de classification hiérarchique, et qu'elle donne lieu à de bonnes performances de classification.

La taxonomie hiérarchique utilisée peut être considérée comme efficace :

- elle a été générée automatiquement au moyen d'une approche de clustering exploitant des distances probabilistes robustes ;
- elle peut être facilement interprétée par l'Homme en ce sens que tous ses nœuds portent des étiquettes musicalement significatives, permettant des classifications intermédiaires utiles.

L'avantage majeur de l'approche choisie réside dans le fait qu'elle permet d'éviter les problèmes ardus de l'estimation de fréquences fondamentales multiples et de la séparation de sources musicales. Au contraire, notre système peut aider à la résolution de ces problèmes puisqu'il permet d'effectuer la segmentation de la musique par rapport aux instruments (ou plus simplement par rapport au nombre de sources musicales) en présence, ce qui permettrait aux systèmes de séparation de source de bénéficier à tout instant d'une information sur la structure harmonique des spectres des signaux.

Des taux de reconnaissance plus élevés pourraient être obtenus en utilisant des fenêtres de décision plus longues. Nous pensons que le système proposé peut donner lieu à différentes applications utiles acceptant des requêtes réalistes puisqu'il est potentiellement capable de prendre en charge un contenu musical quelconque, indépendamment de l'orchestration (impliquant éventuellement de la batterie ou une voix chantée). En particulier, notre approche peut être efficace pour l'identification de l'orchestration d'une pièce musicale (sans nécessairement se soucier des variations des mélanges d'instruments à différents instants de la pièce), en adoptant des stratégies de décision appropriées.

Conclusions et perspectives

Le travail mené au cours de cette thèse a permis d'obtenir un système de reconnaissance des instruments de musique performant, capable de prendre en charge des enregistrements sonores reflétant la diversité de la pratique musicale et des conditions d'enregistrement rencontrées dans le monde réel.

L'architecture de notre système final exploite un schéma de classification hiérarchique qui repose sur une taxonomie des instruments et des mélanges d'instruments. Cette taxonomie a été inférée automatiquement, au moyen d'un algorithme de clustering hiérarchique, en considérant de façon séparée, les données des extraits mono-instrumentaux et ceux des extraits multi-instrumentaux. Cela permet en effet d'obtenir des taxonomies plus efficaces et plus lisibles.

Nous avons atteint cette architecture en essayant de "systématiser" la façon d'atteindre des réalisations efficaces des deux "grands modules" de traitement : le module de description du signal et le module de classification proprement dite.

Afin de produire un ensemble d'attributs efficace, nous avons expérimenté un grand nombre de descripteurs de l'état-de-l'art pouvant être extraits de façon robuste à partir d'un contenu musical quelconque, et nous avons proposé de nouveaux descripteurs qui s'avèrent des plus utiles. Les plus efficaces de ces attributs ont été retenus au moyen d'un nouvel algorithme de sélection, baptisé FSFC[4], qui vient concurrencer des approches de sélection bien établies. FSFC nous a permis de regrouper les attributs présentant des distributions de valeurs similaires, et de les trier au sein de ces groupes par ordre d'efficacité pour la discrimination des instruments. Il a été obtenu au terme d'une étude que nous avons mené sur le comportement d'un certain nombre d'algorithmes de sélection des attributs dans le contexte des données audio. Nous avons, par

[4]Fisher-based Selection of Feature Clusters

ailleurs, mis en évidence qu'il est avantageux de réaliser cette opération de sélection de façon binaire en recherchant un sous-ensemble d'attributs optimal pour la discrimination de chaque paire de classes. En plus d'être performante, cette méthode offre la possibilité d'acquérir une meilleure compréhension du problème de classification et de suggérer des voies d'amélioration du système.

Ensuite, nous nous sommes penchés sur les fonctions de classification. Les machines à vecteurs support ont été élues. Elles sont utilisées dans une configuration "1 contre 1" et avec des sorties probabilistes. Une attention particulière a été portée au réglage des paramètres des SVM et nous avons réussi à cerner un ensemble de paramètres efficaces pour la tâche de la reconnaissance des instruments.

Nous avons comparé, dans le cas mono-intrumental, les performances réalisées par le système de classification basé sur la taxonomie hiérarchique à celles atteintes par une taxonomie naturelle des instruments. Nous avons montré que notre taxonomie donnaient des résultats légèrement supérieurs tout en indiquant que les deux taxonomies pouvaient être remises en cause. Les deux font l'hypothèse qu'il est utile de regrouper systématiquement les instruments ayant des propriétés acoustiques et perceptuelles similaires dans les mêmes nœuds, alors que nos résultats suggèrent qu'il serait plus intéressant de positionner ces instruments dans des nœuds distincts aux niveaux bas de la taxonomie.

Un effort important a été consacré à la création de bases de données sonores permettant une évaluation pertinente des performances de reconnaissance des systèmes proposés. Des extraits sonores de pièces musicales jouées en *solo* ont été collectionnés pour 19 instruments (représentant toutes les familles instrumentales) à partir d'albums différents, traduisant des styles de jeu et des conditions d'enregistrement variées. L'évaluation a été faite en assurant une séparation complète entre les sources (albums) fournissant les extraits utilisés dans la phase d'apprentissage et ceux dont sont tirés les extraits inclus dans l'ensemble de test. Nous avons ainsi mis en évidence le bon comportement en généralisation des schémas de classification que nous avons construit. Notons qu'aucune base d'extraits de musique instrumentale de cette taille et de cette diversité n'a pu être utilisée dans les travaux précédents, ce qui rend difficile la comparaison des performances de notre système avec d'autres propositions, mais donne du crédit à notre évaluation.

Une base d'extraits musicaux comprenant plusieurs instruments a également été constituée afin de tester les performances du système de reconnaissance multi-instrumental. Notre architecture parvient à identifier jusqu'à quatre instruments joués simultanément, à partir d'extraits de

musique *jazz* incluant des percussions, et ce avec des taux de reconnaissance pouvant dépasser les 80%, en exploitant des fenêtres de décision courtes (de deux secondes de longueur). Notre système est le premier à pouvoir reconnaître autant d'instruments joués simultanément dans des conditions réalistes. Il présente l'avantage de ne nécessiter aucune séparation préalable des sources musicales, et il ne repose sur aucune étape d'estimation de fréquences fondamentales.

Perspectives

Plusieurs améliorations peuvent être apportées aux différents blocs constituant notre schéma de classification. Nous revenons ici sur chacun de ces blocs.

Nous pouvons d'abord relever deux limitations majeures du système proposé, que l'on retrouve d'ailleurs dans la plupart des systèmes de classification audio. Elles concernent la façon dont le problème est envisagé, et elles ont des répercutions sur la manière de concevoir tous les blocs de traitement :

1) d'abord, la variation temporelle des signaux audio n'est pas efficacement prise en compte, car le système exploite des observations de paramètres sur des fenêtres temporelles de durée fixe (généralement de l'ordre de 30ms) en supposant l'indépendance de ces observations ;
2) le traitement d'un signal multi-canal (au minimum stéréophonique) est ramené à un seul canal, obtenu généralement en moyennant les différents canaux.

