DOCTORAT EN CO-ACCREDITATION
TELECOM SUDPARIS ET L'UNIVERSITE EVRY VAL D'ESSONNE

Spécialité :
INFORMATIQUE

Ecole doctorale : Sciences et Ingénierie

Présentée par

M. Mohamed El Amine SEHILI

Pour obtenir le grade de
DOCTEUR DE TELECOM SUDPARIS

RECONNAISSANCE DES SONS DE L'ENVIRONNEMENT DANS UN CONTEXTE DOMOTIQUE

Soutenue le 05 Juillet 2013 devant le jury composé de :

Rapporteurs

M. Jacques DEMONGEOT, HDR, Professeur des Universités, *Laboratoire AGIM, La Tronche*
M. Jean-François BONASTRE, HDR, Professeur des Universités, *LIA, Avignon*

Examinateurs

Mme. Laurence DEVILLERS, HDR, Professeur des Universités, *LIMSI, Orsay*
M. Dan ISTRATE, HDR, Professeur, *ESIGETEL- ALLIANSTIC, Villejuif* ***(Encadrant)***
M. Jérôme BOUDY, Doctorat, Ingénieur d'étude, *Telecom SudParis, Evry* ***(Encadrant)***
M. Thierry JOUBERT, Ingénieur, *THEORIS, Paris*

Directeur de thèse

Mme. Bernadette DORIZZI, HDR, Professeur, *Telecom SudParis, Evry*

Thèse n° 2013TELE0014

À ma famille.

À tous ceux que j'aime.

À tous ceux qui ont participé de près ou de loin à ma réussite...

TABLE DES MATIÈRES

RÉSUMÉ

Dans beaucoup de pays du monde, on observe une importante augmentation du nombre de personnes âgées vivant seules. Depuis quelques années, un nombre significatif de projets de recherche sur l'assistance aux personnes âgées ont vu le jour. La plupart de ces projets utilisent plusieurs modalités telles que la vidéo, le son, la détection de chute, etc. pour surveiller l'activité de la personne et lui permettre de communiquer naturellement avec sa maison, dite « intelligente », et, en cas de danger, lui venir en aide au plus vite.

Ce travail a été réalisé dans le cadre du projet ANR (Agence Nationale de la Recherche) VERSO, projet de recherche industrielle, SWEET-HOME. Les objectifs du projet sont de proposer un système domotique permettant une interaction naturelle (par commande vocale et tactile) d'une personne âgée avec sa maison, tout en lui procurant plus de sécurité par la détection des situations de détresse. Dans ce cadre, l'objectif premier de cette thèse est de proposer des solutions pour la reconnaissance des sons de la vie courante dans un contexte réaliste.

La reconnaissance du son fonctionnera en amont d'un système de Reconnaissance Automatique de la Parole. La fiabilité de la séparation entre parole et autres sons de l'environnement est très importante pour les performances de la reconnaissance de la parole. Par ailleurs, une bonne reconnaissance de certains sons, complétée éventuellement par d'autres sources informations (détection de présence, détection de chute, etc.) permettrait de bien suivre les activités de la personne et de détecter ainsi les situations de danger.

Dans un premier temps, nous nous sommes intéressés aux méthodes en provenance d'un domaine voisin, celui de la Reconnaissance et Vérification du Locuteur. Dans cet esprit, nous avons testé des méthodes basées sur GMM (*Gaussian Mixture Models*) et SVM (*Support Vector Machines*). Nous avons, en particulier, testé un noyau SVM utilisé pour la classification de séquences, appelé SVM-GSL (SVM GMM *Supervector Linear Kernel*). SVM-GSL est une méthode combinant SVM et GMM qui consiste à transformer une séquence de vecteurs de longueur arbitraire en un seul vecteur de très grande dimension appelé Super Vecteur. Le Super Vecteur est ensuite utilisé en entrée d'un SVM. Les expérimentations ont été menées en utilisant une base de données créée localement (18 classes de sons, pour plus de 1000 enregistrements), puis le corpus du projet SWEET-HOME, en intégrant notre système de reconnaissance dans un système plus complet comportant la détection du son provenant de plusieurs canaux, ainsi qu'un système de reconnaissance de la parole.

Ces premières expérimentations ont toutes été réalisées en utilisant un seul type de coefficients acoustiques, les *MFCC*. Par la suite, nous nous sommes penchés sur l'étude d'autres familles de

coefficients en vue d'en évaluer l'utilisabilité en reconnaissance des sons de l'environnement. Notre motivation fut de trouver des représentations qui soient plus simples et/ou plus efficaces que les coefficients *MFCC* pour la reconnaissance du son.

En utilisant 15 familles différentes de coefficients acoustiques, nous avons également expérimenté deux approches pour transformer une séquence de vecteurs en un seul vecteur, utilisable avec un SVM linéaire. La première consiste à calculer un ensemble de coefficients statistiques pour chaque coefficient acoustique, et à remplacer ainsi la séquence de valeurs par un nombre fixe de coefficients statistiques. La seconde approche, qui constitue l'une des contributions nouvelles de ce travail, fait appel à une méthode de discrétisation pour trouver, pour chaque caractéristique d'un vecteur de coefficients acoustiques, le(s) meilleur(s) point(s) de découpage permettant d'associer une classe donnée à un ou plusieurs intervalles de valeurs de la caractéristique. La probabilité de la séquence est estimée par rapport à chaque intervalle. Les vecteurs de probabilités obtenus (un pour chaque caractéristique) sont empilés pour construire un seul vecteur. Celui-ci remplacera la séquence de vecteurs acoustiques et sera utilisé en entrée d'un SVM.

Les résultats obtenus montrent que certaines familles de coefficients sont effectivement plus adaptées pour reconnaître certaines classes de sons. En effet, pour la plupart des classes de sons de notre base, les meilleurs taux de reconnaissance ont été observés avec une ou plusieurs familles de coefficients, autres que les *MFCC*. Certaines de ces familles sont, de surcroît, moins complexes que les *MFCC* (16 caractéristiques par fenêtre d'analyse sont extraites), à l'image du *Spectral Slope*, et du *Spectral Roll-Off* (une seule caractéristique par fenêtre pour chacune des deux familles).

Mots clefs

Reconnaissance des sons de l'environnement, SVM, GMM, noyaux de discrimination de séquences, coefficients acoustiques, méthodes de discrétisation.

ABSTRACT

In many countries around the world, the number of elderly people living alone has been increasing. In the last few years, a significant number of research projects on elderly people monitoring have been launched. Most of them make use of several modalities such as video streams, sound, fall detection and so on, in order to monitor the activities of an elderly person and to supply them with a natural way to communicate with their house, said smart-home, and to render assistance in case of an emergency.

This work is part of the ANR (Agence Nationale de la Recherche) VERSO project, an Industrial Research project, SWEET-HOME. The goals of the project are to propose a domotic system that enables a natural interaction (using touch and voice command) between an elderly person and their house and to provide them a higher safety level through the detection of distress situations. Thus, the main aim of this thesis is to come up with solutions for sound recognition of daily life in a realistic context.

Sound recognition will run prior to an Automatic Speech Recognition system. The reliability of the separation between speech and non-speech environmental sounds is very important for the speech recognition's performances. Furthermore, a good recognition of a few kinds of sounds, possibly complemented by other sources of information (presence detection, fall detection, etc.) could allow for a better monitoring of the person's activities and a better detection of dangerous situations.

We first had been interested in methods from a neighboring field, Speaker Recognition and Verification. As part of this, we have experimented methods based on GMM (*Gaussian Mixture Models*) and SVM (*Support Vector Machines*). We had particularly tested a Sequence Discriminant SVM kernel called SVM-GSL (SVM GMM *Super Vector Linear Kernel*). SVM-GSL is a combination of GMM and SVM whose basic idea is to map a sequence of vectors of an arbitrary length into one high dimensional vector called Super Vector. The Super Vector is then used as an input of an SVM. Experiments had been carried out using a locally created sound database (containing 18 sound classes for over 1000 records), then using the SWEET-HOMEproject's corpus. Our daily sounds recognition system was integrated into a more complete system that also performs sound detection over multi-channel audio streams, as well as speech recognition.

These first experiments had all been performed using one kind of acoustical coefficients, *MFCC* coefficients. Thereafter, we focused on the study of other families of acoustical coefficients. The aim of this study was to assess the usability of other acoustical coefficients for environmental sounds recognition. Our motivation was to find a few representations that are simpler and/or more

effective than the *MFCC* coefficients for sound classification.

Using 15 different acoustical coefficients families, we have also experimented two approaches to map a sequence of vectors into one vector, usable with a linear SVM. The first approach consists of computing a set of statistical coefficients for each acoustical feature and hence substituting a sequence of values by a fixed number of statistical coefficients. The second one, which is one of the novel contributions of this work, makes use of a discretization method to find, for each feature within an acoustical vector, the best cut point(s) that associate(s) a given class with one or many intervals of values. The likelihood of the sequence is estimated for each interval. The obtained likelihood vectors (one for each feature) are stacked to build one single vector. This vector replaces the sequence of acoustical vectors and is used as an input of an SVM.

The obtained results show that a few families of coefficients are actually more appropriate to the recognition of some sound classes. For most sound classes in our database, we noticed that the best recognition performances were obtained with one or many families other than *MFCC*. Moreover, a few of these families, such as the *Spectral Slope* or the *Spectral Roll-Off* are less complex than *MFCC*. Each of the former two families is actually a one-feature per frame acoustical family, whereas *MFCC* coefficients contain 16 features per frame.

Keywords

Environmental sound recognition, SVM, GMM, sequence discriminant kernels, acoustical coefficients, discretization methods.

ZUSAMMENFASSUNG

In vielen Ländern der Welt wird die Anzahl der alten, alleinlebenden Menschen immer größer. In den letzten Jahren sind zahlreiche Forschungsprojekte zum Thema der Überwachung älterer Personen gestartet worden. Die meisten dieser Projekte verwenden mehrere Modalitäten, wie zum Beispiel Video-Streams, Sound, Sturzerkennung usw. um die Tätigkeiten der älteren Person zu überwachen, ihr eine natürliche Art der Kommunikation mit ihrem Haus (auch intelligente Haustechnik genannt) anzubieten, und in Notfällen Hilfe zu verständigen.

Die hier beschriebene Arbeit ist Teil des ANR (Agence Nationale de la Recherche) VERSO Projektes, eines industriellen Forschungsprojekts, SWEET-HOME. Das Ziel des Projektes ist es, eine Hausautomation (Domotic System) zu entwickeln, die eine natürliche Interaktion (durch Sprachsteuerung und Touch Eingabe) zwischen einer älteren Person und ihrem Haus ermöglicht und welche des Weiteren durch die automatische Erfassung von Notsituationen eine bessere Sicherheit im Alltag gewährleistet. Das Hauptziel der hier beschriebenen Doktorarbeit ist im Speziellen die Erarbeitung von Lösungen, die es ermöglichen, Geräusche des täglichen Lebens in einem realistischen Kontext zu erkennen.

Geräuscherkennung wird eingangsseitig von einem Spracherkennungssystem durchgeführt. Die Zuverlässigkeit der Trennung zwischen einem eigentlichen Sprachlaut und den anderen umgebenden Geräuschen ist für die Leistung der Spracherkennung daher sehr wichtig. Darüber hinaus erlaubt die gute Erkennung einiger Geräuscharten, welche möglicherweise auch durch andere Informationsquellen komplementiert werden kann (z.B. Sturzdetektion, Präsenzerkennung usw.), eine bessere Überwachung der Tätigkeiten der Person sowie eine bessere Erkennung von potentiellen Notfällen.

Zunächst einmal waren wir an Methoden aus einem benachbarten Bereich interessiert, nämlich dem der Sprecherauthentifizierung. In diesem Zusammenhang haben wir mit Methoden experimentiert, die auf GMM (*Gaussian Mixture Models*) und SVM (*Support Vector Machines*) basieren. Wir haben insbesondere einen Kernel zur Klassifizierung von Sequenzen getestet, den so genannten SVM-GSL (SVM GMM *Super Vector Linear Kernel*). SVM-GSL ist eine Kombination von SVM und GMM, deren Grundidee es ist, eine Sequenz beliebiger Länge in einem einzigen Vektor abzubilden, dem Super-Vektor. Der Super-Vektor wird dann als Datengrundlage für eine SVM verwendet. Die Experimente sind zunächst unter Einsatz einer lokalen Datenbank (18 Sounds-Klassen und mehr als 1000 Audioaufzeichnungen) und danach mit dem SWEET-HOMEProjekt Korpus ausgeführt worden. Unser System zur Erkennung täglicher Geräusche wurde schlussendlich in ein vollständigeres System integriert, das sowohl Mehrkanal-Sound-Detektion über verschiedene Audio-Streams als

auch Spracherkennung ermöglicht.

All unsere ersten Experimente wurden unter Nutzung von einer einzigen Art von akustischen Koeffizienten durchgeführt – den *MFCC*-Koeffizienten. Danach haben wir uns mit der Untersuchung anderer Familien von akustischen Koeffizienten beschäftigt. Der Zweck dieser Untersuchung war es, die Verwendbarkeit anderer akustischen Koeffizienten für die Erkennung von Umgebungsgeräuschen zu evaluieren. Unsere Motivation war, einige Repräsentation zur Sound-Erkennung zu finden, die einfacher und/oder wirksamer als *MFCC* sind.

Neben der Benutzung von 15 verschiedenen Familien akustischer Koeffizienten haben wir auch mit zwei Ansätzen zur Abbildung einer Sequenz von Vektoren in einem einzigen Vektor (welcher mit einer linearen SVM zu verwenden ist) experimentiert. Der erste Ansatz bestand darin, für jedes akustische Merkmal eine Anzahl von statistischen Koeffizienten zu errechnen, und folglich eine Sequenz von Werten durch eine feste Anzahl von statistischen Koeffizienten zu ersetzen. Der zweite Ansatz, welcher ein Novum dieser Arbeit darstellt, nutzt eine Diskretisierungsmethode aus, um für jedes Merkmal eines akustischen Vektors jene Schnittpunkte (cut points) zu finden, welche eine bestimmte Klasse mit einem oder mehreren Intervallen am besten assoziieren. Die Wahrscheinlichkeit von jeder Sequenz wird für jedes Intervall geschätzt. Die sich ergebenden Wahrscheinlichkeitsvektoren (einer für jedes Merkmal) werden dann gestapelt, um einen einzigen Vektor zu errichten. Die Sequenz wird durch diesen Vektor ersetzt und folglich als Eingabe für eine SVM verwendet.

Die erzielten Ergebnisse zeigen, dass manche Koeffizientenfamilien tatsächlich besser zur Erkennung mancher Klassen von Geräuschen geeignet sind. Für die meisten Geräusche in unserer Datenbank haben wir festgestellt, dass die besten Erkennungsleistungen mit einigen Koeffizientenfamilien erzielt werden können, die sich von *MFCC* unterscheiden. Einige dieser Familien, wie beispielsweise das *Spectral Slope* oder das *Spectral Roll-Off*, sind auch einfacher anzuwenden als *MFCC*. Diese beiden hier genannten Familien verwenden zum Beispiel nur ein Merkmal pro Frame, während die *MFCC*-Koeffizienten 16 Merkmale pro Frame benötigen.

Schlagwörter

Umgebungsgeräuscherkennung, SVM, GMM, Sequenzdiskriminante Kernels, akustische Koeffizienten, Diskretisierungsmethoden.

ملخص

يلاحظ في العديد من دول العالم ازدياد محسوس لعدد الأشخاص المسنين الذين يعيشون بمفردهم. منذ عدة سنوات يشهد عدد مشاريع البحث العلمي التي تُعنى بكبار السن ارتفاعا ملحوظا. إن الغالبية من هذه المشاريع تلجأ إلى استعمال العديد من أنماط البيانات كالفيديو و الكشف عن سقوط الأشخاص و غيرها بغرض مراقبة نشاط الشخص و تمكينه من التفاعل مع منزله الذكي و توفير المساعدة له بأسرع ما يمكن في حال تعرضه لخطر ما.

إن العمل الذي بين أيدينا قد تم إنجازه في إطار مشروع بحث علمي صناعي من صنف «فارسو»، برعاية من الوكالة الوطنية للبحث العلمي بفرنسا، يدعى «سويت هوم». أهداف المشروع تتمثل في توفير نظام منزلي آلي يسمح للشخص المسن بالتفاعل بشكل طبيعي مع بيته (عن طريق الأوامر الصوتية و كذا عن طريق اللمس) كما يعمل على توفير مستوى عال من الأمن عن طريق الكشف الآلي عن حالات الإستغاثة. في هذا الإطار فإن المبتغى الأساسي لمذكرة الدكتوراه هذه هو إقتراح حلول بغرض التعرف على أصوات الحياة العامة في سياق واقعي.

إن التعرف على أصوات المحيط موجه للعمل كخطوة آنية لنظام آخر يُعنى بتحويل الكلام الى نص بشكل آلي. من هنا فإن موثوقية الفصل بين الكلام و ما سواه من الأصوات البيئية تكتسي أهمية حيوية بالنسبة لأداء نظام التعرف على الكلام. من جانب آخر، فإن التعرف الدقيق على بعض أصناف الأصوات، و الذي يمكن تكملته بمصادر أخرى للمعلومات (كالكشف عن الحضور، الكشف عن السقوط و غيرهما) يمكنه المساعدة على تتبع نشاطات الشخص المختلفة و بالتالي الكشف عن حالات الخطر.

في بادىء الأمر، كان تركيزنا منصباً على الطرق و الأساليب التي تم تطويرها في مجال مجاور لمجال عملنا هذا، إنه مجال التعرف على شخصية المتكلم. في هذا الصدد، قمنا باختبار أساليب مبنية على نماذج خليط غاوس («جي أم أم») و أخرى على أشعة الدعم الألي («أس في أم»). على وجه الخصوص قمنا باستعمال نواة «أس في أم» موجهة لتصنيف التسلسلات تدعى «أس في أم - جي أس أل». يعتمد أسلوب ال«أس في أم - جي أس أل» على دمج طريقتي «أس في أم» و «جي أم أم» و يتلخص في تحويل سلسلة من الأشعة ذات طول اعتباطي الى شعاع وحيد ذو بعد هائل يدعى الشعاع الممتاز. يتم بعدها استعمال الشعاع الممتاز كبيانات مدخلة لل«أس في أم». الإختبارات التي قمنا بها تمت باستعمال قاعدة بيانات تم إنشاؤها محليا (في المختبر، تحتوي على ثمانية عشر صنفا من الأصوات و أكثر من ألف تسجيل صوتي) ثم باستعمال قاعدة البيانات التي تم تسجيلها في إطار مشروع «سويت هوم» و هذا بدمج نظامنا (للتعرف على الأصوات) في نظام مكتمل يضم أيضا الكشف عن الأصوات الصادرة عن عدة قنوات و نظاما آخر للتعرف على الكلام.

مجموعة الإختبارات الأولى هذه تمت كلها باستعمال صنف واحد من المعاملات الصوتية، معاملات «أم أف سي سي». في المرحلة التالية عكفنا على دراسة عائلات أخرى من المعاملات الصوتية بغرض اختبار إمكانية استعمالها للتعرف على الأصوات المحيطة. في هذا الشأن كان المحفز الأساسي هو إيجاد معاملات أقل تعقيدا و \ أو أكثر فاعلية من ال«أم أف سي سي» للتعرف على الأصوات.

باستعمال خمسة عشر عائلة مختلفة من المعاملات الصوتية، قمنا أيضا باختبار طريقتين أخرتين لتحويل سلسلة من الأشعة إلى شعاع وحيد يمكن استعماله مع «أس في أم» خطي. الطريقة الأولى تعتمد على حساب عدد من المعاملات الإحصائية لكل من المعاملات الصوتية و استبدال سلسلة الأشعة بشعاع وحيد يتكون من عدد ثابت من المعاملات الإحصائية. الطريقة الثانية، و التي تشكل إحدى المساهمات العلمية لهذا العمل، تستعين بطريقة لتحويل القيم المستمرة إلى قيم متقطعة و ذلك لإيجاد نقاط التقطيع لكل خاصية في الشعاع و التي تسمح بربط فئة معينة من المعطيات بواحد أو أكثر من المجالات المنتمية لهذه الخاصية. يتم بعدها تقدير احتمال تواجد السلسلة بكل مجال و استعمال هذه القيم الجديدة، أي الإحتمالات، لتشكيل شعاع جديد يعوض السلسلة بأكملها. النتائج المتحصل عليها تظهرفعلا أن بعض العائلات من المعاملات الصوتية أنسب من غيرها للتعرف على

بعض الفئات من الأصوات. في الواقع، فإنه بالنسبة لجل فئات الأصوات في قاعدة البيانات المستعملة تم الحصول على أعلى معدلات التعرف باستعمال واحدة أو أكثر من عائلات المعاملات الصوتية التي تختلف عن ال«أم أف سي سي». إضافة إلى ذلك، فإن بعض هذه المعاملات تعتبر أقل تعقيدا من ال «أم أف سي سي» (ستة عشر خاصية لكل نافذة تحليل لهذه العائلة) كال«سبيكترال سلوب» و كذا ال«سبيكترال رول أوف» (خاصية وحيدة لكلتيهما في النافذة).

الكلمات الرئيسية

التعرف على الأصوات المحيطة، «أس في أم»، «جي أم أم»، نواة للتمييز بين السلسلات، معاملات صوتية، طريقة لتحويل القيم المستمرة إلى قيم متقطعة.

REZUMAT

In multe tari din lume [1], observam o marire importanta a numarului de persoane in varsta traind singure. De mai multi ani, un numar semnificativ de proiecte de cercetare pe tematica asistentei persoanelor in varsta au luat nastere. Marea majoritate a acestor proiecte utilizeaza mai multe modalitati precum : imaginea, sunetul, detectarea caderii, etc. pentru a supraveghea activitatile persoanei si a lui permite sa comunice in mod natural cu "casa inteligenta" si, in caz de pericol, sa fie ajutat foarte rapid.

Aceste activitati de cercetare au fost realizate in cadrul proiectului ANR (Agentia Nationala de Cercetare) VERSO pe nume SWEET-HOME (proiect de tip cercetare industriala). Obiectivele acestui proiect sunt de a propune un sistem domotic permitand o interactiune naturala (prin intermediul comenzii vocale si tactile) intre o persoana in varsta si casa sa, asigurandui securitatea prin detectia automata a situatiilor de urgenta. In acest cadru, primul obiectiv al acestei teze de doctorat est de a propune solutii pentru recunoasterea sunetelor vietii curente intr-un context realist.

Recunoasterea sunetelor va functiona in amontul unui sistem de recunoastere automata a vorbirii. Fiabilitatea separarii intre vorbire si celelalte sunete ale mediului este foarte importanta pentru performantele sistemului de recunoastere a vorbirii. De altfel, o buna recunoastere a anumitor sunete, completata de alte surse de informatie (detectia prezentei, detectia caderii, etc.) ar permite o urmarire fiabila a activitatilor persoanei si astfel o detectare a situatiilor de urgenta.

Intr-o prima etapa, autorul a studiat metodele domeniilor vecine cum ar fi cel al recunoasterii si verificarii vocii unei persoane. Metode bazate pe GMM (*Gaussian Mixture Models*) si SVM (*Support Vector Machines*) au fost evaluate. Autorul a evaluat in mod particular o metoda bazata pe un nucleu SVM utilizata pentru clasificarea secventelor numita SVM-GSL (SVM GMM *Super Vector Linear Kernel*). SVM-GSL este o metoda care combina SVM si GMM si care consta in a transforma o secventa de vectori de lungime arbitrara intr-unul singur vector de dimensiune mare numite Super-Vector. Super-Vectorul este utilizat la intrarea unui SVM. Experimentarile au fost realizate utilizand o baza de date creata in laborator (18 clase de sunete, mai mult de 1000 de inregistrari) si mai tarziu utilizand corpusul proiectului SWEET-HOME, integrand sistemul de recunoastere a sunetelor intr-un sistem complet compus din detectia sunetelor provenind de la mai multe canale si sistemul de recunoastere a vorbirii.

Aceste prime evaluari au fost realizate folosind un singur tip de parametrii acustici : *MFCC*. In continuare, autorul a studiat alte familii de coeficienti pentru a ii utiliza in recunoasterea sunetelor.

1. Ce résumé en Romain a été écrit par mon cher encadrant Mr Dan Istrate.

Scopul a fost de a gasi reprezentari mai simple dar mai eficace decat *MFCC* pentru recunoasterea sunetelor.

Utilizand 15 familii de coeficienti diferiti, 2 metode de transformare a unei secvente de vectori intr-unul singur utilizabil cu un SVM linear au fost studiate. Prima metoda consta in a calcula un ansamblu de coeficienti statistici pentru fiecare coeficient acustic si a inlocui secventa de vectori cu un numar fix de coeficienti statistici. Metoda a 2a, care este una din contributiile originale ale acestei lucrari, utilizeaza o metoda de discretizare pentru a gasi, pentru fiecare caracteristica a unui vector de coeficienti acustici, punctele cele mai adaptate de decupare permitand asocierea unei clase date la unul sau mai multe intervale de valori ale caracteristicii. Probabilitatea secventei este estimata in raport cu fiecare interval. Vectorii de probabilitate obtinuti (unul pentru fiecare caracteristica) sunt concatenati pentru a obtine un singur vector. Acest vector va inlocui secventa de vectori acustici si va fi utilizat in intrarea SVM.

Rezultatele obtinute demonstreaza ca anumite familii de coeficienti sunt in mod efectiv mai adaptate pentru a recunoaste anumite clase de sunete. In sfarsit, pentru majoritatea claselor de sunete din baza testata, cele mai bune rapoarte de recunoastere au fost identificate utilizand o sau mai multe familii de coeficienti diferite de *MFCC*. Anumite dina ceste familii sunt mai putin complexe deact *MFCC* (16 coeficienti pentru fiecare fereastra de analiza) ca de exemplu *Spectral Slope* sau *Roll-off point* (un singur coeficient pentru fiecare fereastra).

Cuvinte cheie

Recunoasterea sunetelor mediului, SVM, GMM, nuclee de discriminare de secvente, coeficienti acustici, metode de discretizare.

LISTE DES FIGURES

LISTE DES TABLEAUX

CHAPITRE 1

INTRODUCTION

L'intégration des nouvelles technologies à notre vie quotidienne a réalisé un saut spectaculaire au cours des trois dernières décennies. Il y a un peu plus de trente ans, le seul moyen pour une personne d'être reliée à un ordinateur passait par l'autorisation de se connecter à une machine installée dans un centre de calcul, qui se trouvait dans une université ou dans une grande entreprise.

Au fil des années, les ordinateurs ont connu d'importantes baisses de prix et de taille, avec, parallèlement, une puissance de calcul en constante croissance. L'apparition de la Micro-informatique ne les a pas seulement rendus accessibles aux foyers mais en a fait un équipement important au sein de chaque maison, voire dans chaque pièce de la maison. Par ailleurs, ces évolutions des ordinateurs en termes de taille, de coût et de performances, ont été un facteur déterminant pour le type d'applications qui pouvaient être développées. Un feu de signalisation ou un petit jouet pour enfants, par exemple, fonctionnent parfaitement avec un processeur embarqué de seulement 4 bits qui, de plus, ne coûte qu'une somme dérisoire de nos jours. Avec l'arrivée des machines plus puissantes, de nouvelles applications bien plus complexes et plus gourmandes en ressources (jeux vidéo, applications multimédia, etc.) ont vu le jour.

L'évolution des téléphones portables a pris un chemin comparable à celui des ordinateurs. Les fonctionnalités d'un téléphone portable de première génération, par exemple, se résumaient au fait de passer ou de recevoir des appels téléphoniques et, au mieux, en un réveil ou un simple jeu. Les téléphones portables d'aujourd'hui, appelés plus volontiers **smartphones** ou **téléphones intelligents**, disposent de performances comparables à celles d'un ordinateur de bureau des années 90, et sont munis d'un nombre foisonnant d'applications.

L'évolution des applications informatiques n'est pas seulement liée à la puissance des ordinateurs mais aussi aux besoins des usagers et aux avancées réalisées en télécommunications et en technologies de l'information. L'apparition d' applications ayant trait à l'**intelligence artificielle** (reconnaissance de la parole, systèmes de dialogue homme-machine, etc.) a également contribué à l'intégration des ordinateurs et des smartphones dans notre vie quotidienne.

En dépit de cette présence massive des nouvelles technologies dans notre vie, il nous est parfois à peine possible (et intéressant) de les percevoir. D'après Weiser [Weiser, 1991], les technologies

les plus profondes sont celles qui disparaissent dans le tissu de la vie quotidienne jusqu'à en faire partie. Un exemple typique de la vision de Weiser est le concept de **maison intelligente** (*Smart Home*), qui doit interagir avec ses occupants de façon efficace et transparente.

1.1 Maison Intelligente

Dans [Chan et al., 2009], une maison intelligente est définie comme une résidence équipée de technologies permettant de surveiller ses occupants, de contribuer à leur l'indépendance et de les maintenir en bonne santé. La définition donnée par l'Intertek [1] est la suivante : un habitat équipé d'un réseau de communication permettant de connecter les équipements clés et les services, et offrant la possibilité d'y accéder, de les contrôler ou de les surveiller à distance.

Dans [Yuan and Peng, 2012], une maison intelligente est une combinaison de plusieurs technologies avancées. Elle consiste à incorporer de très petites puces, qui possèdent des capacités de communication sans-fil, de perception et de traitement de l'information, dans les articles à usage quotidien. Le but étant de créer un environnement informatique transparent pour l'habitant. Pour être en mesure de fournir des services, le système doit être capable d'acquérir, de traiter, et de transmettre l'information à tout moment. Il doit également être capable de comprendre les besoins de l'usager et de contrôler les différents équipements de façon intelligente, afin de rendre l'environnement plus confortable. De plus il doit permettre de réduire la consommation d'énergie sans influencer les habitudes de l'habitant.

D'après [Spencer, 2000], une maison intelligente utilise des dispositifs basiques avec des capacités de communication pour construire un environnement où plusieurs opérations seront automatisées. Une communication efficace entre les différents dispositifs implique, pour un élément donné, la possibilité d'envoyer des requêtes à d'autres éléments, leur demandant d'exécuter certaines fonctions si un certain nombre de conditions sont réalisées. De cette façon, plusieurs dispositifs séparés peuvent être organisés et programmés pour exécuter, ensemble, des fonctions plus complexes.

1. http ://www.intertek.com/

1.2 Assistance aux personnes âgées

Depuis quelques années, de nombreux travaux de recherche s'intéressent à la conception de maisons intelligentes pour une population bien spécifique, celle des personnes âgées. Les individus de cette population vivent souvent seuls et soufrent de diverses pathologies liées à l'âge.

Le rapport qualité/prix de plusieurs types de capteurs (caméras, microphones, capteurs infra-rouge, etc.) et l'émergence des technologies de l'information ont rendu de plus en plus intéressante l'idée d'équiper les maisons des personnes âgées de capteurs et de dispositifs de communication afin de surveiller leur activité et de prévenir et signaler toute situation anormale nécessitant une intervention extérieure telle qu'une chute, une longue période d'inactivité ou un message de détresse. Cela permettrait également de réduire les coûts des soins et d'alléger la charge des personnes qui doivent intervenir en cas de danger (membres de la famille, infirmiers, médecins, etc.) en réduisant le nombre de déplacements inutiles. La plupart des projets de recherche existants se fixent ces éléments comme objectifs de base. D'autres projets, plus ambitieux, ont également pour objectifs d'offrir à la personne une vie sociale plus riche, en facilitant notamment la communication avec les membres de la famille, et en permettant à la personne d'interagir avec son environnement via des interfaces adaptées et ergonomiques. Ces solutions devraient être idéales pour les personnes âgées souffrant de différents niveaux d'handicap en [Spencer, 2000] :

— Surveillant l'environnement pour s'assurer de la sécurité de l'individu,
— Rendant automatiques certaines tâches de la vie quotidienne, difficiles ou impossibles à exécuter par la personne,
— Prévenant les proches en cas de danger,
— Maintenant l'individu dans un état actif (exercices cognitifs, communication, etc.),
— Facilitant la réhabilitation de la personne (incitations visuelles ou auditives).

En dépit de l'aspect pratique plus ou moins attrayant de ces solutions, deux problèmes importants se posent : le problème d'éthique et celui d'acceptabilité. En effet, beaucoup de personnes n'accepteraient pas d'être surveillées chez elles en permanence par des caméras ou simplement de mettre leur maison sous écoute. D'autres seraient également réticentes quant au port de dispositifs électroniques. Par ailleurs, ces systèmes peuvent facilement devenir très complexes dès que l'on se met à ajouter davantage de modalités, qui doivent, de surcroît, travailler en concert et interagir avec l'habitant. Enfin, diverses technologies, impliquant des compétences de plus d'un domaine (électronique, informatique, médecine, sciences sociales, ergonomie, etc.), sont nécessaires pour ce

type de projets.

1.3 Contexte et motivations de ce travail

1.3.1 Projet SWEET-HOME

Cette thèse fait partie du projet de Recherche Industrielle VERSO SWEET-HOME, financé par l'Agence Nationale de la Recherche (ANR). Le projet a démarré en novembre 2009 et a pris fin en mai 2013. Les objectifs du projet sont les suivants :

— Permettre aux personnes âgées vivant seules d'interagir naturellement avec leur lieu de vie. L'interaction naturelle est réalisée par commande vocale et tactile.
— Augmenter leur sécurité par la détection de situations de détresse.

La partie la plus substantielle du projet est bâtie autour des technologies audio. Dans ce contexte, deux problématiques de recherche sont abordées :

1. Reconnaissance des sons de l'environnement dans une maison intelligente.
2. Reconnaissance de la parole pour personnes âgées.

1.3.2 Thèse dans le contexte du projet

Objectif principal : L'objectif principal de cette thèse peut être énoncé comme suit :

Fournir des solutions pour séparer la parole des autres sons de l'environnement dans un contexte réaliste (maison intelligente), et pour reconnaître ces sons en prenant en considération l'existence de plusieurs canaux audio.

Défis : Pourquoi la reconnaissance des sons de l'environnement constitue-elle un domaine de recherche particulièrement complexe ?

La complexité du domaine peut être attribuée à plusieurs facteurs :

Premièrement, il n'y a nul doute que le nombre des sons de l'environnement est très élevé. En effet, plus le nombre de classes de sons augmente, plus la complexité du système utilisé pour les reconnaître est importante. En **reconnaissance automatique du locuteur** (RAL) et en **reconnaissance automatique de la parole** (RAP), on ne s'intéresse qu'à un seul type de signal, celui de la parole. Bien entendu, cela ne signifie point que ces deux domaines sont moins complexes, mais en reconnaissance des sons de l'environnement on s'intéresse à des types de sons qui peuvent être très différents, ce implique d'étudier les caractéristiques de plusieurs types de signaux. Cela étant dit, ce même facteur pourrait constituer un bon élément pour discriminer certains types de sons.

Deuxièmement, les variations intra-classes de sons (différences entre événements acoustiques appartenant à une même classe de sons), et les similitudes inter-classes constituent l'un des problèmes majeurs en reconnaissance des sons de l'environnement. Les variations intra-classes sont pratiquement inévitables car elles sont souvent dues à des facteurs extérieurs difficiles, voire impossibles à contrôler. Pour la classe de sons « fermeture de porte », par exemple, des différences entre les enregistrements (variations intra-classes) plus ou moins importantes peuvent surgir du fait que les portes elles-mêmes, l'environnement où elles se trouvent, la manière dont on les ferme ou bien le matériel utilisé pour l'enregistrement sont différents. Par ailleurs, certains sons appartenant à deux ou plusieurs classes différentes peuvent être difficiles à distinguer, même pour une écoute à l'oreille humaine.

Troisièmement, l'existence du bruit de l'environnement est un facteur handicapant. Par bruit, nous entendons tout type de sons qui n'a pas d'intérêt particulier pour l'application et qui peut tout simplement être ignoré. Il s'agit d'un problème particulièrement délicat car, en présence du bruit, le signal du son d'intérêt peut subir des modifications plus ou moins importantes. Dans des conditions réalistes, le bruit est omniprésent. Par ailleurs, deux ou plusieurs sons d'intérêt peuvent avoir lieu au même moment et peuvent, de ce fait, se chevaucher partiellement ou complètement et se masquer l'un l'autre.

Quatrièmement, ce domaine est moins étudié en comparaison d'autres domaines tels que la RAL, la RAP ou la reconnaissance de la musique. De plus, les efforts de recherche se trouvent répartis sur plusieurs sous-domaines. En effet, les travaux de recherche s'intéressent souvent à un nombre limité de sons, voire à un seul type de sons. Les bases de données et les techniques utilisées pour une application particulière ne sont pas toujours utilisables pour une autre application.

Hypothèses : Tout au long de cette thèse, nous avons posé un certain nombre d'hypothèses concernant la reconnaissance des sons de l'environnement. Ces hypothèses ont largement contribué aux directions que notre travail a prises. Les deux hypothèses les plus importantes sont les suivantes :

Hypothèse 1 : *Les méthodes utilisées en reconnaissance automatique du locuteur pourraient être utilisées pour la reconnaissance des événements acoustiques, du moins pour une première étude.*

Hypothèse 2 : *Les coefficients acoustiques* MFCC *ont été proposés pour la reconnaissance automatique de la parole et ont par la suite également été utilisés en reconnaissance automatique du locuteur. Rien ne suggère qu'ils soient les coefficients les plus appropriés pour la reconnaissance des autres sons de l'environnement.*

1.3.3 Contributions de la thèse

- Évaluation de quelques algorithmes de l'état de l'art de la RAL en reconnaissance des sons de l'environnement (GMM, SVM).
- Création d'une base de données contenant 18 classes de sons de la vie quotidienne.
- Utilisation d'un noyau SVM de discrimination de séquences, issu de la RAL, pour la reconnaissance des sons de l'environnement (comparaison avec des GMMs).
- Annotation du corpus du projet SWEET-HOME.
- Évaluation du noyau SVM de discrimination de séquences en utilisant une partie du corpus du projet SWEET-HOME(détection et reconnaissances des sons de l'environnement dans des flux audio multi-canaux continus).
- Utilisation de plusieurs familles de coefficients acoustiques (en plus des *MFCC*), utilisées notamment en reconnaissance de la musique, pour la reconnaissance des sons de l'environnement. Évaluation des performances de chaque famille de coefficients par rapport à chaque classe de sons.
- Proposition d'un nouveau noyau SVM de discrimination de séquences, basé sur une méthode de discrétisation.

1.4 Organisation de la thèse

Chapitre 2 : Ce chapitre définit notre problématique, et la situe par rapport aux domaines voisins. Nous parlerons en particulier de la particularité de la reconnaissance des sons de l'environnement en la comparant à la RAP, à la RAL et à la reconnaissance de la musique.

Chapitre 3 : Ce chapitre présente certains des travaux les plus intéressants en reconnaissance des sons de l'environnement et fait le lien avec les techniques les plus utilisées et leur domaine de provenance originel (RAP ou RAL).

Chapitre 4 : Ce chapitre présente les trois méthodes retenues, toutes basées sur des travaux en RAL (SVMs utilisant des vecteurs acoustiques, GMMs, SVMs à noyau de discrimination de séquence). Nous y décrirons la base de données de l'ESIGETEL ainsi que le corpus du projet SWEET-HOME. Nous présenterons et analyserons les résultats obtenus avec les deux bases de données.

Chapitre 5 : Dans ce chapitre nous étudions plusieurs familles de coefficients acoustiques et testons chaque famille séparément en utilisant des SVMs. Des tests incluant toutes les familles de coefficients sont également réalisés. Deux méthodes de transformation de séquences (autres que le noyau utilisé au chapitre 4, issu de la RAL) sont également présentées.

Chapitre 6 : Ce chapitre fait une courte synthèse du travail réalisé dans cette thèse, en rapporte les conclusions, et identifie les voies de recherche qui nous semblent les plus prometteuses.

1.5 Conventions

Les termes ou concepts en Français, mentionnés pour la première fois dans le texte, sont souvent en gras (exemple : **paysage sonore**), leur traduction Anglaise, si elle est utilisée, en italique (*Soundscape*). Les acronymes des algorithmes et des méthodes de calcul sont en Anglais, en majuscules (SVM, VQ, GMM, etc.). Les familles de coefficients acoustiques, en Anglais, en italique (*MFCC*, *Spectral Roll-Off*, etc.). Il en va de même pour les classes de sons (*Breathing*, *Water*, etc.).

CHAPITRE 2

RECONNAISSANCE DU SON

Ce chapitre traite de la reconnaissance du son et positionne notre problème par rapport aux domaines voisins. Il présente les classes de sons qui nous intéressent dans le cadre du projet SWEET-HOME, les coefficients acoustiques utilisés dans la littérature et certaines des méthodes de classification les plus courantes. Enfin, une discussion est proposée sur une exploitation possible des avancées réalisées dans d'autres domaines pour notre problème de reconnaissance des sons de l'environnement.

2.1 Introduction

Bien que l'intérêt pour la reconnaissance des sons de l'environnement ait commencé il y a deux décennies environ, notamment avec les travaux de Bregman [Bregman, 1990] [Bregman, 1994], il est aisé d'observer que l'écart entre ce qui a été atteint en reconnaissance automatique de la parole (RAP) et reconnaissance automatique du locuteur (RAL) d'une part, et en reconnaissance des sons de l'environnement d'autre part, s'est considérablement creusé au fil des années. Les systèmes de reconnaissance de la parole, initialement conçus dans le but de reconnaître des mots isolés (chiffres), ont graduellement progressé pour reconnaître des phrases lues, des émissions de radio, puis des conversations spontanées. Par la suite, d'autres aspects tels que l'accent et la prosodie ont été également étudiés.

Une autre comparaison avec la **vision par ordinateur** (*Computer Vision*) mène au même constat : la reconnaissance des sons, outre la parole, reste de loin moins explorée. Les travaux de Viola et Jones sur la détection d'objets [Viola and Jones, 2001] [Viola and Jones, 2004], par exemple, constituent aujourd'hui un bon exemple d'un problème de vision par ordinateur presque « résolu ». Des systèmes de détection du visage sont aujourd'hui intégrés dans les caméras numériques, les smartphones et les réseaux sociaux.

Il n'est pourtant pas difficile d'admettre que les sons de l'environnement encapsulent une quantité considérable d'informations dont nous nous servons dans la vie de tous les jours. Qu'il s'agisse d'une sonnerie de téléphone, d'un klaxon de voiture, d'un chant d'oiseau ou encore d'un cri humain, les sons autour de nous attirent notre attention et peuvent être informatifs au même niveau que la parole, voire plus dans certaines situations (une alarme incendie par exemple).

Par analogie à la vision par ordinateur, Richard Lyon parle de *Machine Hearing* ou **audition par ordinateur** [Lyon, 2010]. Cette notion fait référence à la capacité des machines à percevoir les sons de manière similaire à celle des être humains ; c'est-à-dire, la faculté de distinguer entre parole, musique et autres sons et d'appliquer un traitement approprié à chaque type de signal, de reconnaître la source et la provenance d'un son, et de ne tenir compte que des sons « intéressants ». Cela implique également le fait d'être en mesure d'apprendre les noms d'objets, de locuteurs, de genres musicaux, etc. et d'être capable de rechercher des sons désignés par ces noms dans de grandes bases de données. Enfin, cela implique également une capacité à reconnaître les événements acoustiques importants et à interagir avec l'environnement en temps réel, que ce soit dans une usine, lors d'un concert musical ou pendant une conversation téléphonique. Cette définition, d'une connotation quelque peu « futuriste », est bien plus large que celle proposée par Bregamn [Bregman, 1994], et qui sera traitée dans la section suivante.

John Treichler (*Applied Signal Technology Inc.*), identifie dans sa colonne du *Exploratory DSP*, *A View of the Future*, un certain nombre de domaines du traitement du signal qui se trouvent aujourd'hui « au milieu d'un long parcours de développement » [Treichler, 2009]. Parmi les domaines mentionnés certains sont en rapport direct avec le traitement du son. Ils incluent l'échographie, l'exploration sismique, la téléphonie sans fil, l'enregistrement et la compression de la musique, l'informatique embarquée dans les véhicules, la télé-présence, la reconnaissance et la synthèse de la parole et, enfin, la détection et la reconnaissance d'objets sous-marins (sonar). L'échographie, l'exploration sismique et le sonar n'ayant en fait pas de lien avec la perception humaine du son.

Pour bien situer le travail réalisé dans cette thèse, et éviter ainsi de se perdre dans les méandres des différentes applications, il convient ici de préciser notre objectif et de bien définir les frontières avec les autres domaines.

Dans ce travail, on s'intéresse à la **reconnaissance des événements acoustiques** (REA). On pourrait également indifféremment utiliser les termes : **reconnaissance des événements audio** ou **reconnaissance des sons de l'environnement**. Le domaine peut, au même titre que la reconnaissance automatique de la parole, de la reconnaissance du locuteur ou de la reconnaissance de la musique, se situer dans un domaine plus vaste qu'on appelle **reconnaissance du son**.