Nous pensons qu'un système de classification doit pouvoir tirer un meilleur partie de ces deux paramètres de *temps* et d'*espace*, et ce, aux différentes étapes du schéma de classification.

A l'étape d'extraction des descripteurs Des études ont montré l'intérêt d'extraire les descripteurs à des échelles temporelles différentes (fenêtres d'analyse de durées différentes) et nous avons mis en évidence le potentiel que représente la description spécifique des segments de signal de natures différentes (attaque, partie stable,...). Outre la nécessité de déterminer les échelles et les segments les plus appropriées aux différents descripteurs, se pose le problème de l'intégration des attributs multi-échelles dans une représentation qui puisse être efficacement exploitée par un classificateur. Des alternatives aux résumés de ces attributs par leurs moyennes et leurs variances peuvent être envisagées, par exemple en utilisant des modèles de leur évolution dans le temps.

La diversité spatiale peut être également mise à profit de différentes manières. Par exemple, des opérations sur les différents canaux (somme, différence) permettent d'accentuer ou de diminuer certaines caractéristiques des signaux dans le but de cibler un aspect particulier de leur description.

A l'étape de sélection des descripteurs La sélection d'attributs doit également profiter des informations temporelles et spatiales. En effet, il peut être considéré que les attributs se montrant peu stables dans le temps et sur les différents canaux sont moins efficaces. Il est également nécessaire d'effectuer la sélection d'attributs non plus à partir d'observations ponctuelles séparées mais en considérant la succession temporelle et les réalisations spatiales de sous-ensembles d'observations en utilisant des critères prenant en compte l'articulation de ces descripteurs.

En outre, il peut être avantageux de réaliser la sélection en utilisant des critères de séparabilité dans l'espace de dimension supérieure induit par le noyau utilisé par le classificateur, c'est en effet dans cet espace que ce classificateur agit.

Enfin, il est nécessaire de disposer d'un critère qui permette de déterminer automatiquement le nombre d'attributs à sélectionner d. Cela serait particulièrement profitable à l'approche de sélection binaire.

A l'étape de construction de la taxonomie Comme nous l'avons indiqué, il serait avantageux d'envisager l'inférence de taxonomies pour la classification sous un angle nouveau. L'objectif serait de regrouper aux premiers niveaux les classes "proches", tout en s'assurant qu'aux niveaux les plus bas, où les décisions finales sont prises, celles qui sont susceptibles d'être confondues se retrouvent dans des nœuds différents.

A l'étape de classification Plusieurs améliorations peuvent être envisagées à l'étape de conception des machines de classification, en particulier dans le cas de la classification par SVM qui a prouvé son efficacité pour notre tâche :

- en adoptant une stratégie de fusion : des classificateurs différents peuvent être utilisés sur les attributs associés à des échelles de temps différentes et à des canaux différents (originaux ou obtenus par transformation des canaux originaux), dans un système qui fusionne les décisions prises par tous les classificateurs ;
- en combinant des attributs issus de différentes descriptions (obtenues à des échelles temporelles différentes, à partir de canaux différents, etc.) dans une représentation vectorielle

unique de dimension élevée et en profitant de la capacité des SVM à défier le problème de la dimensionalité (*curse of dimensionality*), ce qui permet de modéliser implicitement les dépendances temporelles et spatiales entre les différentes représentations ;

- en modélisant les dépendances spatiales et temporelles de façon explicite au travers de chaînes de Markov Cachées ou de réseaux bayésiens exploitant des SVM probabilisés ;
- en dégageant les *invariances* du problème de classification à partir des différentes versions des descripteurs (issus de canaux ou de segments temporels différents, de versions des signaux retouchées par des effets d'ingénierie du son : réverbération, filtrage, etc...) pour un meilleur apprentissage des SVM, éventuellement en ayant recours aux SVM virtuels [Shölkopf et Smola, 2002].

Signalons enfin que le système de classification que nous proposons doit pouvoir être utilisé pour d'autres tâches que la reconnaissance des instruments de musique. Il serait intéressant de tester ses performances dans des contextes d'application différents, en particulier : la détermination de l'orchestration d'une pièce musicale, la discrimination de la parole et de la musique, l'identification de l'artiste, l'identification de fréquences fondamentales, etc.

ANNEXES

A. Calcul des distances probabilistes

Soit l_i le nombre d'observations de la classe Ω_i, soit $\Psi_i = [\phi_1, ..., \phi_{l_i}]$, avec $\phi_n = \Phi(\mathbf{x}_n)$, soit $\mathbf{s}_i$ un vecteur colonne de taille l_i tel que $\mathbf{s}_i = \frac{1}{l_i}\mathbf{1}$, avec $\mathbf{1}$ un vecteur de 1, soit $\mathbf{K}_i = \Psi_i^T \Psi_i$ ($\mathbf{K}_i$ est une matrice de Gram), soit $\mathbf{J}_i = \frac{1}{\sqrt{l_i}}(\mathbf{I}_{l_i} - \mathbf{s}_i \mathbf{1}^T)$ et $\bar{\mathbf{K}}_i = \mathbf{J}_i^T \mathbf{K}_i \mathbf{J}_i$. Les r_i valeurs propores et vecteurs propres de la matrice $\bar{\mathbf{K}}_i$ sont notés par $\{(\lambda_{n,i}, \mathbf{v}_{n,i})\}_{n=1}^{r_i}$, $\mathbf{V}_{r_i,i} = [\mathbf{v}_{1,i}, ..., \mathbf{v}_{r_i,i}]$ et $\Lambda_{r_i,i}$ est la matrice diagonale dont les éléments diagonaux sont $\{\lambda_{1,i}, ..., \lambda_{r_i,i}\}$ (r_1 et r_2 sont à choisir et sont tels que $r_i << l_i << F$). Soit $\mathbf{K}_{ij} = \Psi_i^T \Psi_j$ (peut être calculée en utilisant le noyau), $\mathbf{A}_i = \mathbf{J}_i \mathbf{J}_i^T$ et

$$\mathbf{B}_j = \mathbf{J}_j \mathbf{V}_{r_j,j} \Lambda_{r_j,j}^{-1} \mathbf{V}_{r_j,j}^T \mathbf{J}_j^T, \tag{A.1}$$

alors l'approximation de la divergence dans l'espcace de dimension supérieure s'exprime par

$$\hat{J}_D(p_1, p_2) = \hat{J}_R(p_1||p_2) + \hat{J}_R(p_2||p_1) \tag{A.2}$$

où

$$\hat{J}_R(p_1||p_2) = \frac{1}{2}\{\hat{\theta}_{121} + \hat{\theta}_{222} - \hat{\theta}_{122} - \hat{\theta}_{221} + tr[\Lambda_{r_1,1}] - \hat{\eta}_{12}\}, \tag{A.3}$$

$$\hat{\theta}_{ijk} = \mathbf{s}_i^T \mathbf{K}_{ik} \mathbf{s}_k - \mathbf{s}_i^T \mathbf{K}_{ij} \mathbf{B}_j \mathbf{K}_{jk} \mathbf{s}_k \tag{A.4}$$

et

$$\hat{\eta}_{ij} = tr[\mathbf{A}_i \mathbf{K}_{ij} \mathbf{B}_j \mathbf{K}_{ji}]. \tag{A.5}$$