Le domaine de la REA est lui-même assez large et peine à se positionner clairement au sein des communautés du traitement du signal et celle de la reconnaissance des formes. Cela s'explique par les différents travaux et publications qui se trouvent éparpillés sur plusieurs champs applicatifs. C'est ainsi qu'on trouve des travaux sur la reconnaissance de chants d'oiseaux [Chen and Maher,

2006] [Somervuo et al., 2006], de bruit d'insectes [Le-Qing, 2011], de sons produits par d'autres animaux [Duan et al., 2011], de querelles [Andersson et al., 2010], de coups de feu [Chen et al., 2006a] [Valenzise et al., 2007], de types de respiration [Bahoura and Pelletier, 2004], d'événements acoustiques ayant lieu en salle de réunion [Temko and Nadeu, 2005], en salle de bain [Chen et al., 2005a] [Chen et al., 2005b], aux passages piétons [Lee and Rakotonirainy, 2011], au foyer d'une personne âgée (auxquels ce travail s'intéresse) etc. En raison de la nature différente des sons auxquels s'intéresse chacun des sous-domaines mentionnés, il est souvent difficile de comparer les méthodes de reconnaissance utilisées et encore plus d'utiliser des bases de données communes.

Dans ce travail, nous nous intéressons, bien entendu, à la reconnaissance des sons dans un contexte domotique. Plus particulièrement aux sons révélant une situation de détresse d'une personne âgée vivant seule. Pour cela, il convient ici de bien identifier les classes de sons qui peuvent avoir lieu dans un tel contexte et d'en repérer celles qui représentent un intérêt pour l'application. Ces aspects sont traités dans la section suivante.

2.2 Sons de l'environnement

Les sons intéressants pour un système de reconnaissance des événements acoustiques dépendent largement des applications. Chaque application est souvent conçue autour d'un nombre limité de sons et considère tout le reste comme du bruit. Il est indéniablement difficile d'établir une liste de tous les sons de l'environnement, mais VanDerveer [VanDerveer, 1979] propose une liste de quatre points permettant d'identifier un son de l'environnement :

1. Il est produit par des événements réels.
2. Il a un sens en vertu d'événements causals.
3. Il est plus compliqué que les sons purs générés en laboratoire.
4. Il ne fait pas partie d'un système de communication telle que la parole.

Cette définition fait bien la distinction entre la parole et les autres sons. De façon plus générale, elle exclut tout son faisant partie du système de communication humaine. Il y a certainement plus que la parole dans la communication humaine « sonore ». Un raclement de gorge, un sifflement ou encore un rire sont autant d'exemples de sons qui peuvent, dans certains cas, servir de moyen de communication. Dans ce travail nous considérons tous ces types de sons, y compris la parole,

comme des sons de l'environnement. Même si notre objectif n'est pas de transcrire la parole mais d'en détecter la présence.

Pour une meilleure lisibilité et compréhension du domaine étudié, il est également courant de définir les sons de l'environnement sous forme de taxonomie, en plaçant les sons dans des groupes ou des sous-groupes . David Gerhard [Gerhard, 2003] propose une taxonomie pour les sons de l'environnement regroupés de façon à se rapprocher d'une perspective humaine de la perception du son (figure 2.1). La distinction est d'abord faite entre les sons audibles et ceux non audibles. Les sons audibles sont ensuite subdivisés en cinq catégories : bruit, son naturel, son artificiel, parole et musique. Selon l'auteur, il est difficile de donner une définition objective au bruit. Un genre musical apprécié par une personne peut être perçu comme un bruit par une autre personne. Un son naturel est tout son produit sans aucune influence humaine. Les sons artificiels sont caractérisés par leur source et leur « intention ». Un son artificiel peut ainsi être produit dans l'intention de transmettre un message (sonnerie de téléphone, sirène d'une ambulance, etc.) ; ce n'est pas le cas d'un marteau-piqueur par exemple. Enfin, la parole, qu'elle soit naturelle ou synthétisée, ainsi que la musique sont deux types de sons avec un nombre très important d'éléments de classification pour les humains. Elles sont placées dans deux catégories distinctes.

Cette taxonomie, certes intéressante, ne nous est pas d'une grande aide dans notre projet. Premièrement, elle inclut un nombre très considérable de sons naturels et artificiels qui, à l'exception d'une éventuelle provenance de la télévision ou de la radio, ne risquent pas de se produire régulièrement dans la maison de la personne âgée (chutes d'arbres dans les forêts tropicales ou bruit d'une tronçonneuse, par exemple). Deuxièmement, en vue de limiter notre ensemble de **sons d'intérêt**, beaucoup de sons seront considérés comme du bruit. Exemples de ces sons sont : bruit de la pluie qui bat contre les vitres de la maison, martèlement chez les voisins, insectes, etc. Troisièmement, elle n'aborde pas les caractéristiques physiques de bas niveau du signal, aspect important pour distinguer certains groupes de sons.

Dans cet esprit, nous considérons la taxonomie suivante (figure 2.2) pour le projet SWEET-HOME. Même pour une personne vivant seule, il est très difficile de cerner tous les sons, et encore plus d'identifier les sons intéressants. Cette taxonomie est largement influencée par les objectifs du projet, mais surtout par notre expérience d'indexation du corpus du projet enregistré dans une maison intelligente (chapitre 4).

En premier lieu, une distinction est faite entre les sons humains et les autres sons. Les sons humains seraient plus pertinents pour reconnaître des situations de détresse, mais cela n'est que partiellement

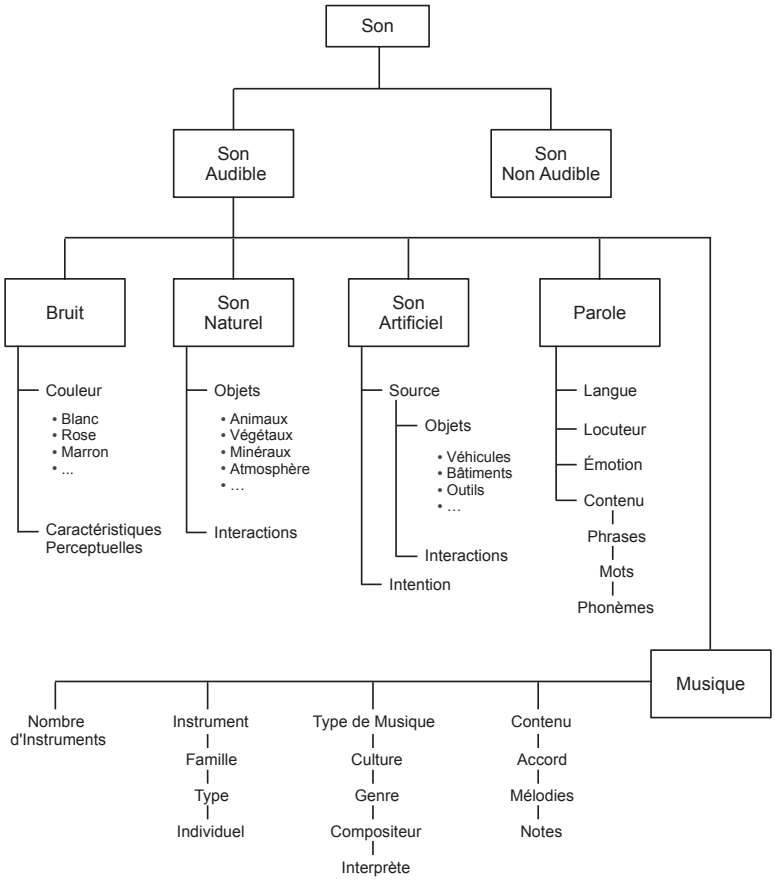

Figure 2.1: Une taxonomie pour les sons perçus par les humains

correct. D'autre sources d'information peuvent en effet améliorer la fiabilité du système. Certains sons non humains peuvent également renseigner le système quant à la présence d'une situation anormale. Des endroits tels que la salle d'eau ou la cuisine sont en général des lieux plus propices pour des accidents de la vie quotidienne. La reconnaissance de certains sons non humains tels que les chocs, les bris d'objets, les moteurs ou encore l'écoulement d'eau pourrait être très précieuse pour le système en vue de détecter une situation anormale. Nous nous intéressons par la suite à deux caractéristiques du signal : la stationnarité et la périodicité. Pour chaque type de sons, humains ou non humains, les sons sont regroupés en fonction de ces deux critères. D'autres détails concernant les bases de sons utilisées et les classes considérées dans nos expérimentations seront présentés au chapitre 4.

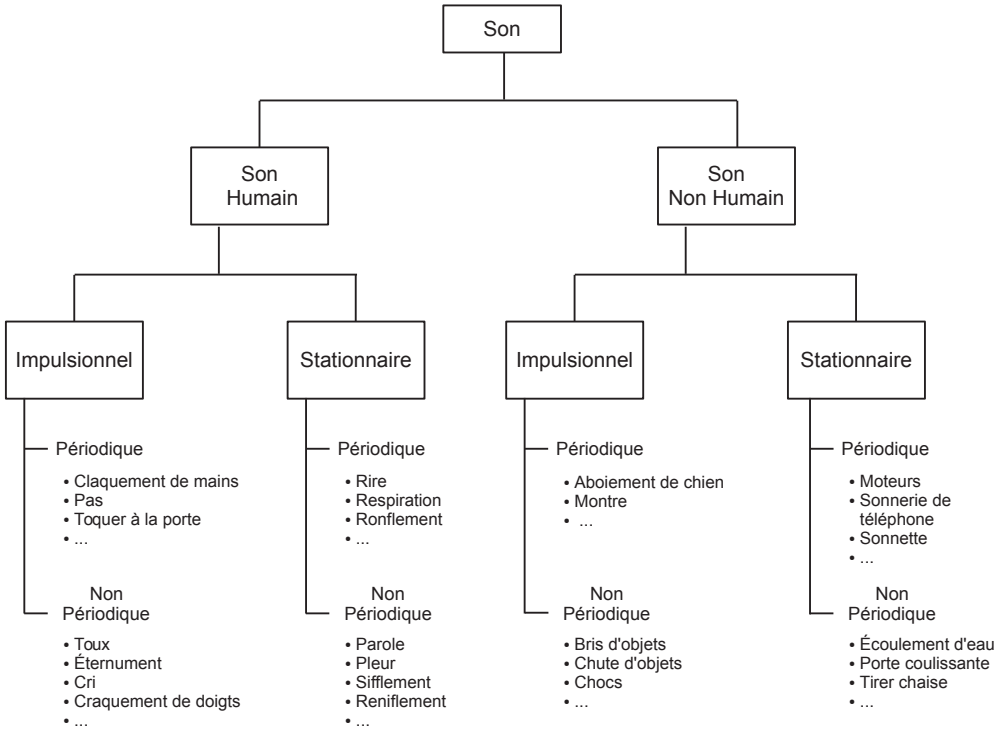

Figure 2.2: Une taxonomie pour les sons dans le projet SWEET-HOME

2.3 Analyse de la scène auditive et reconnaissance du son

Dans un système de reconnaissance automatique de la parole, il est en principe supposé que le signal en entrée correspond à un ou plusieurs mots prononcés par une seule personne. Il en va de même pour la reconnaissance du locuteur. Pour un système de reconnaissance des sons de l'environnement, on part également de l'hypothèse établissant que le signal à reconnaître ne correspond qu'à un seul événement acoustique. Ces hypothèses restent vraies pour peu qu'il n'y ait qu'une seule source sonore à la fois, que la distance de la source par rapport au capteur soit suffisamment courte et que la qualité du signal soit supérieure à un certain seuil. La reconnaissance de la parole, la reconnaissance des genres ou des instruments musicaux ou encore la reconnaissance des sons de l'environnement sont quelques applications, parmi tant d'autres, de la reconnaissance ou de la classification du son.

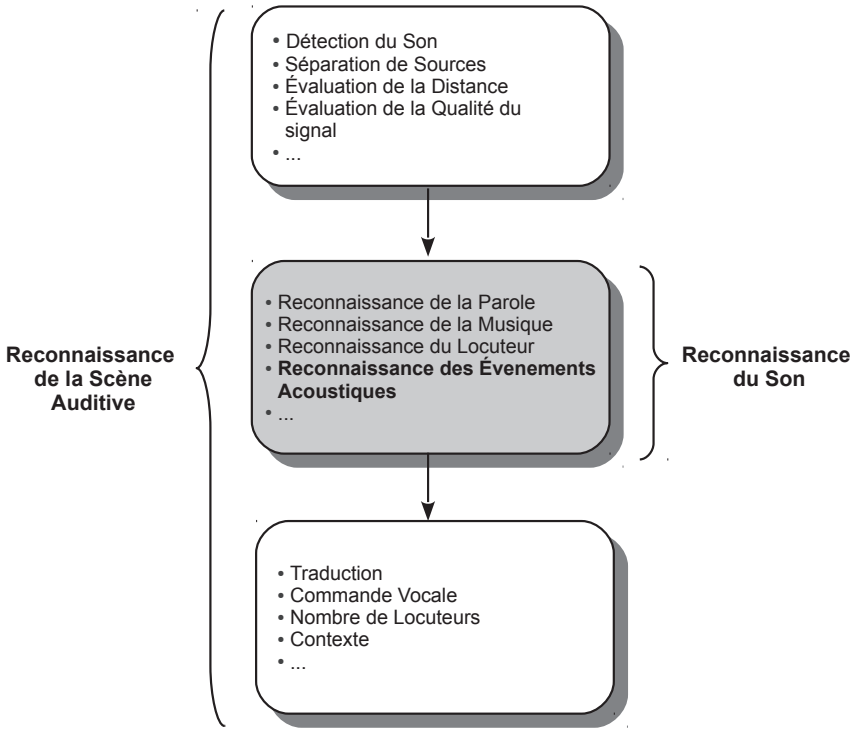

Figure 2.3: Architecture du système SWEET-HOME

2.3.1 Analyse de la scène auditive

Pour un système d'**analyse de la scène auditive** (ASA ou *Auditory Scence Analysis*), les types de systèmes qu'on vient de citer, ne constituent qu'une étape dans une chaîne de traitement plus complexe [Gerhard, 2003]. D'une point de vue « auditif », les sons qui se produisent dans un environnement particulier forment un **paysage sonore** (*Soundscape*) [Schafer, 1969a] [Schafer, 1969b] propre à cet environnement ; comme le font les images dans la notion traditionnelle du terme (paysage ou *Landscape*). La figure 2.3 montre la place de la reconnaissance du son dans système ASA.

Le problème classique en ASA est celui du *Cocktail Party* [Bregman, 1994] [Brown and Cooke, 1994] [Bronkhorst, 2000] [McDermott, 2009]. Dans un environnement où plusieurs sources sonores sont superposées, un être humain peut rester concentré sur une conversation tout en étant capable

d'identifier les quelques sons intéressants qui se produisent autour de lui (son nom par exemple) et d'ignorer tous le reste. D'après Bregamn [Bregman, 1994], différents processus ont lieu dans le système nerveux humain et permettent de ne sélectionner, parmi plusieurs sources en concurrence (on parle également de **mélange de sons** ou *Mixture of Sounds*), que les sons d'intérêt, de les reconnaître et de les utiliser pour construire une « image » de l'environnement, ou une **scène auditive**.

Le terme analyse de scène fut initialement utilisé en vision par ordinateur pour désigner la façon dont un ordinateur analyse et reconnaît des objets dans une photographie. Dans une photographie d'une complexité normale, la partie visible de chaque objet dans la scène se trouve souvent partiellement éclipsée par un autre objet. L'analyse de scène est la stratégie utilisée par l'ordinateur pour reconnaître un objet en se basant sur ses caractéristiques tels que le contour, la texture, la couleur et la distance. De façon similaire, l'analyse de la scène auditive désigne le processus qui consiste à utiliser les caractéristiques des événements acoustiques d'un environnement donné, qui ont lieu dans un certain laps de temps, en vue de les reconnaître [Bregman, 1994].

2.3.2 Reconnaissance du son

Selon la définition de l'ASA, la reconnaissance du son dont il est question dans cette thèse, ainsi que les autres domaines de reconnaissance du son, ne s'apparentent que partiellement à la manière dont les humains perçoivent l'information sonore dans des conditions réalistes. Faute de concevoir des systèmes d'un très haut niveau de complexité, qui aient une perception fine et complète de l'environnement sonore, on s'attaque, depuis de nombreuses années, aux différents sous-domaines de la reconnaissance du son séparément. D'après la vision de Lyon [Lyon, 2010], de tels systèmes « primitifs » devraient être déployés dans les maisons intelligentes, dans les voitures, dans les salles de réunions, etc. Ils pourraient, par la suite, être progressivement enrichis pour déboucher sur des systèmes plus complets.

C'est d'ailleurs un fait qu'on observe sur le terrain. En effet, loin d'être des systèmes ASA, maintes applications utilisant ces systèmes « primitifs » trouvent déjà leur utilité dans plusieurs domaines. Un système de dictée automatique par exemple n'aura souvent pas besoin de traiter le problème de la multiplicité des sources et celui du bruit environnant. Il pourrait, par exemple, utiliser un système sommaire pour la détection du son (basé sur le *ZCR* [1] [Scheirer and Slaney, 1997] ou l'énergie du signal par exemple) afin de ne retenir que le signal utile et se passer du reste, pourvu que toute

1. *Zero Crossing Rate* ou nombre de passages par zéro du signal

activité acoustique détectée corresponde à de la parole. Une approche plus élaborée pourrait être d'utiliser une étape de classification préliminaire pour distinguer entre la parole et les autres sons qui pourraient se présenter lors de la dictée (sonnerie de téléphone, toux, etc.).

De même, un système où un robot se déplace dans une pièce et qui, en tapotant sur le sol avec un marteau en caoutchouc, pourra reconnaître les pièces de carrelage cassées ou mal fixées, n'aura pas besoin de comprendre tout ce qui se passe dans l'environnement en terme d'activité acoustique. Enfin, les applications de recherche de contenus musicaux, qu'on trouve souvent installées sur les smartphones, se désintéressent des sons de l'environnement dans lequel le son est acquis.

La figure 2.4 illustre l'architecture d'un système de reconnaissance du son typique. Les données représentant les classes de sons que l'on désirera reconnaître sont étiquetées par un ou plusieurs annotateurs humains. Depuis le signal audio de chaque classe, on extrait les caractéristiques pertinentes du signal (voir section 2.4 pour les coefficients acoustiques) utilisables en entrée d'un algorithme d'apprentissage. À l'issue de phase d'apprentissage, on obtiendra des modèles représentant les différentes classes de sons. Pour identifier la classe d'un son inconnu (phase de reconnaissance), les coefficients acoustiques extraits du signal sont comparés aux modèles préalablement créés. Le meilleur modèle désignera la classe du son inconnu.

2.3.3 ASA ou reconnaissance du son pour SWEET-HOME ?

Dans un système de reconnaissance du son dans un contexte domotique, comme celui développé dans le projet SWEET-HOME, il paraît, de prime abord, qu'un système ASA soit nécessaire pour analyser le paysage sonore de la maison intelligente. Plusieurs arguments peuvent aisément se présenter en faveur d'un système ASA. Les sons de la vie quotidienne sont tellement variés et les possibilités de superposition des événements ne peuvent être exclues. Une personne pourrait tenir une conversation téléphonique ou lancer une commande vocale en présence d'une télévision ou d'une radio en marche. Elle pourrait également, depuis la salle d'eau, demander de l'aide pendant que l'eau coule du robinet. Enfin, elle pourrait avoir un accident en faisant le ménage et lancer un message de détresse pendant que l'aspirateur fonctionne à pleine puissance.

Lors de la définition des besoins du projet SWEET-HOME, plusieurs hypothèses ont cependant été posées. D'abord, les personnes âgées auxquelles le système est destiné vivent seules. Le nombre d'événements acoustiques simultanés et, plus généralement, le nombre d'activités quotidiennes, reste inférieur à celui qu'on trouve dans un foyer pour une petite famille ou dans une salle de

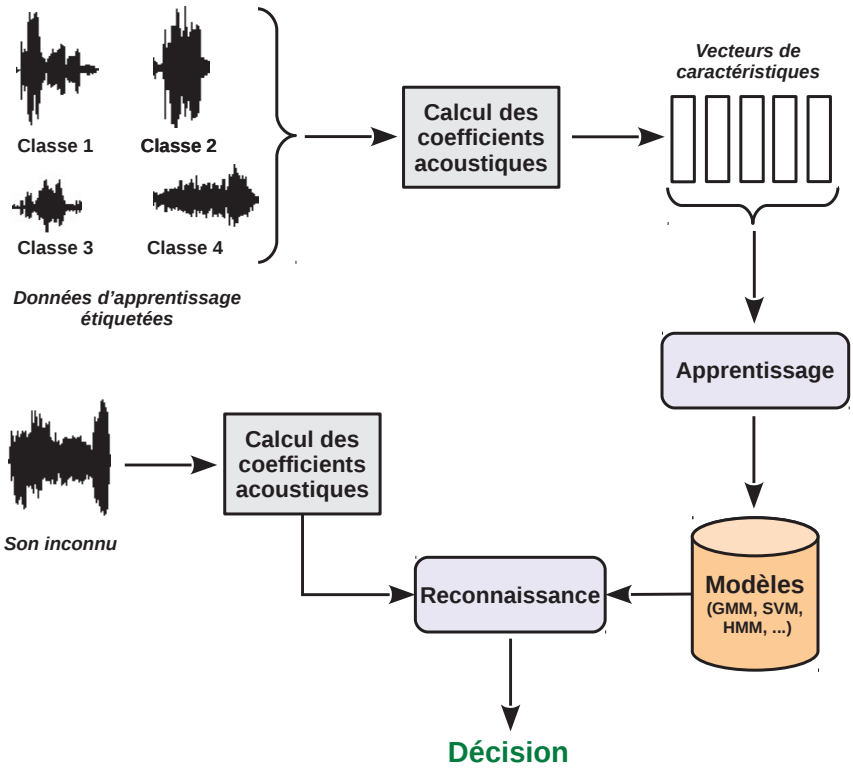

Figure 2.4: Architecture de base d'un système de reconnaissance du son

réunion. Le problème du *Cocktail Party* pourrait à n'importe quel moment se poser mais cela se ferait beaucoup plus sporadiquement que dans les exemples mentionnés. D'autre part, la maison est équipée de plusieurs microphones, ce qui, en se basant sur la qualité du signal acquis par chaque microphone, permettrait de bien se rendre compte que des événements acoustiques simultanés ont eu lieu à deux ou plusieurs endroits différents. Cependant, le même événement pourrait bien être acquis par plusieurs microphones. Pour les expérimentations faites sur le corpus du projet, enregistré en présence de plusieurs microphones, l'approche retenue face à ce type de scénarios était de classifier chacun des signaux acquis par les différents canaux (et qui se trouvent en chevauchement) et de les trier en terme de **rapport signal sur bruit** (RSB). Plusieurs stratégies peuvent alors être utilisées pour décider de la nature de l'événement acoustique. Dans nos expérimentations, nous avons retenu le résultat de reconnaissance du meilleur canal en terme de RSB (chapitre 4).

L'algorithme de classification fonctionne en aval d'un algorithme de détection des événements acoustiques. Pour éviter que celui-ci ne bloque la chaîne de traitement pour de très longues durées, « accaparé » par des événements stationnaires de longue durée (moteurs, écoulement d'eau, etc.),

il est important de lui fixer des limites temporelles. Il doit, en principe, être capable d'alimenter l'algorithme de reconnaissance régulièrement. L'information quant à la présence d'un événement acoustique d'une très longue durée, qui pourrait éventuellement être révélatrice d'une situation anormale, pourra être inférée par plusieurs reconnaissances successives du même (long) événement. Un bon algorithme de détection se doit également d'être capable de continuer à détecter des sons impulsionnels en présence d'un long événement stationnaire.

L'architecture du système SWEET-HOME se présente sous forme de plusieurs couches séparées (voir 4.12, chapitre 4). Le système se compose de trois éléments essentiels : la détection des sons en provenance de plusieurs microphones, la classification des sons de l'environnement (différenciation entre parole et autres sons, puis classification des sons autres que la parole) et la reconnaissance de la parole. À part la détection du son et l'évaluation de la qualité du signal, rien n'est fait en ce qui concerne la distinction des événements sonores se produisant en même temps. En dépit de toutes les hypothèses qui « simplifieraient » le système du projet, il paraît que les futurs systèmes, ceux qui seront déployés chez les personnes âgées, ne devraient pas faire abstraction complète des couches supérieures. En revanche, la conception observée ici, et qui consiste donc à aborder individuellement certaines des composantes d'un système plus complet, semble appropriée pour réaliser les premières expérimentations et évaluer les différentes composantes du système. Par ailleurs, cela corrobore la vision de Lyon (section 2.3.2) qui consiste à concevoir des systèmes d'audition par ordinateur en intégrant graduellement les différents modules.

Par ailleurs, et en dépit du facteur « ergonomie » mis en avant par les solutions basées principalement sur le son, dont fait partie ce projet, même une reconnaissance des événements acoustiques d'un très haut niveau de fiabilité aura ses limites face à certaines situations. Une personne pourrait pousser un cri suite à une chute ou à une blessure, ou parce qu'elle vient d'apercevoir une souris traverser la pièce en diagonale. D'autres modalités peuvent facilement se montrer intéressantes (capteurs de présence, caméra infra-rouge, etc).

2.4 Coefficients acoustiques

Dans cette section, nous décrivons les coefficients acoustiques que nous avons utilisés dans ce travail.

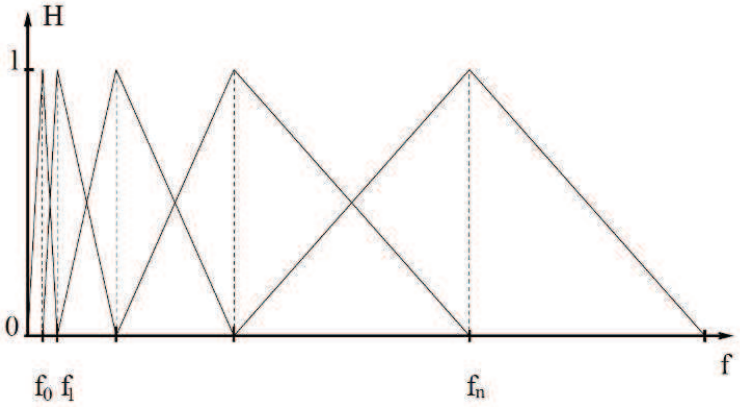

Figure 2.5: Filtres en fréquences de Mel

***MFCC* :** *MFCC* (*Mel-Frequency Cepstral Coefficients*) ou coefficients cepstraux sur l'échelle de Mel (figure 2.5) sont des coefficients cepstraux très utilisés en RAP et en RAL. Le calcul des filtres Mel est est basé sur la perception humaine de la parole [Davis and Mermelstein, 1980] [Zheng et al., 2001].

Pour calculer les coefficients *MFCC*, le spectre du signal est filtré en utilisant des filtres triangulaires qui correspondent à des bandes passantes de même largeur dans le domaine de fréquences de Mel. Les coefficients sont obtenus en appliquant une transformée de Fourier inverse puis une **transformée en cosinus discrète** (DCT ou *Discrete Cosine Transform*) sur les coefficients en sortie des filtres (figure 2.6).

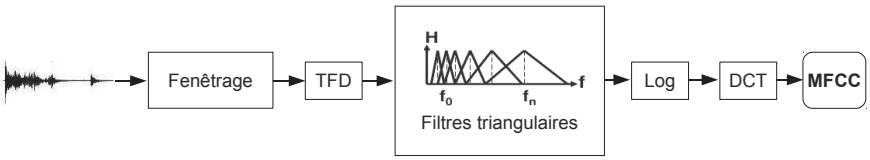

Figure 2.6: Calcul des coefficients MFCC avec DCT

***Loudness* :** Les coefficients de *Loudness* [Moore et al., 1997] [Peeters, 2004] représentent l'énergie du signal dans chaque bande de fréquences dans l'échelle de Bark [Zwicker, 1961] [Smith III and Abel, 1999], divisée sur la somme totale.

***Spectral Flatness* :** Le *Spectral Flatness* [Johnston, 1988] [Dubnov, 2004] est le rapport entre la moyenne géométrique et la moyenne arithmétique des amplitudes de plusieurs bandes de fréquence.

$$SpectralFlatness = \frac{\exp(\frac{1}{K}\sum_{k=1}^{K}\log(a_k))}{\frac{1}{K}\sum_{k=1}^{K}a_k} \tag{2.1}$$

où a_k est l'amplitude de la bande k et K est le nombre de bandes. Dans [Peeters, 2004], les bandes de fréquences suivantes sont utilisées :

— de 200Hz à 500Hz
— de 500Hz à 1000Hz
— de 1000Hz à 2000Hz
— de 2000Hz à 4000Hz

***Spectral Flatness Per Band* :** Représente le *Spectral Flatness* calculé sur des bandes de fréquences logarithmiquement espacées, de 1/4 octave (standard MPEG-7).

***Spectral Crest Factor Per Band* :** Le coefficient *Spectral Crest* est lié au *Spectral Flatness* [Peeters, 2004]. Il représente le rapport entre l'amplitude maximale et la moyenne arithmétique des amplitudes.

$$SpectralCrest = \frac{\max_k(a_k)}{\frac{1}{K}\sum_{k=1}^{K}a_k} \tag{2.2}$$

Le *Spectral Crest Factor Per Band* est le *Spectral Crest* calculé sur des bandes de fréquences logarithmiquement espacées, de 1/4 octave.

***Complex Domain Onset Detection* :** Détection du *Onset* (début d'une note musicale) en utilisant la méthode décrite par [Duxbury et al., 2003].

***Perceptual Sharpness* :** Le coefficient *Perceptual Sharpness* est basé sur les coefficients de *Loudness* [Peeters, 2004]. Il est calculé comme suit :

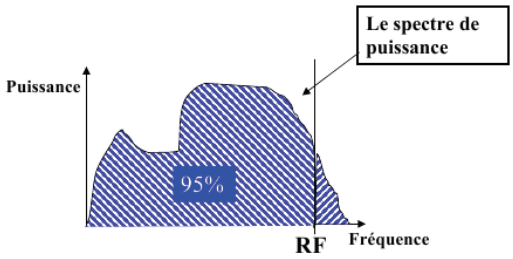

Figure 2.7: *Roll-off* sur le spectre de puissance du signal

$$PerceptualSharpness = 0.11 \frac{\sum_{k=1}^{K_{Bark}} g(k)L(k)}{T} \tag{2.3}$$

où K_{Bark} est le nombre de bandes de Bark, $L'(k) = L(k)^{0.23}$ ($L(k)$ est le k-ième coefficient de *Loudness*) et $T = \sum_{k=1}^{K} L(k)$. La fonction $g(k)$ est définie par :

$$g(k) = \begin{cases} 1 & \text{Si } k < 15 \\ 0.066 \exp(0.171k) & \text{Si } k \geq 15 \end{cases}$$

Perceptual Spread **:** Le calcul du *Perceptual Spread* est également basé sur les coefficients de *Loudness* [Peeters, 2004] :

$$PerceptualSpread = \left(\frac{T - \max_k(L'(k))}{T} \right)^2 \tag{2.4}$$

Spectral Roll-Off **:** Le *Spectral Roll-Off* [Scheirer and Slaney, 1997] est la fréquence au-dessous de laquelle se trouve 95% de l'énergie du signal (figure 2.7). Certaines implémentations peuvent utiliser une valeur plus grande, 99% par exemple.

Spectral Decrease **:** Le *Spectral Decrease* représente le taux de diminution de l'amplitude spectrale [Peeters, 2004]. Il est calculé comme suit :

$$SpectralDecrease = \frac{1}{\sum_{k=2}^{K} a_k} \sum_{k=2}^{K} \frac{a_k - a_1}{k-1} \tag{2.5}$$

Spectral Variation **:** Ce coefficient représente la variation du spectre entre deux fenêtres successives [Peeters, 2004]. Il est calculé comme suit :

$$SpectralVariation = 1 - \frac{\sum_{k=1}^{K} a_k^{t-1} a_k^t}{\sqrt{\sum_{k=1}^{K} (a_k^{t-1})^2} \sqrt{\sum_{k=1}^{K} (a_k^t)^2}} \tag{2.6}$$

Spectral Slope **:** Ce coefficient est calculé par régression linéaire de l'amplitude spectrale :

$$SpectralSlope = \frac{1}{\sum_k a_k} \frac{K \sum_k f_k a_k - \sum_k f_k \sum_k a_k}{K \sum_k f_k^2 - (\sum_k a_k)^2} \tag{2.7}$$

D'autres coefficients sont également utilisés dans nos expérimentations. Les coefficients *Temporal Shape Statistics*, *Spectral Shape Statistics*, *Envelope Shape Statistics* sont des coefficients statistiques de haut niveau (*Centroid*, *Spread*, *Skewness* et *Kurtosis*) calculés pour les échantillons du signal, l'amplitude spectrale, et l'enveloppe d'amplitude du signal, respectivement. Plus de détails sur le calcul des coefficients statistiques seront donnés au chapitre 5.

2.5 Méthodes de classification

Les algorithmes de classifications utilisent les coefficients acoustiques pour décider de la classe d'un événement inconnu. Entre l'extraction des coefficients acoustiques et la classification proprement dite, il est possible que les données subissent des étapes de traitement supplémentaires telle qu'une normalisation ou une transformation. Ces étapes sont souvent propres à une méthode d'extraction ou celle d'une classification et ne peuvent pas être considérées comme élément indépendant du système. Nous les évoquons, le cas échéant, lors de la description de l'algorithme qui en fait usage.

Plusieurs méthodes de reconnaissances des formes ont été utilisées en reconnaissance du son. Nous décrirons dans ce qui suit deux méthodes très utilisées : les **modèles de mélange Gaussiens**

(GMMs ou *Gaussian Mixture Models*) et les **machines à vecteurs de support** (SVMs ou *Support Vector Machines*). Elles seront retenues et utilisées tout au long de nos expérimentations. D'autres méthodes, pas moins intéressantes, tels que les **modèles de Markov cachés** (HMMs ou *Hidden Markov Models*) [Rabiner, 1989], sont également utilisées dans la littérature. Nous les présenterons brièvement, en cas de besoin, au fur et à mesure que nous les rencontrerons.

2.5.1 Modèles de Mélange Gaussiens

Un modèle de mélange Gaussien est un modèle statistique exprimé sous forme d'une somme pondérée de K composantes gaussiennes [Everitt and Hand, 1981] [Titterington et al., 1985] [Reynolds, 2009] :

$$p(\mathbf{x}|\lambda) = \sum_{i=1}^{K} w_i \mathcal{N}(\mathbf{x}|\mu_i, \mathbf{\Sigma}_i) \tag{2.8}$$

où $\mathbf{x}$ est un vecteur de coefficients de dimension D, $w_k, k = 1, \ldots, K$ sont des poids et $\mathcal{N}(\mathbf{x}|\mu_k, \mathbf{\Sigma}_k), k = 1, \ldots, K$ sont des des composantes de densités gaussiennes. Chaque composante est de la forme :

$$\mathcal{N}(\mathbf{x}|\mu_k, \mathbf{\Sigma}_i) = \frac{1}{(2\pi)^{D/2}|\mathbf{\Sigma}_k|^{1/2}} \exp\left(-\frac{1}{2}(\mathbf{x} - \mu_k)^T \mathbf{\Sigma}_k^{-1}(\mathbf{x} - \mu_k)\right) \tag{2.9}$$

où μ_k est la moyenne et $\mathbf{\Sigma}_k$ est la matrice de covariance.

2.5.2 Machines à Vecteurs de Support

Contrairement aux GMMs, qui représentent un modèle statistique, les SVMs [Vapnik, 2010] [Burges, 1998] [Schölkopf and Smola, 2002] sont une méthode dite discriminative. L'objectif de l'algorithme d'apprentissage est de trouver un hyperplan permettant de séparer des points appartenant à deux classes, de sorte que les erreurs de classification soient les plus faibles possibles. Comme le montre la figure 2.8, le nombre d'hyperplans pouvant séparer les deux classes peut être très élevé. L'idée de base cette méthode est de trouver l'hyperplan qui maximise la marge entre les

points des deux classes (hyperplan A dans la figure 2.8).

La figure 2.9 illustre ce principe. L'hyperplan H_0 est défini par :

$$w \cdot \mathbf{x} - b = 0 \tag{2.10}$$

où w est la norme de l'hyperplan et b est le décalage par rapport à l'origine. Étant donné un ensemble de N points $\mathbf{x}_i \in R^p$ $(1 \geq i \geq N)$, ayant comme étiquettes $y_i \in \{-1, 1\}$ respectivement, l'objectif est de trouver l'hyperplan qui maximise la marge entre les points des deux classes tout en respectant les contraintes suivantes :

$$w \cdot \mathbf{x}_i - b \geq 1$$

pour $y_i = 1$, et

$$w \cdot \mathbf{x}_i - b \leq -1$$

pour $y_i = -1$, ce qui peut être écrit par :

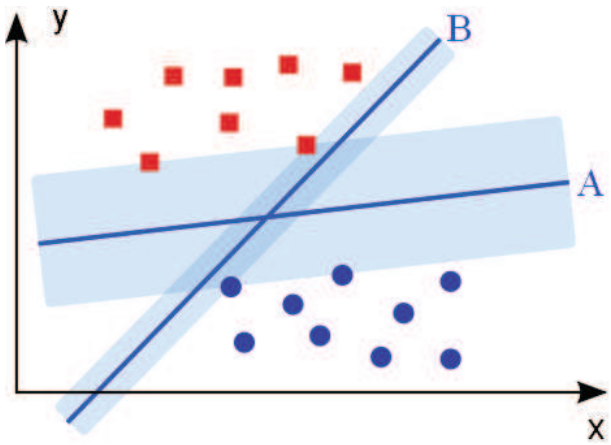

Figure 2.8: Plusieurs hyperplans peuvent séparer les deux classes, le meilleur étant celui qui maximise la marge entre les deux classes.

$$y_i(\mathbf{x}_i \cdot w + b) - 1 \geq 0, \forall i \tag{2.11}$$

La distance entre H_1 et H_2 est $\frac{2}{||w||}$. De ce fait, le problème peut être exprimé comme suit : minimiser $||w||$ sous les contraintes de l'équation (2.11).

Le problème peut être formulé comme un problème de programmation quadratique :

$$L_p = \frac{1}{2}||w||^2 - \sum_{i=1}^{N} \alpha_i y_i(\mathbf{x}_i \cdot w + b) + \sum_{i=1}^{N} \alpha_i \tag{2.12}$$

où les α_i sont les multiplicateurs de Lagrange.

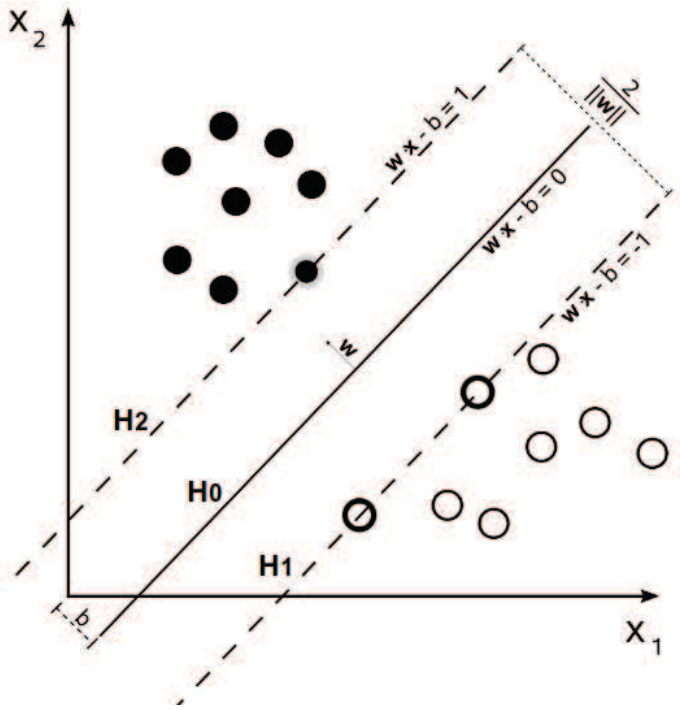

Figure 2.9: Exemple d'un classifieur linéaire.

Dans la figure 2.9, on peut voir que certains points se trouvent disposés sur les frontières H_1 et H_2. Ce sont les vecteurs de support et leurs multiplicateurs α_i sont supérieurs à zéro.

Dans beaucoup de problèmes, il n'est pas toujours possible de trouver un hyperplan permettant

une séparation parfaite des deux classes. En d'autres termes, il n'existe pas d'hyperplan qui puisse séparer tous les points sans faire la moindre erreur de classification. Ce problème est résolu par l'introduction d'une technique dite marge souple. Elle consiste à utiliser des variables ressort (*Slack Variables*) $\xi_i \geq 0$ afin de tolérer les erreurs de classification pour certains points tout en continuant à maximiser la marge. De ce fait, le problème devient :

Minimiser :

$$\frac{1}{2}||w||^2 + C\sum_{i=1}^{N}\xi_i$$

avec les contraintes :

$$y_i(\mathbf{x}_i \cdot w + b) \geq 1 - \xi_i, \forall i \tag{2.13}$$

où $C > 0$ est une constante utilisée pour contrôler le compromis entre les erreurs de classification et la largeur de la marge.

Pour beaucoup de problèmes réels, aucune séparation linéaire des données n'est possible (à moins d'avoir un nombre très élevé d'erreurs de classification). Une solution à ce problème consiste à projeter les données dans un espace d'une très grande dimension, voire d'un nombre infini de dimensions, dans lequel une séparation linéaire des données serait possible. Cependant, traiter des données dans un tel espace peut facilement s'avérer impossible en pratique. Une autre solution consiste à utiliser un autre type de noyau que le noyau linéaire. Les fonctions noyau les plus utilisées sont :

$$\text{Linéaire :} \qquad K(\mathbf{x}, \mathbf{y}) = \mathbf{x} \cdot \mathbf{y} \tag{2.14}$$

$$\text{Polynomiale :} \qquad K(\mathbf{x}, \mathbf{y}) = (\gamma \mathbf{x} \cdot \mathbf{y} + c)^p \tag{2.15}$$

$$\text{RBF (}\textit{Radial Basis Function}\text{)} : \qquad K(\mathbf{x}, \mathbf{y}) = \exp(-\Gamma|\mathbf{x} - \mathbf{y}|^2) \tag{2.16}$$

La fonction de décision finale prend la forme :

$$f(\mathbf{x}) = \sum_{i=1}^{N_{sv}} \alpha_i y_i K(\mathbf{x}, \mathbf{x}_i) + b \tag{2.17}$$

où N_{sv} est le nombre de vecteurs de support. Le signe de la fonction f désigne la classe du vecteur d'entrée $\mathbf{x}$.

2.6 Sélection de caractéristiques

Dans beaucoup de problèmes, le nombre de caractéristiques des vecteurs en entrée est très élevé. Les méthodes de sélection de caractéristiques permettent de réduire le nombre de dimensions des données sans affecter les performances de classification. Dans certains cas, on peut même améliorer les performances en supprimant les caractéristiques qui peuvent être source de confusion entre les classes.

La méthode proposée par [Guyon et al., 2002] consiste à éliminer, via une procédure itérative, les caractéristiques les moins discriminantes d'un modèle SVM. L'algorithme commence par créer un modèle SVM en utilisant toutes les caractéristiques. Par la suite, on calcule un certain score c_k, pour chaque caractéristique k. Pour un noyau linéaire, le score c_k est calculé par :

$$c_k = \left(\sum_{i=1}^{N_{sv}} \alpha_i y_i \mathbf{x}_i[k] \right)^2 \tag{2.18}$$

où $\mathbf{x}_i[k]$ est la k-ième caractéristique du vecteur de support $\mathbf{x}_i$ du modèle SVM.

Un nouveau modèle SVM est créé, en éliminant la caractéristique au plus faible score, et évalué en utilisant des données de développement. L'algorithme s'arrête lorsqu'une baisse en performances est constatée, sinon une nouvelle itération est exécutée pour éliminer une autre caractéristique. Ce processus est particulièrement long, bien que les auteurs mentionnent la possibilité d'éliminer

plusieurs caractéristiques à chaque itération.

En pratique, et pour les problèmes dont le nombre de dimensions est particulièrement élevé, un nombre réduit de caractéristiques s'avèrera souvent utile pour la classification. Dans ce travail (chapitre 5), nous utiliserons une version simplifiée de l'algorithme : le pourcentage de caractéristiques à conserver est fixé à l'avance ; une seule itération donc est nécessaire. Nous appelons cette variante SVM-Wrapper (la méthode présentée dans [Guyon et al., 2002] est appelée SVM RFE pour *Recursive Feature Elimination* ou élimination récursive des caractéristiques).

Une autre méthode de sélection de caractéristiques est basée sur un coefficient appelé F-Score [Chen and Lin, 2006]. Le F-Score (équation 2.19) permet mesurer la corrélation entre une classe et une des caractéristiques.

$$F\text{-}Score(k) = \frac{\left(\mu_k^+ - \mu_k\right)^2 + \left(\mu_k^- - \mu_k\right)^2}{\sigma_k^+ + \sigma_k^-} \tag{2.19}$$

où μ_k^+, μ_k^- et μ_k sont les moyennes de la caractéristique k pour les exemples positifs, les exemples négatifs et tous les exemples, respectivement. σ_k^+ et σ_k^- sont les écarts-types de la caractéristique k des exemples positifs et ceux négatifs, respectivement. Les caractéristiques au score le plus important sont les plus discriminantes. À l'instar de SVM-Wrapper, nous pouvons, avec cette méthode, choisir le pourcentage de caractéristiques à conserver pour la classification.

Les deux méthodes SVM-Wrapper et F-Score évaluent chaque caractéristique indépendamment des autres. Nous utiliserons également une méthode permettant de mesurer la corrélation entre les caractéristiques. Elle est appelée CFS (*Correlation based Feature Selection*) [Hall, 1999] [Hall, 2000].