Soit $\mathbf{L}_{12} = \mathbf{V}_{r_1}^T \mathbf{J}_1^T \mathbf{K}_{12} \mathbf{J}_2 \mathbf{V}_{r_2}$,

$$\mathbf{L} = \begin{bmatrix} 0.5\ \Lambda_{r_1,1} & 0.5\ \mathbf{L}_{12} \\ 0.5\ \mathbf{L}_{12}^T & 0.5\ \Lambda_{r_2,2} \end{bmatrix}, \tag{A.6}$$

$$\mathbf{P} = \begin{bmatrix} \sqrt{0.5}\,\mathbf{J}_1\mathbf{V}_{r_1,1} & 0 \\ 0 & \sqrt{0.5}\,\mathbf{J}_2\mathbf{V}_{r_2,2} \end{bmatrix} \tag{A.7}$$

et $\check{\mathbf{B}} = \mathbf{P}\mathbf{L}^{-1}\mathbf{P}^T$. L'approximation de la distance de Bhattacharryya dans l'espace de dimension supérieure est donnée par

$$\hat{J}_B(p_1, p_1) = \frac{1}{8}\{\hat{\xi}_{11} + \hat{\xi}_{22} - 2\hat{\xi}_{12}\}. \tag{A.8}$$

où

$$\hat{\xi}_{ij} = \mathbf{s}_i^T\mathbf{K}_{ij}\mathbf{s}_j - \mathbf{s}_i^T[\mathbf{K}_{i1}\mathbf{K}_{i2}]\,\check{\mathbf{B}}\begin{bmatrix} \mathbf{K}_{1j} \\ \mathbf{K}_{2j} \end{bmatrix}\mathbf{s}_j. \tag{A.9}$$

B. Analyse des confusions des systèmes hiérarchiques aux nœuds intermédiaires

B-1. Système basé sur la taxonomie naturelle

Les matrices de confusions correspondants aux résultats de classification aux nœuds intermédiaires sont données dans les tableaux B.1 et B.2, en prenant des décisions sur les fenêtres d'observation ($N_t = 1$).

Le taux de reconnaissance moyen au sommet de l'arbre est de 82.3%. Les instruments du nœud N0.2 sont correctement reconnus, en moyenne, dans 96.4% des tests. A l'exception du basson et du trombone, toutes les classes de N0.2 sont correctement reconnues dans plus de 95% des cas. Le résultat obtenu pour la contrebasse *con arco* (75%) ne peut pas être pris en compte à ce stade puisque si cette classe et associée au label Bs (contrebasse pizzicato) par la suite, cela n'est pas considéré comme une erreur de reconnaissance de l'instrument. La reconnaissance du piano ne pose pas de problème en comparaison avec celle de la guitare qui est confondue dans 30.1% des cas avec la super-classe N0.2. La batterie est également fréquemment confondue avec cette super-classe (dans 22.8% des tests).

La tâche se complique au deuxième niveau :

- le taux de reconnaissance moyen au nœud N0.1 est de 84.0% ; les principales confusions concernent les paires (guitare vs piano) et (contrebasse vs guitare) ;
- les performances sont plus dégradées au nœud N0.2, le taux de reconnaissance moyen n'est que de 64.3% ; les principales difficultés rencontrées concernent la classification de la flûte (confondue avec la super-classe N0.2.3 dans 43.4% des tests), du tuba (assigné à N0.2.3 dans 41.8% des cas), mais aussi du violon, du basson, du saxophone ténor, et du saxophone soprano dont les taux de reconnaissance sont inférieurs à 65%.

N0	Pn-Gt-Bs	Vl-Va-Co-Ba-Tr-Tb-Fh-Ta-Cl-Cb-Ts-As-Ss-Ob-Bo-Fl	Dr
Pn	92.3	7.7	0.0
Gt	67.4	30.1	2.4
Bo	7.1	92.9	0.0
Ob	0.0	100.0	0.0
Cl	0.9	99.1	0.0
Fh	1.9	98.1	0.0
Tr	0.3	99.7	0.0
Co	7.2	92.8	0.0
Vl	0.0	100.0	0.0
Ba	24.4	75.1	0.5
As	1.5	98.5	0.1
Ts	1.5	98.5	0.0
Ss	0.3	99.7	0.0
Fl	1.0	99.0	0.0
Tb	10.5	89.5	0.0
Ta	4.7	95.3	0.0
Va	0.2	99.8	0.0
Bs	57.5	42.2	0.3
Dr	1.8	22.8	75.4

Tab. B.1 Matrice de confusions au nœud N0 (premier niveau).

Au dernier niveau nous observons des cas de confusions importantes :

- pour les cordes frottées (N0.2.1), violoncelle et alto posent problème, le premier est classé alto 27.3% du temps, et le deuxième est largement confondu avec le violon (34.3%) ;
- pour les bois (N0.2.3), les résultats de reconnaissance de la clarinette (39.0%), du saxophone ténor (27.7%) et du saxophone soprano (12.7%) ne sont pas acceptables. La clarinette est largement confondue avec le saxophone alto (37.0%). La distinction entre les différents saxophones est une tâche connue pour être difficile, les confusions sont importantes dans ce cas. Nous remarquons aussi un cas de confusion inattendu : le saxophone soprano est confondu avec le basson dans 25.2% des cas. Cela peut être dû au fait qu'une part importante des extraits musicaux de ces deux instruments est issue d'œuvres contemporaines dans lesquelles des styles de jeu extrêmes sont employés qui modifient fortement le timbre habituel des instruments ;
- pour les cuivres (N0.2.2), les cas difficiles sont : (cor vs trombone) et (tuba vs trombone).