2.7 Exploiter les avancées dans les champs voisins : possibilités et limites

Dans cette section, nous examinons les similitudes entre la reconnaissance des événements acoustiques et quelques domaines voisins. Nous expliquons les possibilités d'une utilisation des techniques proposées dans chaque domaine, et ce que nous avons effectivement retenu dans ce travail. Les domaines en question sont : la reconnaissance de la parole, la reconnaissance du locuteur, la reconnaissance des émotions et la reconnaissance de la musique. Avant de discuter chaque domaine séparément, nous commençons par présenter les caractéristiques communes et celles propres à

chaque type de signal.

Si les signaux de la parole et de la musique ont bien été étudiés dans la littérature, peu de travail est réalisé concernant la caractérisation des sons de l'environnement. [Yamakawa et al., 2010] présentent une initiative intéressante visant à comparer les signaux relatifs aux domaines cités. Le tableau 2.1 montre le résultat de cette analyse. Certaines caractéristiques bien connues de la parole et de la musique restent difficiles à déterminer pour les sons de l'environnement.

On pourra, par ailleurs, imaginer d'autres caractéristiques en complément de ce tableau. Le nombre de sources susceptibles de produire un son, par exemple, constitue également un facteur qui partage la parole et la musique d'une part et les sons de l'environnement d'autre part, à deux niveaux de complexité bien éloignés. La parole est, en principe, produite par l'appareil vocal humain, la musique par un nombre déterminé d'instruments, mais il reste difficile de définir toutes les provenances des sons de l'environnement. Cela fait penser à un autre facteur, celui de l'ambiguïté. Par ambiguïté on fait référence, ici, à la difficulté que peut éprouver l'oreille humaine à distinguer deux ou plusieurs sons, sans utiliser d'autres sources d'informations (images, texte, contexte, etc.), même si les sons sont produits par des processus très différents. Il est souvent facile de savoir que le son que l'on vient d'entendre est bien de la parole, même si la langue parlée et son système phonétique nous sont complètement étrangers. Il n'est pas très courant de confondre la parole avec un autre son, même si l'effet d'une pédale wah-wah d'une guitare électrique donne parfois l'illusion d'entendre une voix humaine. Le problème devient un peu plus difficile en musique. Certaines notes d'une guitare basse et celles d'une batterie peuvent parfois être difficiles à distinguer. Il en va de même pour la guitare électrique et le clavier. Mais l'ambiguïté devient de loin plus importante dès qu'il s'agit des sons de l'environnement. Plus de détails concernant ce point sont abordés à la sous-section suivante.

Cette difficulté à contourner les caractéristiques des sons de l'environnement de façon pragmatique a contribué à la création de communautés de chercheurs qui s'intéressent chacune à un problème particulier (animaux, respiration, moteurs, etc).

2.7.1 Reconnaissance des Événements Acoustiques *versus* Reconnaissance de la Parole

Il semble aisé de réaliser que, parmi tous les domaines cités, la reconnaissance de la parole est celui qui a toujours eu le plus d'intérêt. De nos jours, des systèmes de reconnaissance de la parole sont commercialisés pour plusieurs langues. Le succès des outils utilisés en reconnaissance de la

Tableau 2.1: Caractéristiques des signaux de la parole, de la musique et des sons de l'environnement

Characteristiques Acoustiques	**Parole**	**Musique**	**Sons de L'environnement**
Nbr. de classes	Nbr. de phonèmes	Nbr. de notes	Non défini
Longueur de fenêtre d'analyse	Courte (fixe)	Longue (fixe)	Non définie
Décalage de fenêtre d'analyse	Court (fixe)	Long (fixe)	Non défini
Largeur de bande	Étroite	Relativement étroite	Large, Étroite
Harmoniques	Claires	Claires	Claires, Non claires
Stationnarité	Stationnaire Non stationnaire	Stationnaire (sauf percussions)	Stationnaire, Non stationnaire
Structure répététive	Faible	Faible	Faible, Forte

parole (coefficients acoustiques et algorithmes de reconnaissance) n'a pas tardé à profiter à d'autres domaines telle que la reconnaissance du locuteur et l'identification de la langue. Ainsi, il n'est pas étonnant de voir des systèmes de reconnaissance et vérification du locuteur utiliser des coefficients *MFCC* et des méthodes tels que les HMMs (du moins les systèmes de reconnaissance dépendant du texte prononcé).

La différence entre la reconnaissance de la parole et celle du son réside principalement dans le fait que la première bénéficie de plusieurs sources d'informations qui, pour la reconnaissance du son, semblent inaccessibles ou bien difficiles à définir. Les sources d'informations qui marquent la différence entre les deux domaines sont le modèle acoustique, le modèle de langage et le contexte.

Ballas et Howard [Ballas and Howard, 1987], soutiennent que la perception humaine des sons de l'environnement peut être assimilée à celle d'une forme de langage. Ils mettent cependant l'accent sur l'absence d'un alphabet phonétique pour les sons de l'environnement. La raison de cela vient du fait que la parole est produite par l'appareil vocal humain en utilisant un nombre limité d'actions, tandis que les sons de l'environnement peuvent provenir d'un nombre beaucoup plus important de sources.

Ils expliquent également comment la connaissance du contexte peut être importante importante pour la reconnaissance, que ce soit de la parole ou celle des sons de l'environnement. Cela conduit à deux types de traitements différents au niveau du système auditif humain, un traitement descendant (*top-down*) et un traitement ascendant (*bottom-up*). À l'instar des mots et des phrases que véhicule le

signal de parole, les sons de l'environnement peuvent avoir une description sémantique, à l'opposé d'une description basée sur les caractéristiques du signal. Un son est par exemple défini par « bris de glace » et non pas par « son rapide avec des éléments aigus qui varient dans le temps ».

Toujours d'après Ballas et Howard, la fiabilité du système auditif humain baisse si un même son peut être produit par diverses sources. Les sons semblables ayant des sources différentes sont appelés **sons homonymes**. Comme pour les homonymes linguistiques, tels que *night* et *knight* en langue Anglaise, qui ne peuvent être distingués sans connaissance du contexte, les sons homonymes ont aussi besoin d'être « entourés » d'autres sons en guise de contexte pour être reconnus correctement par des humains.

Cette théorie est soutenue pas des expérimentations dans lesquelles des sujets humains tentent d'identifier une scène en écoutant une série de plusieurs sons. Il a été constaté que l'interprétation donnée à une scène est directement lié à l'ordre dans lequel les sons se présentent. Par exemple, si un bruit de fracas métallique est précédé par un grincement très fort, l'interprétation de la scène peut être un accident de voiture. Si, en revanche, le même fracas métallique est accompagné d'un son d'écoulement d'eau et de celui d'un jet d'air, l'interprétation est typiquement le bruit d'une machine dans une usine.

En substance, cette théorie du traitement descendant suggère que, comme elle l'est pour le langage parlé, la capacité humaine à distinguer les sons de l'environnent est étroitement liée à la connaissance acquise *à priori* des sons à reconnaître.

Comment peut-on alors se servir des techniques utilisées en reconnaissance de la parole pour reconnaître les sons de l'environnement ? Plus particulièrement, comment peut-on les appliquer au projet SWEET-HOME et aux projets similaires ? Il semble qu'il serait possible de cerner l'*alphabet phonétique* d'un seul habitat accueillant une seule personne, en faisant des enregistrements de toutes ses activités pendant une certaine période de temps. S'ensuivent alors une annotation de tous les événements acoustiques et une création de tous les modèles « personnalisés » pour les classes de sons de l'appartement. On pourrait même créer un modèle par source de sons au lieu d'un modèle pour tous les sons appartenant à la même classe. C'est à dire, on pourrait créer un modèle pour le bruit d'ouverture ou de fermeture de porte pour chaque porte de l'appartement au lieu de créer un modèle pour la classe correspondante englobant toutes les portes. Cette solution semble, toutefois, difficilement utilisable à grand échelle. Une alternative à cela pourrait être d'utiliser des modèles génériques qui pourront probablement être adaptés en utilisant une quantité limitée d'enregistrements réalisés dans l'appartement cible.

D'autre part, comme les mots d'un texte en langage naturel ne peuvent pas être rangés dans n'importe quel ordre, beaucoup d'activités de la vie courante sont composées d'actions élémentaires qui ne se présentent pas dans un ordre arbitraire. Après tout, on ne peut pas boire du thé avant de l'avoir versé dans une tasse, encore moins avant de l'avoir préparé. Une manière d'utiliser, en reconnaissance des sons de l'environnement, ce qui serait l'équivalent du modèle de langage en reconnaissance de la parole, pourrait se faire par la définition de scénarios d'activités qui ne sont qu'une suite d'événements acoustiques dans un ordre particulier. Un scénario *Aller aux toilettes*, par exemple, se composerait des événements suivants :

— Allumer la veilleuse
— Descendre du lit
— Mettre ses pantoufles
— Marcher vers les toilettes
— Allumer la lumière
— Ouvrir la porte
— Faire ses besoins
— Tirer la chasse d'eau
— Sortir en fermant la porte
— Eteindre la lumière

De manière similaire, on pourrait imaginer des scénarios *Faire la vaisselle*, *Rentrer à la maison*, *Préparer une boisson chaude*. Bien évidemment, ce processus reste très difficile à généraliser et requiert, de surcroît, l'utilisation de grosses quantités de données et l'établissement de scénarios par des experts.

De la reconnaissance de la parole, nous retenons, dans le cadre de cette thèse, l'utilisation des coefficients *MFCC* comme caractéristiques acoustiques de base. Ils sont utilisés dans nos premières expérimentations et servent de base de comparaison dans les expérimentations suivantes.

2.7.2 Reconnaissance des Événements Acoustiques *versus* Reconnaissance du Locuteur

À la différence de la parole, les performances d'un système de reconnaissance du locuteur ne dépendent pas d'une bonne délimitation du début et de la fin de la phrase prononcée à moins, bien entendu, qu'il s'agisse d'un système d'identification du locuteur basé sur un texte figé. Cette propriété est aussi valable pour les événements acoustiques, quoique seulement partiellement. En

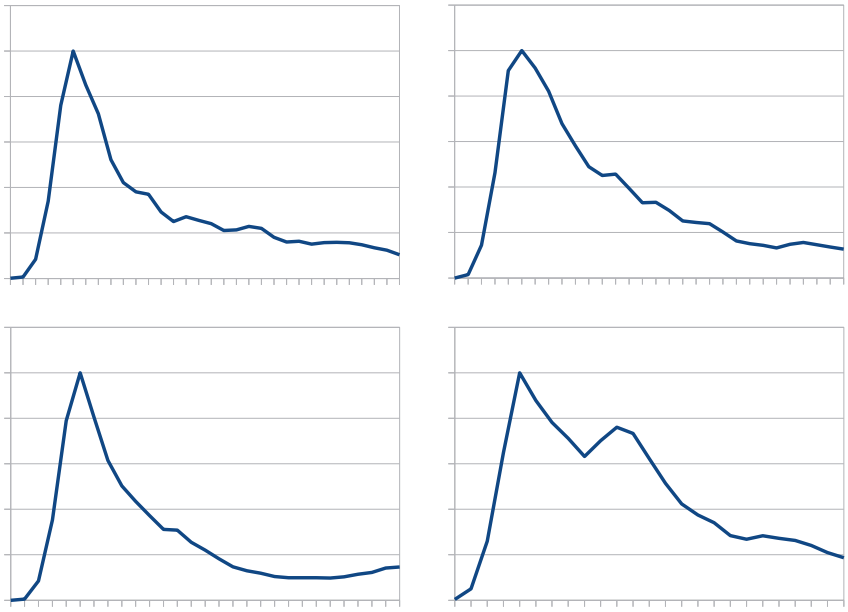

Figure 2.10: Enveloppe d'amplitude pour 4 enregistrements de bris de glace

effet, si le système de reconnaissance des sons de l'environnement est conçu de manière à exploiter l'information temporelle, il devient nécessaire de bien détecter un événement sonore du début jusqu'à la fin. Pour beaucoup de classes de sons, la variation du signal dans le temps peut, en quelque sorte, être une caractéristique du son. Pour d'autres, cependant, la notion de début et fin de l'événement acoustique reste imprécise. Pour illustrer ces propos, observons l'enveloppe d'amplitude pour deux classes de sons différentes. Dans les figures 2.10 et 2.11 on voit l'enveloppe d'amplitude de quatre enregistrements d'un bris de glace et d'un sèche-cheveux respectivement. Pour exploiter l'information temporelle en classification, en utilisant les HMMs par exemple, il est important, pour le bris de glace, de capter tout le signal du début jusqu'à la fin, d'autant plus que ce type de son est très court. À l'opposé de cela, le début et la fin d'événement pour le bruit d'un sèche-cheveux ne sont pas très clairs. Une tranche du signal suffisamment longue pourrait être utilisée pour la reconnaissance.

De toute façon, des travaux utilisant des HMMs pour la classification des sons de l'environnement existent, ils seront présentés au chapitre 3. Les techniques utilisées en reconnaissance du locuteur deviennent très intéressants dès lors qu'on réalise qu'il s'agit, pour les deux problèmes, de comparer des signaux de longueur différente. En reconnaissance du locuteur on a recours à plusieurs

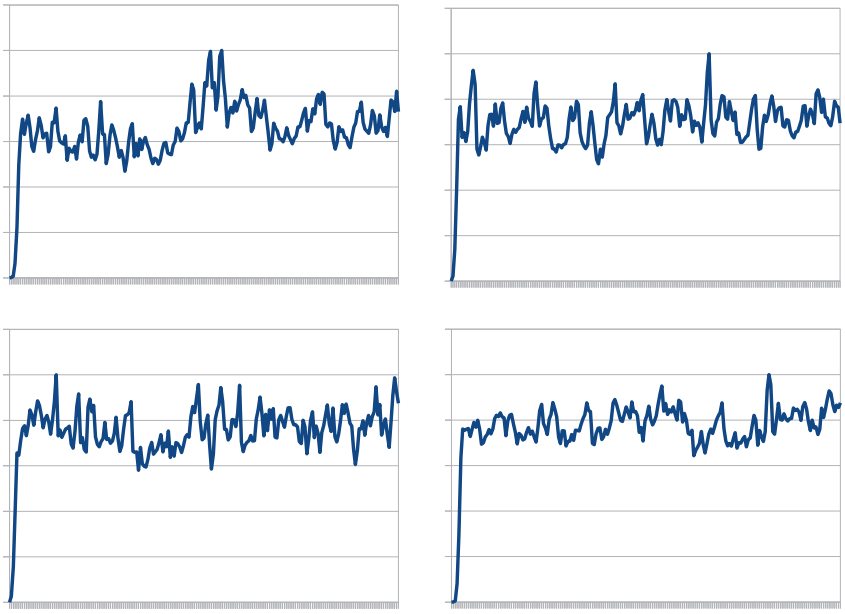

Figure 2.11: Enveloppe d'amplitude pour 4 enregistrements du bruit d'un sèche-cheveux

techniques pour contourner ce problème. Dans les années 90, les systèmes basés sur les GMMs avaient beaucoup de succès en reconnaissance du locuteur. Dans les premières tendances de mettre à profit les avancées en reconnaissance du locuteur, beaucoup de travaux sur la reconnaissance des sons de l'environnement ont utilisé les GMMs, avec plus ou moins de succès.

Depuis de nombreuses années, de nouveaux noyaux SVM linéaires, dits de discrimination de séquences (*Sequence Discriminant Kernels*) ont été proposés. Ils ont rapidement été adoptés et améliorés par la communauté de la RAL. Les performances obtenues ont rapidement permis à ces techniques de prendre le dessus sur les GMMs en tant que méthode standard pour la reconnaissance et la vérification du locuteur. Dans cette thèse, un intérêt particulier est accordé aux noyaux SVM de discrimination de séquences. À côté des GMMs, qui serviront de base de comparaison, ce seront les éléments que nous retiendrons de la reconnaissance du locuteur.

2.7.3 Reconnaissance des Événements Acoustiques *versus* Reconnaissance des Émotions

La reconnaissance acoustique des émotions [Dellaert et al., 1996] [Schuller et al., 2011] [Clavel et al., 2008] possède des liens plus ou moins forts avec la reconnaissance des sons de l'environnement. En effet, plusieurs sons humains dont on s'intéresse dans ce travail sont également considérés comme des sons d'intérêt pour la reconnaissance des émotions. On parle alors de signaux non verbaux (rire, soupir, etc.). Cependant, la reconnaissance des émotions ne se limite souvent pas qu'à ce type de signaux. L'expression du visage, les gestes et les signaux physiologiques peuvent, pour ne nommer que ceux-là, également être utilisés pour reconnaître les émotions d'une personne [Devillers et al., 2005].

Par ailleurs, la reconnaissance des émotions basée sur le signal acoustique peut s'intéresser à plusieurs types d'indices qu'on peut regrouper en indices linguistiques (dialogiques et lexicaux) et autres paralinguistiques (prosodie, disfluences, etc.) [Devillers and Vidrascu, 2006]. De tels indices sont étroitement liés à la voix et peuvent être considérés comme des informations de plus haut niveau.

2.7.4 Reconnaissance des Événements Acoustiques *versus* Reconnaissance de la Musique

En dépit des similitudes entre la reconnaissance du locuteur et celle des sons de l'environnement, il ne faut certainement pas aller jusqu'à considérer que les deux problèmes n'en font finalement qu'un seul. Autrement dit, en utilisant les techniques de la RAL pour la REA, il ne faut pas assimiler une classe de sons à un locuteur, les différents enregistrements d'un son étant comparés aux différentes sessions d'enregistrements d'un locuteur. Cette hypothèse est pourtant faite dès lors que les méthodes de la RAL sont « littéralement » appliquées en REA.

La reconnaissance des sons de l'environnement serait plus comparable à la reconnaissance du locuteur pour des applications qui ne s'intéressent qu'à une classe de sons. Reconnaître plusieurs locuteurs serait comparable au fait de tenter de discriminer les claquements des différentes portes dans un corridor ou de discriminer le bruit d'un moteur défectueux de celui produit par un moteur du même mais ne présentant pas de défauts. Or, dans notre problème, tous les bruits de portes sont regroupés dans la même classe, bien qu'ils émanent de sources différentes. Il est donc important d'identifier ce qui existe en commun entre les différents bruits de porte et ce qui permet de les différencier de tout autre type de sons.

Dans cette thèse, la similitude inter-classe et la différence intra-classe sont puisées au niveau des coefficients acoustiques. L'hypothèse formulée à ce sujet est la suivante :

Une seule famille de coefficients acoustiques ne pourrait pas, pour chacune des classes de sons d'intérêt, mettre en relief toutes les similitudes entre les sons appartenant à une classe donnée et provenant éventuellement de sources différentes, et les divergences entre ceux-ci et les sons appartenant aux autres classes.

L'approche empruntée en vue de tester cette hypothèse est d'utiliser plusieurs types de familles de coefficients et d'observer, pour chaque couple de deux classes, celle(s) qui permet(tent) de mieux les discriminer.

En reconnaissance de la musique, les différences entre instruments, notes et genres musicaux sont exploitées. Dans cet esprit, plusieurs coefficients acoustiques sont proposés dans la littérature [Peeters, 2004]. Ce sera notre élément à retenir de la reconnaissance de la musique.

CHAPITRE 3

TRAVAUX SUR LA RECONNAISSANCE DES SONS DE L'ENVIRONNEMENT

Comme nous l'avons vu au chapitre 2, plusieurs types d'applications s'intéressent à la reconnaissance des sons de l'environnement. Cependant, les différentes applications ne s'intéressent systématiquement qu'à un nombre très limité de sons, voire à un seul type de sons (toux, respirations, chants de baleines, etc.). Par conséquent, on trouve dans la littérature un nombre important de travaux différents qui sont, pour la plupart, très difficilement comparables. Pour pouvoir restituer une partie des approches intéressantes de la littérature, les présenter de façon cohérente, et bien spécifier celles qui ont le plus influencé ce travail, il est important de pouvoir les placer dans différentes catégories.

Pour ce faire, plusieurs critères pourraient être individuellement ou conjointement utilisés, comme : les coefficients acoustiques utilisés, les méthodes de classification (GMMs, SVMs, HMMs, etc.), le nombre de classes de sons auxquelles on s'intéresse (une seule classe, plusieurs classes) ou bien la nature des sons d'intérêt (impulsionnels, stationnaires, les deux, etc.). Plusieurs autres critères pourraient être imaginés.

Cela étant dit, vu le panorama des travaux existants et les travaux antérieurs dont ils s'inspirent, il n'est pas évident de trouver des catégories de méthodes qui soient clairement distinctes l'une de l'autre. D'autant plus que ce domaine est encore à ses premiers pas et qu'aucune méthode ne s'est vraiment imposée. Partant de ce point, c'est-à-dire, de l'arrière-plan qui a donné direction aux méthodes de reconnaissance des sons de l'environnement les plus intéressantes, nous abordons les différents travaux fondés sur :

— Le système auditif humain
— La reconnaissance de la parole
— La reconnaissance du locuteur
— Les techniques de traitement d'image

Notons qu'une bonne partie de notre travail a été influencée par la reconnaissance du locuteur (RAL), bien qu'on ait également utilisé les coefficients *MFCC* qui viennent de la reconnaissance

de la parole (RAP), et d'autres coefficients utilisés en reconnaissance de la musique. Les sections suivantes traitent des différentes approches de la reconnaissance des événement acoustiques (REA).

3.1 Approches fondées sur le système auditif humain

Le système auditif humain a toujours été une importante source d'inspiration pour les systèmes de reconnaissance du son et pour les systèmes du type ASA (*Auditory Scene Analysis*). La compréhension du fonctionnement de l'oreille interne, en particulier de la cochlée, a contribué à l'obtention de modèles mathématiques pour ces mécanismes biologiques et à leur application à la reconnaissance du son.

3.1.1 Système auditif humain

La figure 3.1 [1] montre un diagramme du système auditif humain. Le son collecté par l'oreille externe traverse le conduit auditif externe et enclenche des vibrations au niveau du tympan. Le tympan convertit les vibrations externes, provoquées par la pression acoustique, en vibrations mécaniques qui se propagent dans la cavité tympanique (oreille moyenne). Les vibrations sont reçues par la chaîne ossiculaire de l'oreille moyenne composée du marteau, de l'enclume et de l'étrier. Les vibrations mécaniques sont transformées en ondes de compression qui traversent la fenêtre ovale et se propagent dans le milieu liquide de la cochlée, située dans l'oreille interne [Pickles, 2008].

La cochlée est un tube possédant une structure hélicoïdale de deux tours et demi de spire et abritant la membrane basilaire. Les vibrations de la membrane basilaire sont reçues par les cellules ciliées, qui se trouvent disposées tout au long de la cochlée. Elles sont ensuite transformées en stimuli nerveux transmis au cerveau via le nerf auditif (ou nerf vestibulocochléaire).