N0.2	Vl-Va-Co-Ba	Tr-Fh-Tb-Ta	Cb-Cl-Ts-As-Ss-Ob-Bo	Fl
Bo	2.4	31.1	63.1	3.4
Ob	1.2	11.6	85.5	1.7
Cl	9.1	10.5	78.2	2.3
Fh	2.7	70.6	24.4	2.3
Tr	8.7	67.6	23.2	0.5
Co	80.4	0.2	18.9	0.5
Vl	59.2	5.2	33.5	2.2
Ba	83.6	2.2	14.2	0.0
As	8.4	7.0	83.9	0.6
Ts	28.7	8.0	62.8	0.5
Ss	3.0	25.9	60.3	10.8
Fl	9.5	2.8	43.4	44.3
Tb	5.1	65.0	29.4	0.5
Ta	0.7	57.5	41.8	0.0
Va	74.1	1.0	24.7	0.2

N0.1	Pn	Gt	Bs
Pn	82.9	14.9	2.2
Gt	17.3	81.0	1.7
Bs	1.7	10.0	88.2

N0.2.2	Fh	Tr	Tb	Ta
Fh	51.5	8.0	34.5	6.0
Tr	1.2	90.3	8.5	0.0
Tb	5.8	9.3	78.5	6.4
Ta	15.1	0.3	27.4	57.1

N0.2.3	Bo	Ob	Cl	As	Ts	Ss	Cb
Bo	63.8	1.8	13.8	15.8	1.5	3.2	0.0
Ob	0.0	85.8	9.6	2.9	0.1	1.7	0.0
Cl	1.4	6.2	39.0	37.0	4.6	6.5	5.4
As	1.7	0.7	3.9	81.8	8.0	3.3	0.5
Ts	2.3	2.4	5.4	55.5	27.7	4.8	1.8
Ss	25.2	4.8	15.6	33.9	7.7	12.7	0.1

N0.2.1	Co	Vl	Ba	Va
Co	55.7	11.4	5.6	27.3
Vl	2.7	77.4	0.3	19.5
Ba	15.1	0.2	84.0	0.6
Va	3.1	34.4	0.8	61.6

Tab. B.2 Matrices de confusions aux deuxième et troisième niveaux.

B-2. Système basé sur la taxonomie automatique

Nous étudions les résultats de classification aux nœuds intermédiaires indépendamment de leurs prédécesseurs. Les matrices de confusions correspondantes sont données dans les tableaux B.3, B.4, B.5 et B.6, correspondant à des décisions prises sur les fenêtres d'observation (N_t=1).

N0	Pn-Gt-Co-Cb-Ts-Cl-Vl-As-Ss-Va-Fl	Bo-Fh-Tb	Ob-Tr	Ba-Bs-Ta	Dr
Pn	94.5	0.7	0.7	4.1	0.0
Gt	94.6	0.3	0.0	4.1	1.0
Bo	38.8	59.3	0.7	1.3	0.0
Ob	19.1	1.1	79.8	0.0	0.0
Cl	91.0	4.7	3.3	1.0	0.0
Fh	39.8	55.1	1.6	3.5	0.0
Tr	25.6	4.6	69.7	0.0	0.1
Co	96.8	0.1	0.1	3.0	0.0
Vl	95.7	0.3	4.0	0.0	0.0
Ba	27.8	0.2	0.0	71.8	0.2
As	93.6	4.7	1.2	0.3	0.1
Ts	93.2	3.1	3.1	0.6	0.0
Ss	66.3	29.0	4.6	0.1	0.0
Fl	93.1	1.0	5.9	0.1	0.0
Tb	43.0	53.3	3.6	0.1	0.0
Ta	41.0	33.0	0.0	26.0	0.0
Va	98.7	0.3	0.7	0.3	0.0
Bs	16.4	0.3	0.0	83.0	0.2
Dr	23.0	0.2	0.0	0.8	76.0

Tab. B.3 Matrice de confusion au nœud N0 (pemier niveau).

Le taux de reconnaissance moyen au sommet de l'arbre est de 74.4%. Les instruments du nœud N0.1 sont reconnus avec succès dans 92.8% des tests. La super-classe qui pose problème est N0.2 qui n'est correctement reconnue que 56.2% du temps : les trois instruments qui la composent (basson, cor et trombone) sont largement confondus avec la super-classe N0.1. La reconnaissance des autres super-classes du premier niveau reste acceptable, la moyenne dépasse les 70.0% pour N0.3, N0.4 et Dr.

Les taux de reconnaissance au deuxième niveau sont majoritairement supérieurs à 80%. Les cas critiques concernent la clarinette (assignée à la super-classe N0.1.2 48.2% du temps), le violoncelle, le saxophone ténor et le cor (autour de 65% de taux de reconnaissance). Nous remarquons que ces confusions ont lieu entre groupes distincts contenant des instruments appartenant aux

N0.1	Pn-Gt-Co-Cb-Ts-Cl	Vl-As-Ss-Va-Fl
Pn	98.5	1.5
Gt	96.1	3.9
Cl	51.8	48.2
Co	62.9	37.1
Vl	3.4	96.6
As	18.3	81.7
Ts	33.9	66.1
Ss	30.2	69.8
Fl	18.0	82.0
Va	8.9	91.1

N0.2	Bo-Fh	Tb
Bo	84.1	15.9
Fh	65.9	34.1
Tb	25.5	74.5

N0.3	Ob	Tr
Ob	81.4	18.6
Tr	14.8	85.2

N0.4	Ba-Bs	Ta
Ba	98.7	1.3
Ta	31.1	68.9
Bs	93.0	7.0

Tab. B.4 Matrices de confusions au deuxième niveau, nœuds N0.1, N0.2, N0.3 et N0.4.

N0.1.1	Pn-Gt	Co-Cb-Ts	Cl
Pn	97.4	2.1	0.4
Gt	78.3	18.7	3.0
Cl	5.9	33.8	60.3
Co	8.4	80.6	11.0
Ts	4.0	84.5	11.5

N0.1.2	Vl-As-Ss-Va	Fl
Vl	96.6	3.4
As	98.4	1.6
Ss	73.5	26.5
Fl	35.0	65.0
Va	99.5	0.5

N0.2.1	Bo	Fh
Bo	44.0	56.0
Fh	4.4	95.6

Tab. B.5 Matrices de confusion au troisième niveau, nœuds N0.1.1, N0.1.2 et N0.2.1.

N0.1.1.1	Pn	Gt
Pn	83.4	16.6
Gt	17.9	82.1

N0.1.1.2	Co	Ts	Cb
Co	70.9	23.3	5.9
Ts	27.4	68.5	4.0

N0.1.2.1	Vl	As	Ss	Va
Vl	58.3	17.6	11.4	12.6
As	4.9	87.9	3.7	3.5
Ss	7.1	66.1	22.4	4.4
Va	28.6	9.6	8.6	53.3

Tab. B.6 Matrices de confusions aux extrémités de l'arbre, nœuds N0.1.1.1, N0.1.1.2 et N0.1.2.1.

mêmes familles.

Les instruments qui posent problème au plus bas de l'arbre sont principalement le saxophone soprano, largement confondu avec le saxophone alto au nœud N0.1.2.1 (dans 66.1% des cas), le basson (assigné à la classe cor dans 56.0% des tests), le violon et l'alto, correctement reconnus dans moins de 60% des cas.