La première extrémité de la cochlée, connectée à la fenêtre ovale et à la fenêtre ronde, est appelée **base** et constitue la partie la plus fine et la plus rigide du tube. Le diamètre de celui-ci (et la largeur de la membrane basilaire) croît progressivement et atteint sa plus grande valeur à l'autre extrémité, plus large et plus souple, appelée **apex**. La base est de ce fait plus rigide que l'apex [Gelfand, 2004]. Lorsque une onde traverse la cochlée de la base vers l'apex, son amplitude atteint sa plus grande valeur à un endroit bien précis de la membrane basilaire et décroît ensuite très rapidement [Guy and

1. Fichier exploité dans le cadre d'une licence Creative Commons. Source :
http ://upload.wikimedia.org/wikipedia/commons/d/d2/Anatomy_of_the_Human_Ear.svg

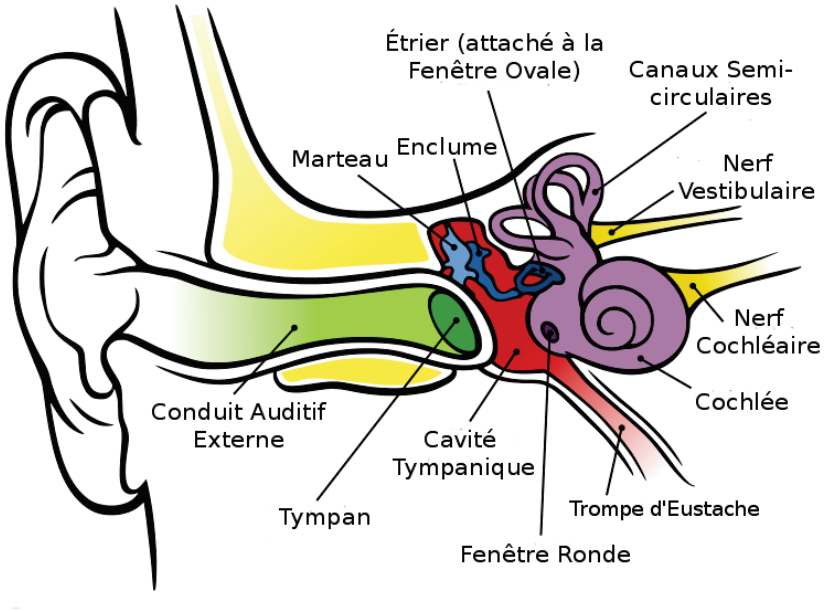

Figure 3.1: Diagramme du système auditif humain

Rémy, 2013]. Cet endroit dépend de la fréquence de l'onde. En effet, les ondes à hautes fréquences provoquent des vibrations plutôt près de la base. Elles perdent donc rapidement en amplitude et ne se propagent pas vers la fin de la membrane basilaire. Les ondes à basses fréquences, quant à elles, peuvent continuer à se propager le long de la membrane basilaire et sont reçues près de l'apex. La figure 3.2 illustre une **carte tonotopique** [Romani et al., 1975] [Talavage et al., 2004] [Gazzaniga et al., 2000] montrant les parties activées de la membrane basilaire et les bandes de fréquences correspondantes. Le modèle d'analyse fréquentielle effectuée par la cochlée rappelle celui d'une transformée de Fourier sur une fenêtre de signal. Il est toutefois plus avantageux car il garantit une analyse continue dans les domaines fréquentiel et temporel.

3.1.2 Filtres auditifs

En psychoacoustique, deux vibrations provoquées par deux stimuli auditifs différents, mais dont les fréquences sont assez proches, sont perçues comme émanant du même type de stimulus et elles sont interprétées comme des battements [Cook, 2001]. La raison de cela vient du fait que les deux fréquences provoquent une résonance au même endroit de la membrane basilaire. Une

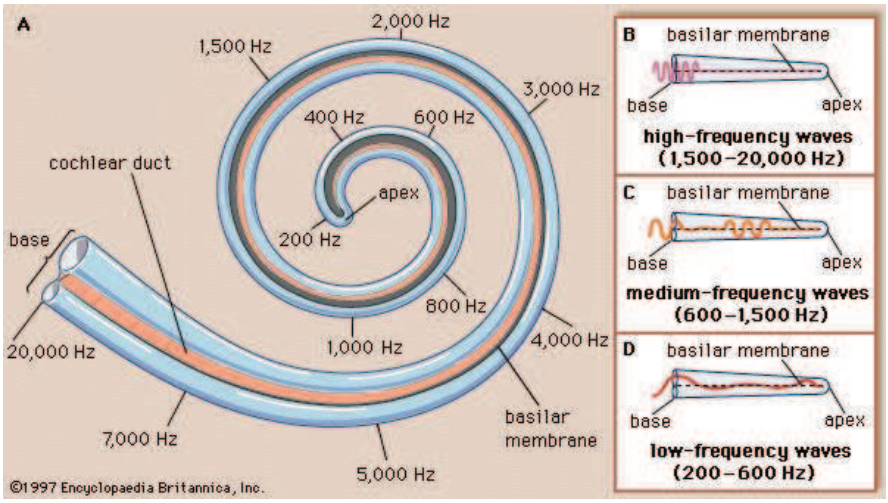

Figure 3.2: Analyse fréquentielle dans la cochlée

bande critique désigne une plage de fréquences perçues indifféremment par le système auditif humain. Pour que deux fréquences différentes soient perçues distinctement, elles doivent être suffisamment éloignées dans le spectre ; plus précisément, elles doivent appartenir à deux bandes critiques différentes [Gelfand, 2004] [Moore, 1986] [Deutsch, 1999] [Campbell and Greated, 1994] [Radocy and Boyle, 2003].

La figure 3.3 [2] illustre le principe de bande critique. F_c est la fréquence centrale de la bande. F_1 et F_2 sont les fréquences de coupure inférieure et supérieure respectivement. Les deux fréquences, qui sont choisies de sorte que la différence entre leurs amplitudes et l'amplitude maximale soit inférieure à 3dB, désignent la largeur de la bande critique.

Les propriétés physiques de la membrane basilaire et sa réponse varient tout au long de sa longueur. Cela lui permet donc de répondre différemment à deux fréquences différentes, ou plus précisément, à deux plages de fréquences différentes [Zemlin, 1998] [Alberti, 2001] [Munkong and Juang, 2008]. Depuis de très nombreuses années, plusieurs travaux se sont intéressés à la modélisation du fonctionnement de la cochlée, en proposant un ensemble de **filtres auditifs** qui, selon une définition des fréquences centrales et des largeurs de bandes associées, tentent de produire un modèle de perception qui s'apparente au système auditif humain [Lyon et al., 2010a].

2. Fichier exploité dans le cadre d'une licence Creative Commons. Source : http ://upload.wikimedia.org/wikipedia/commons/1/18/Band-pass_filter.svg

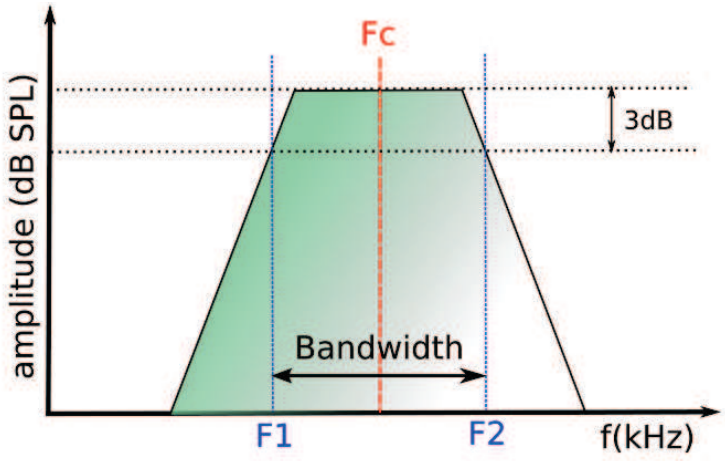

Figure 3.3: Exemple de bande critique

Le terme filtre auditif est étroitement lié à celui de bande critique. Le rôle d'un filtre auditif, en particulier un filtre passe-bande, est d'accentuer une bande de fréquences donnée tout en atténuant les fréquences au-delà de la bande [Gelfand, 2004].

Les variations mécaniques de la membrane basilaire, qui ont pour effet des réponses variables en fonction de la plage de fréquences, peuvent être modélisées par un tableau de filtres auditifs (pouvant être en chevauchement) [Munkong and Juang, 2008]. La notion de filtre auditif et celle de bande critique furent au début interchangeables [Jurado and Robledano, 2007]. Toutefois, le terme filtre auditif est aujourd'hui plus courant, car il permet de prendre en considération plusieurs aspects, notamment le phénomène de masquage qui n'est pas modélisé d'après la définition de la bande critique [Jurado and Robledano, 2007]. De surcroît, les différents filtres auditifs proposés dans les différents travaux, peuvent prendre différentes formes, bien plus complexes que la forme rectangulaire de la bande critique [Patterson, 1976] [Patterson and Henning, 1977] [Sommers and Humes, 1993] [Moore et al., 1995].

Les filtres *Gammatone* [Patterson et al., 1987] [Patterson et al., 1995] [Slaney, 1993] sont considérés comme l'une des modélisations du fonctionnement de la membrane basilaire les plus réputées. Ils sont basés sur un modèle de bandes critiques appelé *Equivalent Rectangular Bandwidth* (*ERB*). Ils ont été créés suite à plusieurs expérimentations impliquant des sujets jeunes ayant une audition normale [Moore and Glasberg, 1983] [Glasberg and Moore, 1990] [Greenwood, 1990] [Peters and Moore, 1992]. Le *ERB* est défini comme suit :

$$ERB = 24.7(4.37F + 1) \tag{3.1}$$

Plusieurs autres modèles ont également été proposés [Smith and Abel, 1999] [Lyon et al., 2010a]. L'un des modèle les plus intéressants est celui proposé par [Lyon, 1982]. Il est composé d'un réseau de filtres parallèles, invariables dans le temps, disposés en cascade.

3.1.3 Applications

Les coefficients basés sur les filtres auditifs modélisant le système auditif humain sont utilisés aussi bien en RAP et RAL qu'en reconnaissance des sons de l'environnement. Dans [Srinivasan and Wang, 2008] un modèle de reconnaissance des mots isolés en présence de plusieurs locuteurs est proposé. Des filtres *Gammatone* sont utilisés. Dans [Schluter et al., 2007], des coefficients basés sur les filtres *Gammatone* sont combinés avec plusieurs autres coefficients acoustiques pour la RAP. Plusieurs autres travaux sur la RAP utilisent des coefficients basés sur le modèle auditif de la cochlée : [Rademacher and Mertins, 2006] [Shao et al., 2009] [Minh and Lee, 2004] [Abdulla, 2002]. En RAL, on trouve également des travaux intéressants : [Zhang and Abdulla, 2005] [Zhao et al., 2012] [Abdulla and Zhang, 2010].

Plusieurs travaux en REA sont également basés sur des filtres modélisant le système auditif humain. [Anniés et al., 2007] utilisent des filtres *Gammatone* pour la reconnaissance des sons de pas. [Valero and Alías, 2012] utilisent une combinaison entre des filtres *Gammatone* et une analyse en odelettes pour la reconnaissance des sons de l'environnement. D'autres travaux incluent : [Hernandez et al., 2007] [Lin and Abdulla, 2007] [Leng et al., 2010] [Leng et al., 2012].

3.2 Approches fondées de la reconnaissance de la parole

Bien que les travaux sur la reconnaissance automatique de la parole soient très abondants, l'utilisation des coefficients *MFCC* avec des modèles de Markov cachés reste l'approche la plus courante [Baker et al., 2009]. Les coefficients *MFCC* sont calculés sur des fenêtres de signal de très courte durée, ce qui ne permet pas de conserver l'information temporelle du signal. Plusieurs techniques, tels que les coefficients *delta* et *double-delta* [Furui, 1981] [Kumar et al., 2011], la technique *RASTA* (*RelAtive SpecTrAl*) [Hermansky and Morgan, 1994] ou bien la technique *TRAPs* (*TempoRAl Pat-*

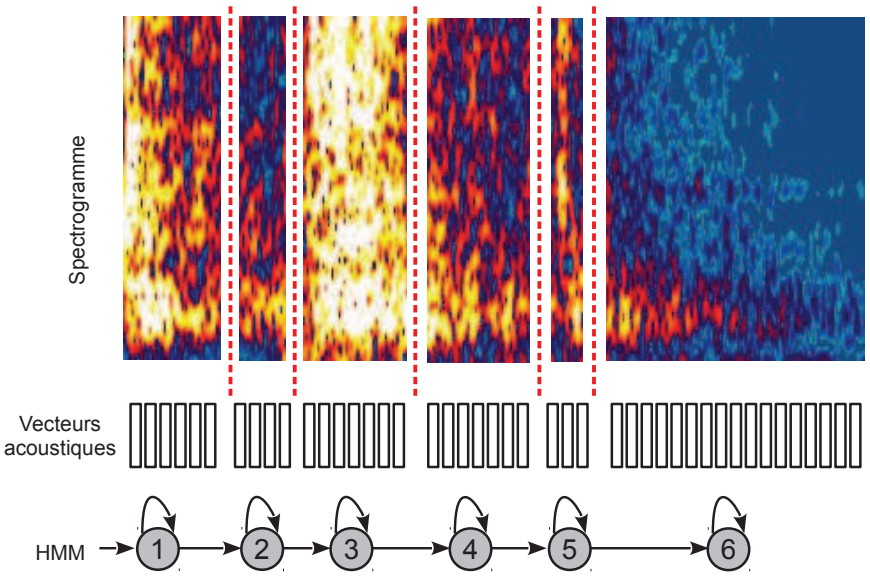

Figure 3.4: Spectrogramme d'un claquement de porte et HMM correspondant à l'évolution temporelle

terns) et ses variantes [Hermansky and Sharma, 1998] [Hermansky and Sharma, 1999] [Jain et al., 2002], sont utilisées pour incorporer l'information temporelle dans les vecteurs de coefficients acoustiques.

Les HMMs sont utilisés à un plus haut niveau pour modéliser l'information temporelle. À l'instar de la RAP, on suppose que la production des autres sons est un système markovien. La figure 3.4 illustre le spectrogramme d'un claquement de porte et une possible modélisation par un HMM. Cet exemple représente un type de son dont l'évolution dans le temps est bien claire (c'est le cas de presque tous les enregistrements de cette classe). Comme nous l'avons vu au chapitre 2 (figures 2.10 et 2.11), cela n'est pas le cas de toutes les classes de sons. Autrement dit, l'information temporelle pourrait, pour certains types de sons, ne pas être significative.

Cowling [Cowling and Sitte, 2002] propose une synthèse des méthodes issues de la RAP et utilisées pour la reconnaissance des sons de l'environnement. D'après l'auteur, les HMMs ne sont pas appropriés pour l'analyse des sons de l'environnement en raison de l'absence d'un alphabet phonétique. Nous restons toutefois sceptiques quant à cette hypothèse. Les HMMs peuvent être utilisés pour la reconnaissance des sons de l'environnement, comme en témoigne l'existence de plusieurs travaux les utilisant. De surcroît, les auteurs n'ont pas inclus les HMMs dans leurs tests, aucune comparaison avec les autres méthodes n'a donc été réalisée.

Parmi les autres méthodes qui y sont mentionnées on trouve la **déformation temporelle dynamique** (DTW pour *Dynamic Time Warping*), la **quantification vectorielle** (VQ pour *Vector quantization*) et les **réseaux de neurones artificiels** (ANNs pour *Artificial Neural Networks*).

Dans [Ma et al., 2006], des HMMs avec des coefficients *MFCC* sont utilisés. Les meilleurs résultats sont obtenus avec des HMMs de type gauche-droite ayant plusieurs états (en comparaison des modèles ergodiques ou ceux à un seul état). Cela confirme que la modélisation de l'information temporelle pourrait apporter un gain en reconnaissance. Dans [Xiang et al., 2010] une comparaison entre les performances des GMMs et des HMMs est faite. Les résultats obtenus confirment les résultats obtenus dans [Ma et al., 2006]. Dans [Eronen et al., 2006], une comparaison entre les performances d'un système basé sur HMMs et celles de sujets humains est réalisée. Les performances du système sont inférieures mais elles restent plutôt intéressantes, du fait que les auteurs utilisent un nombre réduit de coefficients acoustiques.

[Gaunard et al., 1998] utilise des HMMs discrets avec des coefficients *LPC* (*Linear Prediction Coding*). Les vecteurs *LPC* sont transformés en observations discrètes par VQ. Dans [Matos et al., 2006] des HMMs sont utilisés avec des coefficients *MFCC* pour la détection des toux dans des flux audio.

En résumé, les travaux utilisant des HMMs pour la reconnaissance des événements acoustiques, autres que la parole, sont assez similaires. Les coefficients utilisés peuvent parfois différer, mais ils sont souvent des *MFCC* ou bien des *LPC*. D'autres coefficients sont également utilisables (*delta*, *double-delta*, *ZCR*, *Spectral Roll-Off*, etc.). D'autres travaux peuvent être trouvés dans : [Ntalampiras et al., 2009] [Allegro et al., 2001] [Dufaux et al., 2000] [Zienowicz et al., 2008].

3.3 Approches fondées sur la reconnaissance du locuteur

Depuis le début des années 2000, des progrès très importants ont été réalisés en reconnaissance automatique du locuteur. Bien que certains travaux se soient intéressés à l'étude des paramètres acoustiques, les efforts se sont surtout concentrés sur les méthodes de classification et la modélisation de la variabilité entre les différents enregistrements d'un locuteur. Cette variabilité, plus communément appelée **variabilité de session** (*Session Variability*) [Kenny et al., 2007b] [Vogt and Sridharan, 2008], désigne toute différence entre deux enregistrements du même locuteur, et constitue un défi majeur en RAL [Kinnunen and Li, 2010a]. Elle peut être due à plusieurs facteurs :

changement de l'environnement acoustique, changement des outils ou des méthodes d'enregistrement, variations de la voix du locuteur dues à l'âge, à l'état émotionnel, à l'état de santé, etc. Des travaux relativement récents en RAL s'intéressent à la structuration en locuteurs (Speaker Diarization) des flux audio [Zhu et al., 2005] [Zhu et al., 2006]. Ces avancées en RAL, et d'autres encore, devront profiter aux travaux sur la reconnaissance des sons de l'environnement.

Beaucoup de de coefficients acoustiques utilisés en RAL viennent directement de la RAP, dont principalement : [Kinnunen and Li, 2010a] :

— Les coefficients spectraux à court terme (*MFCC*, *LPCC* pour *Linear Predictive Cepstral Coefficients* [Huang et al., 2001], *LSF* pour *Line Spectral Frequencies* [Huang et al., 2001], *PLP* pour *Perceptual Linear Prediction* [Hermansky, 1990], etc.)
— Les coefficients spectro-temporels (*delta*, *double-delta*, *TDCT* pour *Temporal Discreste Cosine Transform* [Kinnunen et al., 2008], etc.)
— Les caractéristiques prosodiques (débit, accentuation intonation, rythme, etc.) [Shriberg et al., 2005]
— Les caractéristiques de la source vocale (fréquence fondamentale ou F_0, impulsion glottique) [Shriberg et al., 2005] [Kinnunen and Alku, 2009] [Espy-Wilson et al., 2006]
— Les caractéristiques de haut niveau (vocabulaire utilisé, etc.) [Doddington, 2001]

Les trois dernières catégories de caractéristiques sont étroitement liées à la parole, elles ne sont, de ce fait, pas d'un grand intérêt pour la REA. Les méthodes de classification utilisées en RAL ont eu, en revanche, une grande influence sur la REA. Les sections suivantes présentent les principales méthodes ainsi que les travaux qui les ont retenues pour la reconnaissance des événements acoustiques.

3.3.1 Reconnaissance en utilisant la quantification vectorielle

La quantification vectorielle [Gersho and Gray, 1992] fut l'une des premières méthodes en RAL [Burton, 1987] [Soong et al., 1985] [Soong and Rosenberg, 1988] [He et al., 1999]. Elle est assez simple de principe. Deux séquences de vecteurs acoustiques $X = \{\mathbf{x}_1, \ldots, \mathbf{x}_M\}$ et $T = \{\mathbf{t}_1, \ldots, \mathbf{t}_N\}$ sont comparées comme suit :

$$D_Q(X,T) = \frac{1}{M}\sum_{m=1}^{M} \min_{1\leq n\leq N} d(\mathbf{x}_m, \mathbf{t}_n) \tag{3.2}$$

où D_Q est la **distorsion moyenne de quantification** (*Average Quantization Distortion*) entre les deux séquences, et $d(.,)$. est une mesure de distance telle que la **distance Euclidienne** ($\|\mathbf{x}_m - \mathbf{t}_n\|$). La séquence inconnue X est attribuée à la classe dont la séquence (d'apprentissage) T donne la plus petite valeur de D_Q. Pour des raisons d'efficacité, le nombre de vecteur est souvent réduit en utilisant une méthode de partitionnement de données tel que **K-moyennes** (K-means [Linde et al., 1980]) ou une de ses variantes (K-means++, Fuzzy c-means [Arthur and Vassilvitskii, 2007] [Cannon et al., 1986] etc.). La séquence est réduite en un petit nombre de vecteurs appelé **dictionnaire** (*Codebook*).

En dépit de quelques résultats prometteurs, en comparaison notamment avec les réseaux de neurones [Cowling and Sitte, 2002], cette méthode a été très peu testée en REA. Dans [Cowling and Sitte, 2003], plusieurs méthodes de classification et plusieurs types de coefficients sont comparés. les meilleurs résultats sont obtenus avec le DTW. La VQ était la deuxième meilleure méthode (devant les GMMs et les ANNs), en utilisant des coefficients issus d'une **transformée en ondelettes continue** (*Continuous Wavelet Transform*). Pour les *MFCC*, les résultats obtenus avec des GMMs étaient meilleurs que ceux obtenus avec la VQ.

3.3.2 Reconnaissance avec des GMMs

Les GMMs [Reynolds and Rose, 1995] [Reynolds et al., 2000] sont l'une des méthodes les plus utilisées en RAL. Pour classifier une séquence X, on considère indépendantes les observations qui la composent. De ce fait, étant donné un modèle GMM $\lambda_c = \{w_k, \mu_k, \mathbf{\Sigma}_i\}_{k=1}^{K}$, estimé à partir des données d'une classe c, la probabilité de X est donnée par :

$$p(X|\lambda_c) = \prod_{m=1}^{M} p(\mathbf{x}_m|\lambda_c) \tag{3.3}$$

En pratique, on utilise le log de la probabilité, ce qui donne :

$$\log p(X|\lambda_c) = \sum_{m=1}^{M} \log p(\mathbf{x}_m|\lambda_c) \tag{3.4}$$

La séquence X est attribuée à la classe la plus probable.