B-3. Système basé sur la taxonomie automatique et la sélection binaire

Les matrices de confusions correpondant aux différents nœuds de la taxonomie automatique sont représentés dans les tableaux B.7, B.8, B.9 et B.10 (avec N_t=1). Nous remarquons une amélioration des performances moyennes aux différents niveaux de la hiérarchie. Nous n'obtenons pas d'amélioration systématique des taux de reconnaissance pour tous les instruments mais une redistribution des confusions plus avantageuse et cela se fait de deux manières :

- par une résolution "bilatérale" des confusions, c'est le cas par exemple de la paire (piano vs guitare) au nœud N0.1.1.1 : dans le cas non binaire, le piano est confondu avec la guitare 16.6% du temps et la guitare confondue avec le piano dans 17.9% des tests, alors que dans le cas binaire le piano est identifié à la guitare dans 12.7% des cas et la guitare assignée à la classe piano dans 14.1% des cas ; les deux instruments sont mieux reconnus ;
- une résolution "partielle" des confusions, c'est le cas par exemple au nœud N0.3 où le hautbois est confondu avec la trompette dans 18.6% des tests et la trompette avec le hautbois 14.8% du temps, avec une sélection non binaire des attributs, alors qu'avec la sélection binaire, la trompette n'est confondue avec le hautbois que dans 10.2% des cas mais le hautbois est plus fréquemment confondu avec la trompette (18.9% du temps) ; en moyenne les confusions sont donc moins importantes.

N0	Pn-Gt-Co-Cb-Ts-Cl-Vl-As-Ss-Va-Fl	Bo-Fh-Tb	Ob-Tr	Ba-Bs-Ta	Dr
Pn	94.4	0.7	0.8	4.1	0.0
Gt	93.1	0.2	0.0	4.1	2.6
Bo	36.9	59.4	2.7	1.0	0.0
Ob	18.5	1.3	80.2	0.0	0.0
Cl	92.2	4.6	2.1	1.0	0.0
Fh	40.8	56.4	0.6	2.2	0.0
Tr	23.2	4.9	71.9	0.0	0.1
Co	96.7	0.1	0.2	3.0	0.0
Vl	96.5	0.3	3.2	0.0	0.0
Ba	27.8	0.1	0.0	71.5	0.6
As	93.0	4.5	0.7	0.4	1.4
Ts	88.5	3.2	6.0	0.5	1.8
Ss	67.3	29.3	3.4	0.1	0.0
Fl	93.8	1.2	5.0	0.0	0.0
Tb	42.2	54.6	3.2	0.0	0.0
Ta	42.3	27.8	0.0	29.3	0.6
Va	98.8	0.3	0.6	0.3	0.0
Bs	16.4	0.2	0.0	83.1	0.3
Dr	21.5	0.1	0.0	1.4	76.9

Tab. B.7 Matrice de confusions au premier niveau (nœud N0) avec une sélection binaire des attributs.

N0.2	Bo-Fh	Tb
Bo	84.1	15.9
Fh	62.7	37.3
Tb	23.6	76.4

N0.3	Ob	Tr
Ob	81.1	18.9
Tr	10.2	89.8

N0.4	Ba-Bs	Ta
Ba	98.3	1.7
Ta	30.1	69.9
Bs	91.9	8.1

Tab. B.8 Matrices de confusions au premier niveau, nœuds N0.2, N0.3 et N0.4. Pas de modifications au nœud N0.1.

N0.1.1	Pn-Gt	Co-Cb-Ts	Cl
Pn	97.5	2.2	0.3
Gt	78.7	18.9	2.4
Cl	6.2	33.7	60.1
Co	8.6	79.2	12.3
Ts	3.8	82.6	13.6

N0.1.2	Vl-As-Ss-Va	Fl
Vl	97.6	2.4
As	97.7	2.3
Ss	75.2	24.8
Fl	24.0	76.0
Va	99.8	0.2

N0.2.1	Bo	Fh
Bo	57.5	42.5
Fh	11.6	88.4

Tab. B.9 Matrices de confusion au troisième niveau, nœuds N0.1.1, N0.1.2 et N0.2.1.

N0.1.1.1	Pn	Gt
Pn	87.3	12.7
Gt	14.1	85.9

N0.1.1.2	Co	Ts	Cb
Co	67.4	19.0	13.6
Ts	14.3	79.1	6.6

N0.1.2.1	Vl	As	Ss	Va
Vl	66.4	14.0	13.1	6.6
As	4.4	89.6	2.2	3.8
Ss	7.5	69.4	20.2	2.8
Va	33.0	8.9	5.6	52.5

Tab. B.10 Matrices de confusions aux extrémités de l'arbre, nœuds N0.1.1.1, N0.1.1.2 et N0.1.2.1.

C. Sélection de publications

Bibliographie

Bibliographie de l'auteur

— Articles de revues —

[Essid *et al.*, 2006a] Slim Essid, Gaël Richard, et Bertrand David. Instrument recognition in polyphonic music based on automatic taxonomies. *IEEE Transactions on Speech and Audio Processing*, janvier 2006. à paraitre.

[Essid *et al.*, 2006b] Slim Essid, Gaël Richard, et Bertrand David. Musical instrument recognition by pairwise classification strategies. *IEEE Transactions on Speech and Audio Processing*, juin 2006. à paraitre.

— Articles de conférences —

[Essid *et al.*, 2005a] S. Essid, P. Leveau, G. Richard, L. Daudet, et B. David. On the usefulness of differentiated transient/steady-state processing in machine recognition of musical instruments. Dans *AES 118th Convention*, Barcelona, mai 2005.

[Essid *et al.*, 2004a] Slim Essid, Gaël Richard, et Bertrand David. Efficient musical instrument recognition on solo performance music using basic features. Dans *AES 25th International Conference*, London, UK, juin 2004.

[Essid *et al.*, 2004b] Slim Essid, Gaël Richard, et Bertrand David. Musical instrument recognition based on class pairwise feature selection. Dans *5th International Conference on Music Information Retrieval (ISMIR)*, Barcelona, Spain, octobre 2004.

[Essid *et al.*, 2004c] Slim Essid, Gaël Richard, et Bertrand David. Musical instrument recognition on solo performance. Dans *European Signal Processing Conference (EUSIPCO)*, Vienna, Austria, septembre 2004.

[Essid *et al.*, 2005b] Slim Essid, Gaël Richard, et Bertrand David. Inferring efficient hierarchical taxonomies for MIR tasks : Application to musical instruments. Dans *6th International Conference on Music Information Retrieval (ISMIR)*, London, UK, septembre 2005.

[Essid *et al.*, 2005c] Slim Essid, Gaël Richard, et Bertrand David. Instrument recognition in polyphonic music. Dans *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Philadelphia, USA, mars 2005.

Bibliographie du document

[Agostini *et al.*, 2001] G. Agostini, M. Longari, et E. Pollastri. Musical instrument timbres classification with spectral features. Dans *International Workshop on Multimedia Signal Processing*, pages 97–102, Cannes, France, octobre 2001.

[Agostini *et al.*, 2003] G. Agostini, M. Longari, et E. Pollastri. Musical instrument timbres classification with spectral features. *EURASIP Journal on Applied Signal Processing*, 1(11), 2003.

[Atal et Rabiner, 1976] B. Atal et L. Rabiner. A pattern recognition approach to voiced-unvoiced-silence classification with applications to speech recognition. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 24 :201–212, 1976.

[Bello *et al.*, 2004] J.P. Bello, C. Duxbury, M. Davies, et M.B. Sandler. On the use of phase and energy for musical onset detection in the complex domain. *IEEE Signal Processing Letters*, 11(6) :553–556, juin 2004.

[Bello *et al.*, 2005] Juan P. Bello, L. Daudet, S. Abdallah, C. Duxbury, M. Davies, et M.B. Sandler. A tutorial on onset detection in music signals. *IEEE transactions on speech and audio processing*, septembre 2005.

[Berthomier, 1983] C. Berthomier. Instantaneous frequency and energy distribution of a signal. *Signal processing*, 1983.

[Blum et Langley, 1997] A. L. Blum et P Langley. Selection of relevant features and examples in machine learning. *Artificial Intelligence Journal*, 97(1-2) :245–271, décembre 1997.

[Brooks,] Mike Brooks. http ://www.ee.ic.ac.uk/hp/staff/dmb/voicebox/voicebox.html, -.

[Brown,] J. Brown. http ://web.media.mit.edu/%7Ebrown/cqtrans.htm, -.

[Brown, 1991] Judith C. Brown. Calculation of a constant q spectral transform. *Journal of the Acoustical Society of America*, 89 :425–434, janvier 1991.

[Brown, 1998] Judith C. Brown. Musical instrument identification using autocorrelation coefficients. Dans *International Symposium on Musical Acoustics*, pages 291–295, 1998.

[Brown, 1999] Judith C. Brown. Computer identification of musical instruments using pattern recognition with cepstral coefficients as features. *Journal of the Acoustical Society of America*, 105 :1933–1941, mars 1999.

[Brown *et al.*, 2000] Judith C. Brown, Olivier Houix, et Stephen McAdams. Feature dependence in the automatic identification of musical woodwind instruments. *Journal of the Acoustical Society of America*, 109 :1064–1072, mars 2000.

[Burges, 1998] Christopher J.C. Burges. A tutorial on support vector machines for pattern recognition. *Journal of Data Mining and knowledge Discovery*, 2(2) :1–43, 1998.

[Campedel et Moulines, 2005] M. Campedel et E. Moulines. Unsupervised feature selection using support vector clustering. *à paraitre*, 2005.

[Chang *et al.*, 2001] S. F. Chang, T. Sikora, et Atul Puri. Overview of the mpeg-7 standard. *IEEE Transactions on Circuits and Systems*, 11(6) :688–695, juin 2001.

[Chetry *et al.*, 2005] N. Chetry, M. Davies, et M. Sandler. Musical instrument identification using lsf and k-means. Dans *AES 118*, Barcelona, 2005.

[Clark *et al.*, 1964] M. Clark, P. Robertson, et D. A. Luce. A preliminary experiment on the perceptual basis for musical instrument families. *Journal of the Audio Engenieering Society*, 12 :199–203, 1964.

[Cohen *et al.*, 2002] I. Cohen, Q. Tian, X. Sean, et T. Huang. Feature selection using principal feature analysis. Dans *IEEE Int.Conf. on Image Processing ICIP*, septembre 2002.

[d'Alessandro, 2002] C. d'Alessandro. *Analyse, synthèse et codage de la parole*. Hermes, Lavoisier, 2002.

[Davis et Mermelstein, 1980] Steven B. Davis et Paul Mermelstein. Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 28 :357–366, août 1980.

[Dempster *et al.*, 1977] A. Dempster, N. Laird, et D. Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society*, 39 :1–38, 1977.

[DePoli *et al.*, 1993] G. DePoli, P. Prandoni, et P. Tonella. Timbre clustering by self-organizing neural networks. Dans *Colloquium on Musical Informatics*. University of Milan, 1993.

[Dubnov, 1996] S. Dubnov. *Polyspectral Analysis of Musical Timbre*. PhD thesis, Hebrew University, 1996.

[Dubnov et Rodet, 1998] Shlomo Dubnov et Xavier Rodet. Timbre recognition with combined stationary and temporal features. Dans *International Computer Music Conference*, 1998.

[Duda *et al.*, 2001] Richard Duda, P. E. Hart, et David G. Stork. *Pattern Classification*. Wiley-Interscience, 2001.

[Dunagan et Vempala, 2001] J. Dunagan et S. Vempala. Optimal outlier removal in high-dimensional. Dans *33-rd Annual ACM symposium on thoery of Comupting*, pages 627–636, Hersonissos, Greece, juillet 2001.

[Eggink et Brown, 2003] Jana Eggink et Guy J. Brown. A missing feature approach to instrument identification in polyphonic music. Dans *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 553–556, Hong Kong, avril 2003.

[Eggink et Brown, 2004] Jana Eggink et Guy J. Brown. Instrument recognition in accompanied sonatas and concertos. Dans *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 217–220, Montreal, Canada, mai 2004.

[Ellis, 1996] D.P.W. Ellis. *Prediction-driven computational auditory scene analysis*. PhD thesis, Dept. of Elec. Eng & Comp. Sci., M.I.T., 1996.

[Eronen, 2001a] Antti Eronen. Automatic musical instrument recognition. Master's thesis, Tampere University of Technology, avril 2001.

[Eronen, 2001b] Antti Eronen. Comparison of features for musical instrument recognition. Dans *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, octobre 2001. New Paltz, New York.

[Eronen, 2003] Antti ERONEN. Musical instrument recognition using ICA-based transform of features and discriminatively trained HMMs. Dans *7th International Symposium on Signal Processing and Its Applications*, Paris, France, juillet 2003.

[Essid, a] Slim ESSID.
http ://www.tsi.enst.fr/%7Eessid/pub/thesis/hrclassif-pairwise-fsa.html, -.

[Essid, b] Slim ESSID.
http ://www.tsi.enst.fr/%7Eessid/pub/ieee-sa-fsa.html, -.

[Feiten et Ungvary, 1991] B. F. FEITEN et T. UNGVARY. Organization of sounds with neural nets. Dans *International Computer Music Conference*, octobre 1991.

[Fletcher et Rossing, 1991] N. FLETCHER et T. ROSSING. *The Physics of Musical Instruments*. Springer Verlag, 1991.

[Fraser et Fujinaga, 1999] Anglea FRASER et Ichiro FUJINAGA. Toward real-time recognition of acoustic musical instruments. Dans *International Computer Music Conference*, octobre 1999.

[Fujinaga, 1998] Ichiro FUJINAGA. Machine recognition of timbre using steady-state tone of acoustic musical instruments. Dans *International Computer Music Conference*, 1998.

[Fujinaga et MacMillan, 2000] Ichiro FUJINAGA et Karl MACMILLAN. Realtime recognition of orchestral instruments. Dans *International Computer Music Conference*, 2000.