Dans [Chu et al., 2009], les GMMs sont utilisés pour la classification des sons de l'environnement. Des coefficients *MFCC* avec des coefficients extraits par une méthode appelée *Matching Pursuit* sont utilisés. Dans [Vacher et al., 2008], une évaluation de la reconnaissance des sons de détresse pour une application de télé-médecine est réalisée. Des HMMs et des GMMs sont testés avec des coefficients *LFCC* (*Linear-Frequencies Cepstral Coefficients*). Les meilleurs résultats sont obtenus avec des HMMs. Dans [Ito et al., 2011], plusieurs modèles GMMs par classe de sons sont créés en utilisant une procédure itérative. À chaque étape, un modèle est estimé, puis les données utilisées en apprentissage sont évaluées par rapport au modèle obtenu. Les vecteurs avec la plus faible probabilité sont utilisés à l'étape suivante pour créer le prochain modèle.

Comme en reconnaissance du locuteur, beaucoup de travaux introduisant d'autres méthodes, incluent les GMMs comme base de comparaison. D'autres travaux utilisant des GMMs peuvent être trouvés dans [Istrate et al., 2006a] [Istrate et al., 2006b] [Vacher et al., 2004] [Sivasankaran and Prabhu, 2013] [Bahoura and Pelletier, 2004].

3.3.3 Reconnaissance avec des SVMs : classification de fenêtres isolées)

Les premiers travaux utilisant les SVMs pour la RAL suivaient le même schéma que celui des GMMs, c'est-à-dire, les vecteurs acoustiques d'une séquence X étaient classifiés séparément. En utilisant une stratégie d'agrégation, un score est calculé pour la séquence toute entière. Les deux stratégies d'agrégations utilisées sont le vote majoritaire ou bien la somme des distances qui résultent de la classification des vecteurs [Schmidt and Gish, 1996] [Wan and Campbell, 2000].

Cette approche est utilisée dans [Ghiurcau and Rusu, 2010] avec des coefficients *MFCC* en entrée. Dans [Wang et al., 2008], elle est appliquée pour la reconnaissance des sons dans une maison intelligente. Les vecteurs *MFCC* sont préalablement transformés par une **analyse en composantes indépendantes** (ICA pour *Independent Component Analysis*) ou par une **analyse en composantes principales** (PCA pour *Principal Component Analysis*) pour réduire la taille des données.

Cependant, cette approche est source d'un problème très important. Les vecteurs acoustiques n'étant en effet pas linéairement séparables, l'utilisation de modèles SVM avec le noyau RBF constitue souvent un meilleur choix en comparaison avec le noyau linéaire. Toutefois, les modèles obtenus sont très volumineux car la plupart des vecteurs sont retenus comme vecteurs de support. De plus, les valeurs de certains paramètres du modèle (paramètre d'apprentissage C et Γ du noyau RBF) ne doivent pas être choisies arbitrairement. Une recherche exhaustive [Hsu et al., 2010] très longue est souvent nécessaire pour déterminer les meilleures valeurs pour chaque modèle SVM. Pour accélérer cette procédure, le nombre de vecteurs peut être réduit en utilisant une méthode tel que K-moyennes.

3.3.4 Reconnaissance avec des SVMs : noyaux de classification de séquences

Une manière d'éviter l'utilisation des vecteurs acoustiques directement en entrée d'un SVM est de transformer une séquence de vecteurs en un seul vecteur de taille fixe. Certains travaux en REA ont recours à cette idée en calculant souvent un vecteur de coefficients statistiques depuis la séquence de vecteurs acoustiques, tel que dans [Guo and Li, 2003]. La même méthode est utilisée dans [Temko and Nadeu, 2005] pour la classification des événements acoustiques en salle de réunion (plusieurs types de paramètres acoustiques sont utilisés : *MFCC*, *ZCR*, *Short time energy*, *Subband energies*, *Filter-Bank Energies*). De meilleurs résultats sont obtenus avec des SVMs en comparaison des GMMs. Dans [Chen et al., 2006b] les sequences de caractéristiques sont remplacées par la moyenne, le minimum et le maximum. Ce type d'approches n'est pourtant pas récent, le même principe fut en effet utilisé en RAL [Markel et al., 1977], avant que les méthode statistiques tels que les GMMs ne s'imposent.

Le succès des SVMs comme méthode de classification aux bases théoriques très solides est soutenu par une efficacité en pratique pour plusieurs tâches de classification. Avec une importante expérience en matière de méthodes génératives (GMMs, HMMs), et quelques premiers résultats prometteurs en utilisant les SVMs, la communauté de la RAL a proposé un certain nombre de méthodes de transformation de séquences, particulièrement adaptées au problème de la reconnaissance du locuteur. L'objectif de ces méthodes est de contourner les problèmes dus à l'utilisation des vecteurs acoustiques directement en entrée d'un SVM

Pour atteindre cet objectif, l'idée de base de ces méthodes est de transformer une séquence d'un nombre arbitraire de vecteurs acoustiques en un seul vecteur, appartenant à un espace vectoriel souvent plus grand, et d'éviter ainsi de comparer des séquences de taille différente. Les vecteurs

obtenus après transformation sont, de surcroît, linéairement séparables, ce qui signifie qu'un noyau SVM linéaire peut être utilisé en classification (au lieu du noyau RBF utilisé avec les vecteurs acoustiques).

Les **noyaux SVM de discrimination de séquences** (*Sequence Discriminant Kernels*) utilisés en RAL furent précédés d'un certain nombre de travaux, dont certains ont eu une influence certaine.

Les travaux de [Jaakkola and Haussler, 1998] ont donné naissance au noyau Fisher, un des noyaux les plus utilisés pour la classification de séquences. Cette approche est généralisée dans [Smith and Gales, 2002] pour une utilisation en reconnaissance de la parole. L'idée de ce type de noyaux est d'exploiter les modèles génératifs pour transformer une séquence en un seul vecteur, en calculant la dérivée première du score de vraisemblance d'une séquence :

$$\psi_{fisher}(X) = \nabla_\theta \log(X|\lambda, \theta) \tag{3.5}$$

où λ est le modèle Gaussien paramétré par l'ensemble de paramètres θ. Appliquée à la classification de séquences biologiques, cette méthode a donné de meilleurs résultats que les HMMs [Jaakkola and Haussler, 1998].

Depuis, plusieurs autres noyaux de discrimination de séquences ont été proposés tels que DTAK (*Dynamic Time Alignment Kernel*) [Noma, 2002], PolyDTW (*Polynomial Dynamic Time Warping*) [Wan and Carmichael, 2005] ou bien celui basé sur le GDTW (*Gaussian Dynamic Time Warping*) [Bahlmann et al., 2002].

Après le noyau PolyDTW, plusieurs noyaux, en RAL, ont également été proposés ou adaptés à partir d'autres domaines. On utilise souvent le terme **super vecteur** (*Super Vector*) pour se référer au vecteur issu de la transformation d'une séquence. Le terme super vecteur est d'habitude utilisé pour un vecteur obtenu par la transformation du noyau GSL. [Kinnunen and Li, 2010a] l'utilisent pour désigner tout vecteur (unique, et souvent d'une très grande taille) issu d'une transformation d'une séquence de vecteurs, et utilisé en entrée d'un SVM.

L'idée du noyau GLDS (*Generalized Linear Discriminant Sequence Kernel*) [Campbell et al., 2006a] est de transformer chaque vecteur acoustique $\mathbf{x}_i$ de la séquence X en un vecteur $\mathbf{b}(\mathbf{x}_i)$. Celui-ci contient tous les monômes de degré d, calculés depuis les caractéristiques du vecteur $\mathbf{x}_i$. Cette procédure est baptisée *Polynomial Expansion* [Campbell et al., 2002]. Pour un vecteur à

deux dimensions, $\mathbf{x} = (x_1, x_2)^T$, $\mathbf{b}(\mathbf{x}) = (1,\ x_1,\ x_2,\ {x_1}^2,\ x_1x_2,\ {x_1}^2)^T$. Le super vecteur GLDS d'une séquence de longueur M est la moyenne des vecteurs $\mathbf{b}(\mathbf{x})_i$: $\mathbf{b}_{\text{avg}} = \frac{1}{M}\sum_{i=1}^{M}\mathbf{b}(\mathbf{x}_i)$.

Le noyau SVM-GSL (*SVM-GMM Supervector Linear Kernel*) [Campbell et al., 2006b] [Dehak and Chollet, 2006] est le fruit d'une combinaison entre GMM et SVM. Son idée de base est d'adapter un modèle GMM du monde λ_{UBM} (UBM pour *Universal Bachground Model*, un modèle GMM créé en utilisant des données en provenance de tous les locuteurs) par la procédure MAP (*Maximum a Posteriori*) [Gauvain and Lee, 1994], en utilisant une séquence de vecteurs X en entrée de l'algorithme d'adaptation. Les vecteurs moyens du modèle adapté qui en résulte, λ_X, sont « empilés » pour former un seul vecteur. Celui-ci remplacera la séquence X et sera utilisé en entrée d'un SVM. Cette approche sera décrite plus en détails au chapitre 4.

Enfin, le noyau SVM-MLLR Supervector utilise la méthode MLLR (*Maximum Likelihood Linear Regression*) [Leggetter and Woodland, 1995] comme méthode d'adaptation d'un HMM (au lieu d'un GMM) [Stolcke et al., 2007].

Les noyaux de classification de séquences, devenus une approche standard en RAL, n'ont reçu que très peu d'intérêt en REA. [Temko et al., 2005] fait une comparaison entre un certain nombre de noyaux pour la classification des événements acoustiques en salle de réunion. Le noyau Fisher a obtenu les meilleures performances en comparaison avec les noyaux GDTW, DTAK et PolyDTW. Les performances de ce dernier sont comparables à celles d'un système à base de GMMs. GDTW et DTAK ont eu des résultats inférieurs. Ce travail a eu lieu avant l'introduction des noyaux GLDS, SVM-GSL et SVM-MLLR. Depuis, à notre connaissance, aucune autre comparaison n'a été faite.

Dans [Zhuang et al., 2010], le noyau SVM-GSL est utilisé dans un système de détection des événements acoustiques dans des enregistrements audio. Le système opère en deux phases. Les événements sont d'abord détectes et pré-classifiés en utilisant un approche hybride, combinant des HMMs avec des réseaux de neurones. Les segments détectés sont ensuite classifiés en utilisant SVM-GSL.

Dans le cadre de cette thèse, nous avons comparé les performances du noyau SVM-GSL avec avec celles des GMMs, pour la classification des sons de la vie quotidienne [Sehili et al., 2012a]. Nous l'avons également retenu pour la classification des événements acoustiques en utilisant une base de données enregistrée dans une maison intelligente [Sehili et al., 2012b].

Plusieurs avancées récentes en RAL, souvent étroitement liées aux deux principales méthodes du

domaine, GMM et SVM, peuvent être consultées dans la littérature. Bien qu'elles ne soient pas retenues dans cette thèse, nous pensons, d'après les résultats que nous avons obtenus en utilisant des techniques issues de la RAL, qu'elles constitueront une bonne perspective de recherche. Nous donnons, dans les paragraphes suivants, une brève description de quelques-unes des techniques les plus intéressantes.

***Nuisance Attribute Projection* (NAP) :** La méthode NAP est utilisée pour réduire la variabilité inter-sessions entre les différents enregistrements d'un locuteur [Solomonoff et al., 2005] [Campbell et al., 2006c]. Elle peut être appliquée aux super vecteurs (de n'importe quel type) avant qu'on les utilise en entrée d'un SVM. La transformation NAP d'un super vecteur $\mathbf{x}$ est :

$$\hat{\mathbf{x}} = \mathbf{x} - \mathbf{U}(\mathbf{U}^T\mathbf{x}) \tag{3.6}$$

où $\mathbf{U}$ est la matrice des **canaux propres** (*Eigenchannel Matrix*) calculée en utilisant des données de développement contenant des enregistrements de plusieurs locuteurs, réalisés sur plusieurs sessions pour chacun. Une fois la matrice $\mathbf{U}$ connue, elle pourra être utilisée pour transformer les vecteurs à reconnaître. la transformation dans l'équation 3.6 a pour objectif de soustraire le super vecteur obtenu par projection sur l'espace des canaux du super vecteur initial. D'autres méthodes comparables à la méthode NAP existent également. Leur but étant de réduire la variabilité intra-classe des vecteurs d'un locuteur et de maximiser la variabilité inter-classes de tous les locuteurs. Les deux méthode les plus utilisées sont WCCN (*Within-Class Covariance Normalization*) [Hatch and Stolcke, 2006] et LDA (*Linear Discriminant Analysis*) [Vogt et al., 2008].

***Factor Analysis* (FA) :** La technique d'analyse factorielle est comparable au NAP mais elle utilisée avec des modèles génératifs (GMMs) pour modéliser la variabilité inter-sessions.

En vérification du locuteur, le modèle GMM d'un locuteur est estimé en adaptant un modèle du monde, λ_{UBM}, par une procédure MAP (voir la section 4.2 pour plus de détails sur le MAP). Le MAP permet d'adapter les vecteurs moyens, les poids et les matrices de covariances d'un modèle UBM, mais en pratique seuls les vecteurs moyens dont adaptés. De ce fait, le modèle d'un locuteur particulier est représenté par l'ensemble des vecteurs moyens adaptés, les poids et les matrices de covariances étant les mêmes pour tous les locuteurs.

La technique JFA (*Joint Factor Analysis*) [Kenny et al., 2007a] [Kenny et al., 2007b] [Kenny

et al., 2008] a été proposée pour modéliser explicitement la variabilité inter-sessions du modèle de locuteur. Comme mentionné au paragraphe précédent, le modèle du locuteur est représenté par le vecteur $\mathbf{M}$, composé des vecteurs moyens adaptés. Le vecteur $\mathbf{M}$ est une combinaison linéaire des composantes représentant le locuteur et celles représentant le canal (ou la session). De ce fait, il peut être écrit comme suit :

$$\mathbf{M} = \mathbf{m} + \mathbf{Vy} + \mathbf{Ux} + \mathbf{Dz} \tag{3.7}$$

où $\mathbf{m}$ est un vecteur indépendant du locuteur et du canal (en général, il est composé des vecteurs moyens du modèle UBM), $\mathbf{V}$ est la matrice de de voix propres (*Eigenvoices*) représentant le locuteur, $\mathbf{U}$ la matrice des canaux propres représentant le canal et $\mathbf{D}$ la matrice diagonale des résidus. $\mathbf{V}$ et $\mathbf{U}$ sont des matrices de rang réduit (*Low Rank*) et $\mathbf{D}$ est une matrice diagonale. $\mathbf{x}$ et $\mathbf{y}$ sont deux variables aléatoires centrées réduites normalement distribuées ($\mathcal{N}(\mathbf{I}, 0)$) représentants les **facteurs** du canal et ceux du locuteur respectivement. Les matrices $\mathbf{V}$, $\mathbf{U}$ et $\mathbf{D}$ sont estimées sur de larges données de développement [Kenny, 2005] [Kenny et al., 2008]. Les vecteurs $\mathbf{x}$, $\mathbf{y}$ et $\mathbf{z}$ sont estimés pour chaque locuteur lors de l'apprentissage. Après la suppression des composantes du canal, le vecteur représentant le locuteur, $\mathbf{s}$, s'écrira comme suit :

$$\mathbf{s} = \mathbf{m} + \mathbf{Vy} + \mathbf{Dz} \tag{3.8}$$

***Identity Vector* (i-vector) :** Dans la méthode JFA, le locuteur et le canal sont représentés par deux espaces distincts (matrices $\mathbf{V}$ et $\mathbf{U}$, équation 3.7). D'après [Dehak, 2009], les facteurs de canal qui modélisent normalement les effets du canal contiennent également des informations sur le locuteur. De ce fait, [Dehak et al., 2011] proposent une nouvelle méthode d'analyse qui ne fait pas de distinction entre le locuteur et le canal :

$$\mathbf{M} = \mathbf{m} + \mathbf{Tw} \tag{3.9}$$

où $\mathbf{m}$ est un vecteur indépendant du locuteur et du canal (super vecteur composé des moyennes du modèle UBM), $\mathbf{T}$ est une matrice de rang réduit et $\mathbf{w}$ est une variable aléatoire centrée réduite normalement distribuée. Les composantes du vecteur $\mathbf{w}$ sont appelées **facteurs totaux** (*Total Factors*) et forment le **vecteur d'identité** ou **i-vector** qui pourra être utilisé en entrée d'un SVM. Plus de détails sur le calcul du i-vector pouvent être trouvés dans : [Dehak et al., 2011], [Senoussaoui

et al., 2010] et [Bousquet et al., 2011].

3.4 Approches fondées sur les techniques du traitement d'image

Une autre tendance récente et très prometteuse en matière de reconnaissance des événements acoustiques fait appel à quelques techniques utilisées en vision par ordinateur. L'idée de base de ces approches est de transformer le signal audio en une représentation visuelle de deux dimensions, qui sera traitée comme une image.

Ce type d'approches trouve ses origines dans les domaines voisins. En reconnaissance de la musique par exemple, [Ke et al., 2005] utilise la célèbre méthode de Viola et Jones [Viola and Jones, 2001] [Viola and Jones, 2004] pour la détection d'objets. Un grand nombre de caractéristiques du type *Haar-like* sont extraites depuis le spectrogramme du signal et sélectionnées en utilisant AdaBoost [Freund and Schapire, 1997] [Schapire and Singer, 1999].

Dans [Liao et al., 2011], des caractéristiques appelées *Local Binary Pattern* sont également extraites depuis le spectrogramme du signal et utilisées en classification avec les **K plus proches voisins** (KNN ou *K-Nearest Neighbors*).

[Dennis et al., 2011] proposent des caractéristiques appelées *SIF* (*Spectrogram Image Feature*) utilisées avec les SVMs. Des performances nettement supérieures à celles d'un système du type *MFCC*-HMM ont été obtenues, même en présence du bruit. Les auteurs ont pu améliorer leur système en utilisant une représentation visuelle basée sur les filtres *Gammatone* [Patterson et al., 1987] (caractéristiques appelées *SPD* pour *Subband Power Distribution.*) L'utilisation des filtres *Gammatone* permettrait une meilleure exploitation de l'information temporelle [Dennis et al., 2013b].

Dans [Janvier et al., 2012] on trouve un autre exemple de l'utilisation des filtres *Gammatone*. La représentation visuelle générée est appellée *SAI* (*Stabilized Auditory Image*) [Lyon et al., 2010b]. Les performances obtenues sont néanmoins comparables avec celles d'un système *MFCC*-HMM.

[Lyon et al., 2010b] [Lyon, 2010] utilisent également, pour la génération d'une représentation visuelle du signal, des filtres basés sur le modèle de la cochlée. Des travaux plus récents utilisant les techniques de traitement d'images peuvent être trouvés dans [Dennis et al., 2012] et [Dennis et al., 2013a].

3.5 Conclusions

Bien que l'intérêt pour la reconnaissance des sons de l'environnement ait commencé il y a longtemps, ce domaine se trouve encore à un stade très préliminaire. En dépit de quelques premiers travaux isolés dans les années 90, la majorité des publications sont en effet très récentes. Les efforts de recherche devront se poursuivent pendant de nombreuses années encore avant que le domaine trouve sa ou ses méthodes de référence, à l'instar de la reconnaissance de la parole ou la reconnaissance du locuteur.

Une grande quantité de travaux exploitent des techniques utilisées en RAP ou en RAL (coefficients acoustiques et algorithmes de classifications) ce qui n'est, de prime abord, pas une mauvaise idée. Le problème est que, hélas, beaucoup de travaux se ressemblent, sans qu'il n'y ait beaucoup d'efforts de comparaison. La comparaison entre les travaux est rendue difficile par les besoins différents des applications et par l'utilisation de bases de données différentes, souvent enregistrées en partie en laboratoire, et complétées par des ressources que l'on trouve sur la toile.

Les noyaux SVM de discrimination de séquences, notamment ceux utilisés en RAL, n'ont pas fait l'objet d'une étude complète et d'une comparaison en REA. Nous avons, dans cette thèse, étudié ces méthodes et en retenu une (SVM-GSL) pour la tester avec la reconnaissance des sons de l'environnement. D'autres avancées en RAL sont encore à explorer.

Les filtre auditifs modélisant le fonctionnement de cochlée ont initialement été proposés en vue d'une utilisation en RAP et en RAL.ils ont visiblement longtemps été en arrière-plan par rapport à d'autres approches. Ils ont en effet été loin de pouvoir concourir avec les techniques existantes qui fonctionnent déjà très bien (notamment les coefficients *MFCC*). Dans un article intitulé *Towards increasing speech recognition error rates*, [Bourlard et al., 1996] avançaient le fait que, pour qu'une nouvelle approche soit évaluée à sa juste valeur, la communauté de la RAP doit, pour un certain temps, tolérer une baisse en performances. L'article mentionne des filtres auditifs entre autres. Lyon [Lyon, 2010], de son côté, pense qu'une meilleure alternative à cela serait d'utiliser ces techniques dans un domaine « mal-desservi », en l'occurrence pour les filtres auditifs, celui de la REA.

Depuis quelques années, l'intérêt pour les filtres auditifs a effectivement commencé à trouver sa place en REA. Une utilisation intelligente des techniques du traitement d'image a permis de redécouvrir la puissance de ces filtres. Les applications qui s'y intéressent sont très récentes, la plupart des travaux étant publiés en parallèle avec le travail réalisé dans cette thèse.

Enfin, nous clôturons ce chapitre par un certain nombre de points dont nous croyons qu'ils méritent une attention particulière en REA :

Coefficients acoustiques : La plupart des travaux emploient les coefficients utilisés et optimisés pour la RAP. Rien ne prouve qu'ils soient appropriés pour les sons de l'environnement. Intuitivement, on pourrait facilement penser que la différence de l'enveloppe d'amplitude du signal (ou du spectrogramme) entre certaines classes de sons suggère l'exploitation d'autres caractéristiques, probablement moins complexes que les *MFCC*, pour les discriminer. La « redécouverte » des filtres auditifs en REA constitue un bon enseignement. Nous abordons le sujet des coefficients acoustiques au chapitre 5.

Information temporelle : Les résultats obtenus avec des HMMs sont souvent meilleurs que ceux des GMMs. Pour certaines classes du moins, l'incorporation de l'information temporelle pourrait être bénéfique. Cependant, l'exploitation des HMMs en REA est moins étudiée en comparaison avec la RAP. Le nombre d'états des modèles HMM utilisés est souvent fixe et est le même pour toutes les classes de sons. Des techniques plus avancées devraient être utilisées pour une meilleure exploitation des HMMs.

Techniques de compensation de variabilité inter-session (*Inter-session Variability Compensation*) : Utilisées en RAL, elles permettent de réduire les différences entres des enregistrements du même locuteur. Elles semblent très intéressantes pour la REA, notamment pour les classes de sons dont les enregistrements proviennent de sources et d'environnements différents.