[Gillet et Richard, 2004] Olivier GILLET et Gaël RICHARD. Automatic transcription of drum loops. Dans *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Montreal, Canada, mai 2004.

[Goto *et al.*, 2003] Masataka GOTO, Hiroki HASHIGUCHI, Takuichi NISHIMURA, et Ryuichi OKA. Rwc music database : Music genre database and musical instrument sound database. Dans *4th International Conference on Music Information Retrieval*, pages 229–230, 2003.

[Goto *et al.*, 2002] Masataka GOTO, Hiroki HASHIGUSHI, Takuishi NISHIMURA, et Ryuichi OKA. RWC music database : Popular, classical, and jazz music databases. Dans *International Conference on Music Information Retrieval (ISMIR)*, Paris, France, octobre 2002.

[Grey, 1977] K. M. GREY. Multidimensional perceptual scaling of musical timbres. *Journal of the Acoustical Society of America*, 61 :1270–1277, 1977.

[Guyon et Elisseeff, 2003] I. Guyon et A Elisseeff. An introduction to feature and variable selection. *Journal of Machine Learning Research*, 3 :1157–1182, 2003.

[Guyon *et al.*, 2002] I. Guyon, J. Weston, S. Barnhill, et Vapnik V.. Gene selection for cancer classification using support vector machines. *Journal of Machine Learning*, 46 :389–422, 2002.

[Hastie et Tibshirani, 1998] Trevor Hastie et Robert Tibshirani. Classification by pairwise coupling. Dans *Advances in Neural Information Processing Systems*, volume 10. The MIT Press, 1998.

[Herrera *et al.*, 2003] Perfecto Herrera, Geoffroy Peeters, et Shlomo Dubnov. Automatic classification of musical sounds. *Journal of New Music Research*, 32(1) :3–21, 2003.

[IOWA, 1997] IOWA. The university of iowa electronic music studios. http ://theremin.music.uiowa.edu, 1997.

[ISO/IEC, 1997] ISO/IEC. MPEG-2 Advanced Audio Coding, AAC. International Standard ISO/IEC 13818-7, ISO/IEC, avril 1997.

[ISO/IEC, 2001] ISO/IEC. Information technology - multimedia content description interface - part 4 : Audio. International Standard ISO/IEC FDIS 15938-4 :2001(E), ISO/IEC, juin 2001.

[Joachims,] Thorsten Joachims. Svm light support vector machine. http ://svmlight.joachims.org/, -.

[Joachims, 1999] Thorsten Joachims. *Making large-Scale SVM Learning Practical*. MIT Press, Cambridge, USA, 1999.

[Joachims, 2000] Thorsten Joachims. Estimating the generalization performance of a svm efficiently. Dans *International Conference on Machine Learning*, 2000.

[Kaminskyj, 2000] Ian Kaminskyj. Multi-feature musical instrument sound classifier. Dans *Australasian Computer Music Conference*, Queesland University of Technology, juillet 2000.

[Kaminskyj et Materka, 1995] Ian Kaminskyj et A. Materka. Automatic source identification of monophonic musical instrument sounds. Dans *IEEE International Conference on Neural Networks*, pages 189– 194, 1995.

[Kashino et Mursae, 1998] Kunio KASHINO et Hiroshi MURSAE. A sound source identification system for ensemble music based on template adaptation and music stream exrtaction. *Speech Communication*, 27 :337–349, septembre 1998.

[Kedem, 1986] B. KEDEM. Spectral analysis and discrimination by zero-crossings. *Proceedings of the IEEE*, 74 :1477–1493, 1986.

[Kendall, 1986] R. A. KENDALL. The role of acoustic signal partitions in listener categorization of musical phrases. *Music perception*, 4 :185–214, 1986.

[Kinoshita *et al.*, 1999] Tomoyoshi KINOSHITA, S. SAKAI, et Hidehiko TANAKA. Musical sound source identification based on frequency component adaptation. Dans *IJCAI Workshop on Computational Auditory Scene Analysis (IJCAI-CASA)*, Stockholm, août 1999.

[Kitahara *et al.*, 2004] T. KITAHARA, M. GOTO, et H.G. OKUNO. Category-level identification of non-registered musical instrument sounds. Dans *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Montreal, Canada., mai 2004.

[Kitahara *et al.*, 2003] Testuro KITAHARA, Masataka GOTO, et Hiroshi G. OKUNO. Musical instrument identification based on f0-dependent multivariate normal distribution. Dans *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Hong Kong, avril 2003.

[Klapuri, 1999] A. KLAPURI. Sound onset detection by applying psychoacoustic knowledge. Dans *ICASSP*, 1999.

[Kohavi et John, 1997] Ron KOHAVI et G. JOHN. Wrappers for featue subset selection. *Artificial Intelligence Journal*, 97(1-2) :273–324, 1997.

[Kostek, 2004] Bozena KOSTEK. Musical instrument recognition and duet analysis employing music information retrieval techniques. *IEEE*, 92(4) :712–729, avril 2004.

[Kostek et Czyzewski, 2001a] Bozena KOSTEK et Andrzej CZYZEWSKI. Automatic recognition of musical instrument sounds - further developments. Dans *110th AES convention*, The Netherlands, mai 2001.

[Kostek et Czyzewski, 2001b] B. KOSTEK et A. CZYZEWSKI. Representing musical instrument sounds for their automatic classification. *J. Audio Eng. Soc.*, 9 :768–785, 2001.

[Krishna et Sreenivas, 2004] A.G KRISHNA et T.V. SREENIVAS. Music instrument recognition : from isolated notes to solo phrases. Dans *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 265–268, Montreal, Canada, mai 2004.

[Lee et Chun, 2002] Jonghyun Lee et Joohwan Chun. Musical instrument recognition using hidden markov model. Dans *Conference Record of the Thirty-Sixth Asilomar Conference on Signals, Systems and Computers*, pages 196–199, novembre 2002.

[Leveau, 2004] Pierre Leveau. Paramétrisation adaptée de transitoires pour la reconnaissance d'instruments de musique. Master's thesis, Laboratoire d'Acoustique Musicale, Université Pierre et Marie Curie, juillet 2004.

[Leveau *et al.*, 2004] P. Leveau, L. Daudet, et G. Richard. Methodology and tools for the evaluation of automatic onset detection algorithms in music, submitted. *Proceedings of ISMIR 2004*, octobre 2004.

[Li et Ogihara, 2005] Tao Li et Mitsunori Ogihara. Music genre classification with taxonomy. Dans *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Philadelphia, USA, mars 2005.

[LibSVM,] LibSVM.
http ://www.csie.ntu.edu.tw/%7Ecjlin/libsvm/, -.

[Linde *et al.*, 1980] Y. Linde, A. Buzo, et R.M. Gray. An algorithm for vector quantiacer design. *IEEE Transactions on Communication*, pages 84–95, 1980.

[Liu et Motoda, 2000] Huan Liu et Hiroshi Motoda. *Feature selection for knowledge discovery and data mining*. Kluwer academic publishers, 2nd édition, 2000.

[Livshin et Rodet, 2004a] Arie Livshin et Xavier Rodet. Instrument recognition beyond separate notes - indexing continuous recordings. Dans *International Computer Music Conference*, Miami, USA, novembre 2004.

[Livshin et Rodet, 2004b] Arie Livshin et Xavier Rodet. Musical instrument identification in continuous recordings. Dans *7th International Conference on Digital Audio Effects (DAFX-4)*, Naples, Italy, octobre 2004.

[Mallat, 2000] S. Mallat. *Une exploration des signaux en ondelettes*. Les Editions de l'Ecole Polytechnique, 2000.

[Marques et Moreno, 1999] Janet Marques et Pedro J. Moreno. A study of musical instrument classification using gaussian mixture models and support vector machines. Rapport Technique, Compaq Computer Corporation, 1999.

[Martin, 1999] Keith Dana Martin. *Sound-Source Recognition : A Theory and Computational Model*. PhD thesis, Massachusets Institue of Technology, juin 1999.

[McAdams *et al.*, 1995] Stephen McAdams, S. Winsberg, S. Donnadieu, G. De Soete, et J. Krimphoff. Perceptual scaling of synthesized musical timbres : common dimensions, specificities and latent subject classes. *Psychological reserach*, 58 :177–192, 1995.

[McKay et Fujinaga, 2004] C. McKay et I. Fujinaga. Automatic genre classification using large high-level musical feature sets. Dans *5th International Conference on Music Information Retrieval (ISMIR)*, Barcelona, Spain, octobre 2004.

[Mitra *et al.*, 2002] P. Mitra, C. Murthy, et S. Pal. Unsupervised feature selection using feature similarity. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2002.

[Moon, 1996] Todd K. Moon. The expectation-maximization algorithm. *IEEE Signal processing magazine*, pages 47– 60, novembre 1996.

[Moore et Glasberg, 1997] Moore et Glasberg. A model for the prediction of thresholds, loudness and partial loudness. *J.Audio.Eng.Soc.*, 45 :224–240, 1997.

[Moreau, 1995] Nicolas Moreau. *Techniques de Compression des signaux*. Masson, Collection technique et scientifique des télécommunications, 1995.

[Nawab et Quatieri, 1988] S. H. Nawab et Th. F. Quatieri. *Short-Time Fourier Transform*. Prentice-Hall, 1988.

[Opolko et Wapnick, 1987] F. Opolko et J. Wapnick. Mc Gill university master samples. McGill University, 1987.

[Pachet et Cazaly, 2000] F. Pachet et D. Cazaly. A taxonomy of musical genres. Dans *Content-Based Multimedia Information Access Conference (RIAO)*, Paris, France, avril 2000.

[Pachet et Zils, 2003] Francois Pachet et Aymeric Zils. Evolving automatically high- level music descriptors from acoustic signals. Dans *1st International Symposium on Computer Music Modeling and Retrieval (CMMR)*, Montpellier, France, mai 2003.

[Painter et Spanias, 2000] Ted Painter et Andreas Spanias. Perceptual coding of digital audio. *IEEE*, 88(4) :451–512, avril 2000.

[Peeters, 2003] Geoffroy Peeters. Automatic classification of large musical instrument databases using hierarchical classifiers with inertia ratio maximization. Dans *115th AES convention*, New York, USA, octobre 2003.

[Peeters, 2004] Geoffroy PEETERS. A large set of audio features for sound description (similarity and classification) in the cuidado project. Rapport Technique, IRCAM, 2004.

[Peeters et Rodet, 2002] Geoffroy PEETERS et Xavier RODET. Automatically selecting signal descriptors for sound classification. Dans *International Computer Music Conference*, Goteborg, septembre 2002.

[Platt, 1999] John C. PLATT. Probabilistic outputs for support vector machines and comparisions to regularized likelihood methods. *Advances in Large Margin Classifiers*, 1999.

[Plomp, 1970] R. PLOMP. Timbre as a multidimensional attribute of complex tones. Dans R. PLOMP et G.F. SMOORENBURG, éditeurs, *Frequency Analysis and Periodicity Detection in Hearing*, pages 197–414, 1970.

[R. Rabiner, 1993] Lawrence R. RABINER. *Fundamentals of Speech Processing*. Prentice Hall Signal Processing Series. PTR Prentice-Hall, Inc., 1993.

[Reynolds et Rose, 1995] Douglas A. REYNOLDS et Richard C. ROSE. Robust text- independent speaker identification using gaussian mixture speaker models. *IEEE Transactions on Speech and Audio Processing*, 3 :72–83, janvier 1995.

[Rodet et Jaillet, 2001] Xavier RODET et Florent JAILLET. Detection and modeling of fast attack transients. Dans *International Computer Music Conference*, septembre 2001.

[Scheirer et Slanely, 1997] E. SCHEIRER et Malcom SLANELY. Construction and evaluation of a robust multifeature speech/music discriminator. Dans *IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 1331–1334, avril 1997.

[Schölkopf *et al.*, 1995] B SCHÖLKOPF, C. BURGES, et V. VAPNIK. Extracting support data for a given task. Dans *International Conference on Knowledge Discovery and Data Mining*, 1995.

[Shölkopf et Smola, 2002] B. SHÖLKOPF et A. J. SMOLA. *Learning with kernels*. The MIT Press, Cambridge, MA, 2002.

[SOL,] SOL. Ircam studio online. http ://www.ircam.fr, -.

[Spider,] SPIDER.
http ://www.kyb.tuebingen.mpg.de/bs/people/spider/, -.

[Theodoridis et Koutroumbas, 1998] Sergios THEODORIDIS et Konstantinos KOUTROUMBAS. *Pattern recognition*. Academic Press, 1998.

[Vapnik, 1995] Vladimir VAPNIK. *The nature of statistical learning theory.* Springer-Verlag, 1995.

[Ventura-Miravet *et al.*, 2003] Raquel VENTURA-MIRAVET, Fionn MURTAGH, et Ji MING. Pattern recognition of musical instruments using hidden markov models. Dans *Stockholm Music Acoustics Conference*, pages 667–670, Stockholm, Sweeden, août 2003.

[Vincent et Rodet, 2004] E. VINCENT et Xavier RODET. Instrument identification in solo and ensemble music using independent subspace analysis. Dans *International Conference on Music Information Retrieval (ISMIR)*, Barcelona, Spain, octobre 2004.

[Zaffalon et Hutter, 2002] M. ZAFFALON et M. HUTTER. Robust feature selection by mutual information distributions. Dans *18th Conference on Uncertainty in Artificial Intelligence*, 2002.

[Zhou et Chellappa, 2006] S. ZHOU et R. CHELLAPPA. From sample similarity to ensemble similarity : probabilistic distance measures in reproducing kernel hilbert space. *IEEE Transactions on pattern analysis and machine intelligence*, 2006. to be published.

Table des figures

Liste des tableaux

Liste des Algorithmes

Index