Bases de données : Difficile de comparer les différents travaux sans utiliser les mêmes données. Les chercheurs travaillant sur la même problématique en REA (REA dans une maison intelligente dans notre cas) devraient faire plus d'efforts pour partager leurs ressources. Après notre publication à Eusipco 2012 [Sehili et al., 2012a], nous avons été contactés par plusieurs autres chercheurs pour récupérer notre base de données.

CHAPITRE 4

MÉTHODES MISES EN ŒUVRE

Dans ce chapitre, nous décrivons les méthodes que nous avons retenues pour la reconnaissance des sons de l'environnement. Comme nous l'avons mentionné dans les deux chapitres précédents, ces méthodes s'inspirent des méthodes utilisées en reconnaissance du locuteur. Les expérimentations ont été réalisées en utilisant deux bases de données différentes. La première base contient des enregistrements audio découpés, de plusieurs sons de la vie courante. La seconde base est créée dans le cadre du projet SWEET-HOME. Elle contient des enregistrements continus de scénarios de la vie courante, enregistrés dans un maison intelligente, en présence de plusieurs microphones. Les deux bases seront décrites plus en détails dans ce chapitre.

4.1 Méthodes retenues

La figure 4.1 montre la chronologie du travail réalisé sur les méthodes de REA (reconnaissance des événements acoustiques) retenues dans cette thèse. L'équipe ANASON de l'ancien laboratoire de l'ESIGETEL, LRIT (Laboratoire de Recherche et d'Innovation Technologique à Fontainebleau), avait déjà travaillé sur un système basé sur des GMMs. Notre première motivation fut donc d'explorer d'autres techniques. L'intérêt pour les SVMs comme méthode de classification nous a incités à prendre cette direction ; c'est-à-dire, à utiliser une méthode discriminative comme alternative aux GMMs qui représentent un modèle génératif. Dans ce qui suit, nous donnons une brève description des méthodes retenues.

SVM-frame-level : La première méthode testée, que nous appelons SVM-frame-level, est comparable aux méthodes décrites dans la section 3.3.3. Autrement dit, les vecteurs acoustiques étaient directement utilisés en entrée d'un SVM. Après maintes expérimentations (tests avec plusieurs types de noyaux, tests de plusieurs schémas de classification multi-classe, etc.) il s'est avéré que, pour une utilisation efficace de cette approche [Sehili et al., 2010], mis à part la normalisation des données, les meilleurs paramètres de chaque modèle SVM doivent être déterminés par recherche exhaustive [Hsu et al., 2010]. À moins que l'on réduise la taille des données (au prix d'une perte en information, en utilisant K-means par exemple), cette recherche peut être extrêmement gourmande en temps de calcul. D'après notre expérience, une semaine nous a été nécessaire, en utilisant un

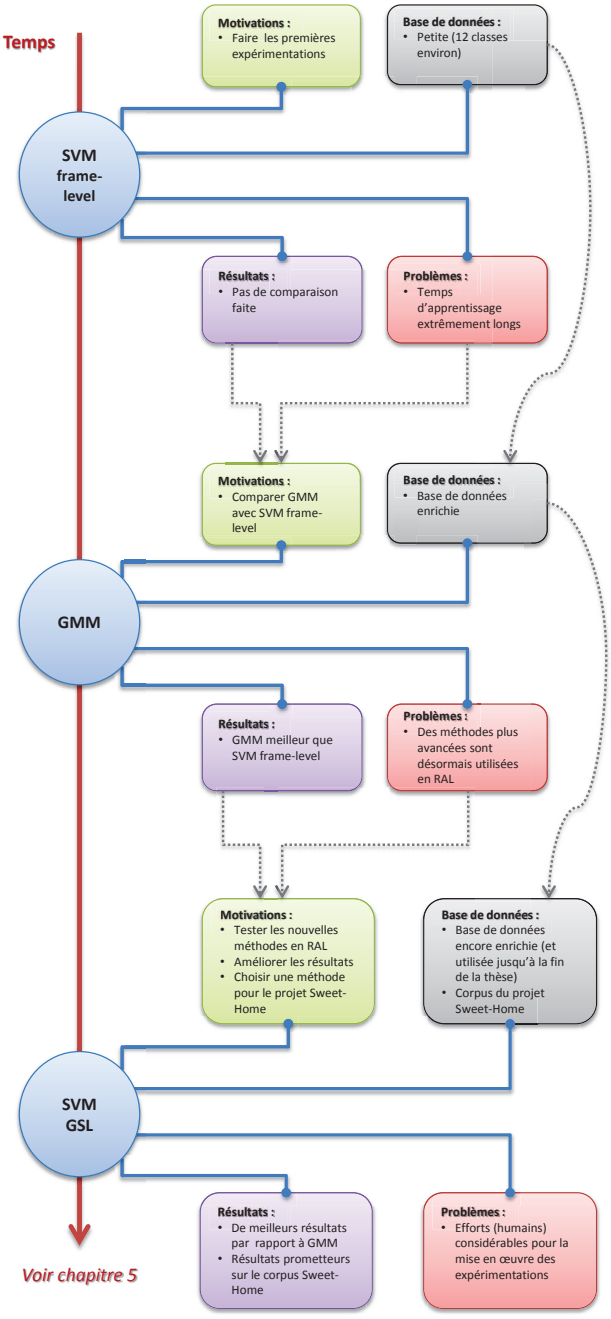

Figure 4.1: Évolution du travail sur les méthodes de reconnaissance

réseau de cinq ordinateurs de bureau (tournant avec un processeur de 2.50GHz $\times$ 4 et 4Go de mémoire chacune) , pour terminer les traitements. L'utilisation de plusieurs coefficients (en plus des *MFCC*), qui augmenterait la taille des données, n'était envisageable avec cette méthode.

GMM : L'utilisation des GMMs fut motivée par deux éléments : accélérer les temps de traitement et avoir une base de comparaison avec SVM-frame-level. Les résultats obtenus avec des GMMs étaient meilleurs que ceux de SVM-frame-level [Vacher et al., 2011]. Nous avons donc décidé d'abandonner cette dernière pour le reste de cette thèse.

SVM-GSL : L'étude de la littérature de la RAL (reconnaissance automatique du locuteur) nous a permis de découvrir certaines des avancées importantes que la communauté a réalisées depuis quelques années. En ce qui concerne les méthodes de reconnaissance, les noyaux SVM de classification de séquences semblent être une alternative intéressante aux solutions classiques basées sur des GMMs. Étant nous-mêmes en quête de méthodes efficaces pour la REA, et vu notre expérience avec la méthode SVM-frame-level, il nous semblait raisonnable de nous tourner vers ces nouvelles approches. Parmi les noyaux qu'on trouve dans la littérature, GLDS et GSL semblaient les plus aisés à mettre en œuvre, en plus de donner des résultats intéressants. Dans [Fauve et al., 2007], de meilleurs résultats sont obtenus avec le noyau GSL en comparaison avec GLDS. Nous avons donc retenu le noyau GSL pour le comparer aux GMMs.

Ce choix comporte tout de même un risque car, si le noyau GLDS n'exige particulièrement pas beaucoup de données, le modèle UBM, utilisé dans la transformation du noyau GSL, requiert souvent une quantité importante de données. En RAL, plusieurs dizaines (ou centaines) d'heures de parole sont souvent utilisées pour créer le modèle UBM [Reynolds et al., 2000]. Ce n'est pas le cas des données qu'on utilisera dans les expérimentations (voir tableau 4.1, un tiers de la base est utilisé pour créer le modèle UBM). De surcroît, lors de la dérivation des super vecteurs, les enregistrements utilisés (un seul à la fois, voir figure 4.3 plus loin) pour adapter le modèle peuvent être très courts. Pour être plus précis, certains sons ne durent guère plus qu'une ou deux secondes (des claquements de portes, des chocs ou des éternuements, par exemple). Ces facteurs pourraient rendre difficile l'utilisation du noyau GSL pour la REA.

4.2 Description du noyau SVM-GSL

L'objectif des noyaux SVM de discrimination de séquences est d'éviter l'utilisation des vecteurs acoustiques directement en entrée d'un SVM. [Fauve et al., 2007] donnent une définition générale d'un noyau SVM de discrimination de séquences. Pour deux séquences de vecteurs X et Y, le noyau est défini par :

$$K(X,Y) = \mathbf{\Phi}(X)^t \mathbf{R^{-1}} \mathbf{\Phi}(Y) \tag{4.1}$$

où $\mathbf{\Phi}(X)$ est une transformation (*Mapping*) de la séquence X en un vecteur appartenant à un autre espace vectoriel et $\mathbf{R}$ est une matrice diagonale de normalisation. De ce fait, pour obtenir un noyau de discrimination de séquences, nous avons donc besoin de définir la fonction de transformation $X \rightarrow \mathbf{\Phi}(X)$ et la matrice de normalisation $\mathbf{R}$.

Moreno [Moreno et al., 2003] proposait ce qui pourrait être vu comme l'ancêtre du noyau SVM-GSL. La distance entre deux séquences X et Y est définie par la **divergence de Kullback-Leibler** des modèles GMM, λ_X et λ_Y, qui les représentent respectivement. N'étant pas symétrique, la divergence Kullback-Leibler ne satisfait pas les *conditions de Mercer* d'une fonction de noyau SVM (symétrique et semi-définie positive). Au lieu d'utiliser la divergence de Kullback-Leibler, [Campbell et al., 2006b] proposent d'utiliser une approximation de celle-ci. Le noyau SVM-GSL est, de ce fait, défini par :

$$\begin{aligned} K(\lambda_X, \lambda_Y) = & \sum_{k=1}^{K} (\sqrt{w_k} \mathbf{\Sigma}_k^{-(1/2)} \mu_k^X)^T (\sqrt{w_k} \mathbf{\Sigma}_k^{-(1/2)} \mu_k^Y) \\ = & \ \mathbf{\Phi}_{\mathrm{GSL}}(X) \mathbf{R}_{\mathrm{GSL}}^{-1} \mathbf{\Phi}_{\mathrm{GSL}}(Y) \end{aligned} \tag{4.2}$$

La transformation $\mathbf{\Phi}_{\mathrm{GSL}}(X)$ (équation 4.3) est définie par les vecteurs moyens $\mu_k^X (k = 1, \ldots, K)$ du modèle GMM, λ_X, adapté d'un modèle GMM universel λ_{UBM} en utilisant la séquence X.

$$\mathbf{\Phi}_{\mathrm{GSL}}(X) = \begin{bmatrix} \mu_1^X \\ \mu_2^X \\ \vdots \\ \mu_K^X \end{bmatrix} \tag{4.3}$$

La matrice de normalisation $\mathbf{R}_{\mathrm{GSL}}$ est définie en utilisant les poids (w_i) et les matrices de covariances $(\mathbf{\Sigma}_i)$ du modèle λ_{UBM} (équation 4.4). Si $\mathbf{r}_{\mathrm{GSL}}$ est la diagonale de $\mathbf{R}_{\mathrm{GSL}}$ alors :

$$\mathbf{r}_{\mathrm{GSL}}^{-\frac{1}{2}} = \begin{bmatrix} \sqrt{w_1}\ \mathrm{diag}(\mathbf{\Sigma}_1^{-\frac{1}{2}}) \\ \sqrt{w_2}\ \mathrm{diag}(\mathbf{\Sigma}_2^{-\frac{1}{2}}) \\ \vdots \\ \sqrt{w_K}\ \mathrm{diag}(\mathbf{\Sigma}_K^{-\frac{1}{2}}) \end{bmatrix} \tag{4.4}$$

L'objectif de la normalisation est d'avoir des caractéristiques dans un rang spécifique de valeurs, afin d'éviter que le résultat du noyau linéaire (multiplication entre deux vecteur) soit « dominé » par certaines caractéristiques (celles aux rangs de valeurs larges) au détriment des autres (celles aux valeurs plus petites).

La figure 4.2 illustre le processus de création de super vecteurs. Le modèle du monde λ_{UBM} est adapté par une procédure MAP (*Maximum a Posteriori*). Seules les moyennes des composantes Gaussiennes sont adaptées (équations 4.5 à 4.9) [Reynolds et al., 2000]. Pour calculer la matrice $\mathbf{R}_{\mathrm{GSL}}$, on utilise les paramètres du modèle λ_{UBM}. Le paramètre r (et donc α_r, équation 4.6) est utilisé pour contrôler l'adaptation du modèle par rapport aux vecteurs en entrée. Plus la valeur de r est petite, plus le modèle adapté diffère du modèle initial et se rapproche des données utilisées en adaptation. La figure 4.3 montre ce processus en partant des vecteurs acoustiques du signal pour arriver aux super vecteurs utilisés en entrée d'un SVM.

$$\hat{\mu}_k = \alpha_k \tilde{\mathbf{x}}_k + (1 - \alpha_k)\mu_k \tag{4.5}$$

où :

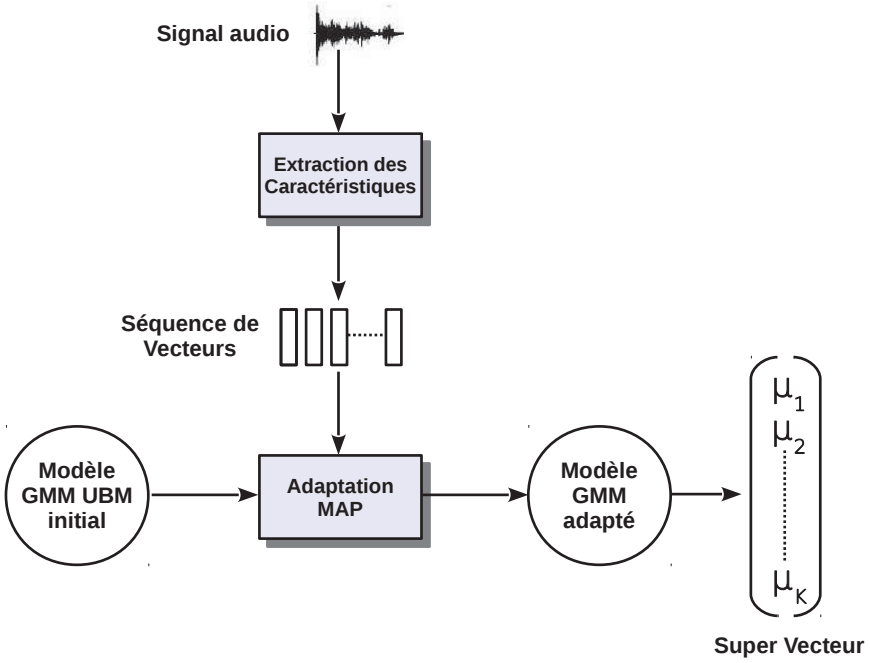

Figure 4.2: Noyau SVM-GSL : transformation d'une séquence de vecteurs acoustiques en super vecteur

$$\alpha_k = \frac{n_k}{n_k + r} \tag{4.6}$$

$$\tilde{\mathbf{x}}_k = \frac{1}{n_k} \sum_{m=1}^{M} Pr(k|\mathbf{x}_m)\mathbf{x}_m \tag{4.7}$$

$$n_k = \sum_{m=1}^{M} Pr(k|\mathbf{x}_m) \tag{4.8}$$

$$Pr(k|\mathbf{x}_m) = \frac{w_k \mathcal{N}(\mathbf{x}_m|\mu_k, \mathbf{\Sigma}_k)}{\sum_{n=1}^{K} w_n \mathcal{N}(\mathbf{x}_m|\mu_n, \mathbf{\Sigma}_n)} \tag{4.9}$$

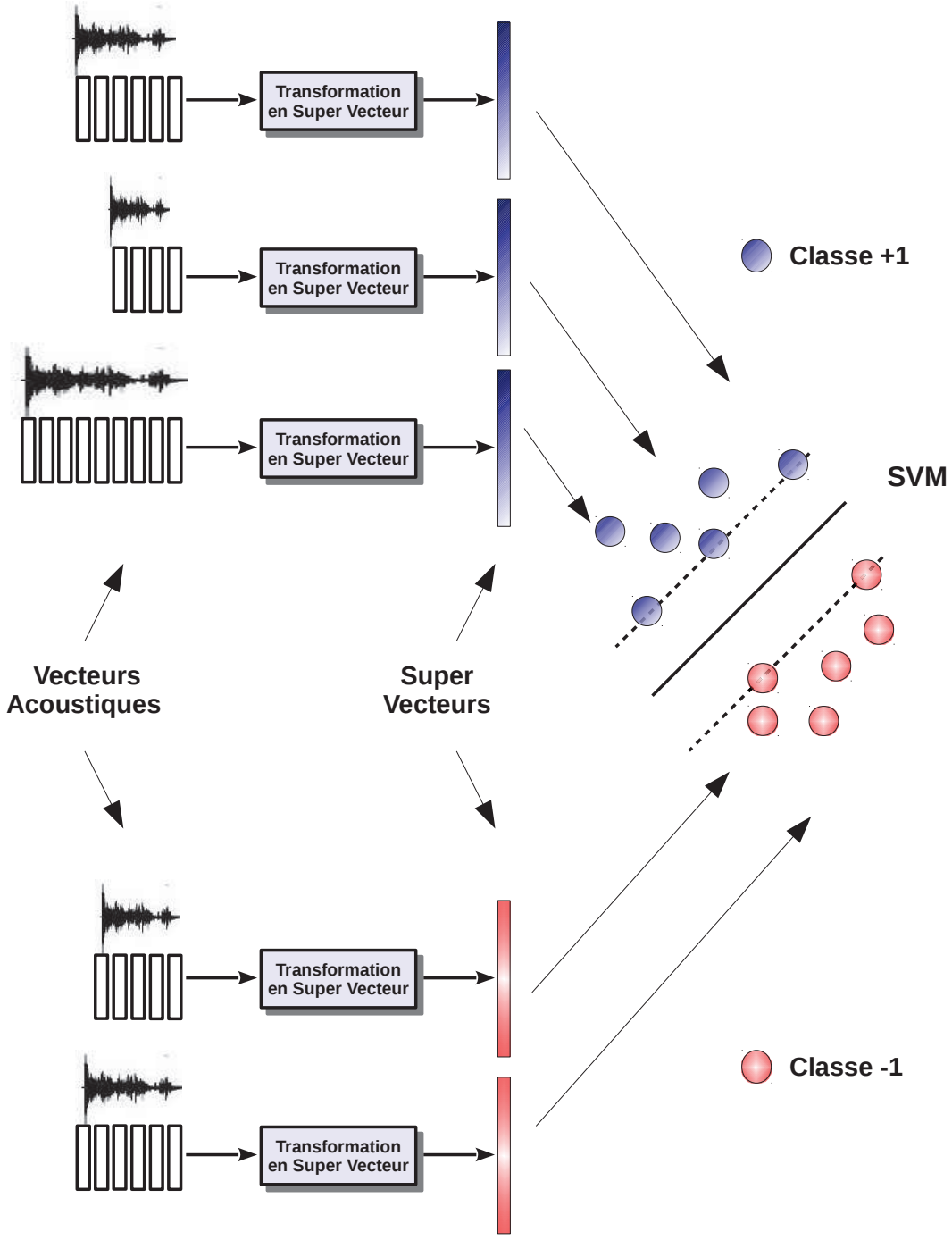

Figure 4.3: Des vecteurs acoustiques au super vecteurs SVM

4.3 Bases de données

4.3.1 Base de données de l'ESIGETEL

La base de données utilisée dans les expérimentations a été l'objet de plusieurs évolutions, notamment durant la première moitié de la thèse. De nouvelles classes ont été ajoutées, d'autres écartées en raison du nombre réduit d'enregistrements, et d'autres encore enrichies. Cependant, une version figée est utilisée depuis début 2012 jusqu'à la fin de la thèse. Elle est ici utilisée pour les expérimentations de comparaison entre les GMMs et SVM-GSL, ainsi que pour les expérimentations du chapitre 5.

La base (tableau 4.1) contient 18 classes de sons de la vie quotidienne. Elle inclut des sons humains (respirations, toux, cris, etc.) et autres sons non humains (moteurs, bris de glace, etc.). Les différents enregistrements qui la composent proviennent de plusieurs sources et sont enregistrés (exceptée la classe *DoorOpening* dont tous les enregistrements ont été réalisés en utilisant une seule porte, et les deux classes *ElectricalShaver* et *HairDryer* dont les fichiers ont été découpés depuis des enregistrements longs de plusieurs minutes) dans des conditions acoustiques qui peuvent être très différentes (matériel et/ou environnement différents, différentes personnes pour les sons humains, etc.). Enfin, la fréquence d'échantillonage des fichiers audio est de 16KHz.

4.3.2 Corpus du projet SWEET-HOME

Une base de données multimodale a été enregistrée dans le cadre du projet SWEET-HOME. L'objectif de ces enregistrements est de caractériser les comportements et les événements acoustiques des sujets, suivant des scénarios de vie quotidienne prédéfinis. Les scénarios ont eu lieu dans l'appartement intelligent, DOMUS, de l'équipe Multicom (Laboratoire LIG, Grenoble). Les données sont utilisées pour l'évaluation des méthodes mises en œuvre par les différentes équipes de recherche participant au projet. 21 sujets (S01 à S21) ont pris part aux enregistrements réalisés entre octobre et novembre 2010.

Tableau 4.1: Classes de sons de la base de données de l'ESIGETEL

Classe de sons	**# de fichiers**	**Durée totale (sec).**	**Conditions d'enreg. variables**
Breathing	50	106.44	Oui
Cough	62	181.69	Oui
Dishes	98	303.77	Oui
DoorClapping	114	62.70	Oui
DoorOpening	21	138.94	**Non**
ElectricalShaver	62	420.33	**Non**
FemaleCry	36	268.19	Oui
FemaleScream	70	216.83	Oui
GlassBreaking	101	99.52	Oui
HairDryer	40	224.86	**Non**
HandsClapping	54	218.65	Oui
Keys	36	166.34	Oui
Laughter	49	272.65	Oui
MaleScream	87	202.11	Oui
Paper	63	330.66	Oui
Sneeze	32	51.67	Oui
Water	54	484.72	Oui
Yawn	20	95.87	Oui
Total	**1049**	**3845.94**	-

4.3.2.1 Plan de l'appartement DOMUS

La figure 4.4 illustre le plan de l'appartement DOMUS. On y retrouve un coin repas cuisine, une chambre, une salle de bain et un bureau. Afin de capter toute l'information sonore ayant lieu aux différents endroits de l'appartement, sept microphones y sont disposés comme le montre la figure 4.5.

4.3.2.2 Scénarios enregistrés

Le corpus a été enregistré en deux phases. La première phase étant la plus importante (car de loin plus riche en activités), c'est elle que nous avons utilisée pour les expérimentations. Elle se compose de plusieurs sous-scénarios que, excepté le premier, les participants peuvent suivre dans n'importe quel ordre. Le tableau 4.2 récapitule les activités des différents sous-scénarios.

Figure 4.4: Plan de l'appartement DOMUS

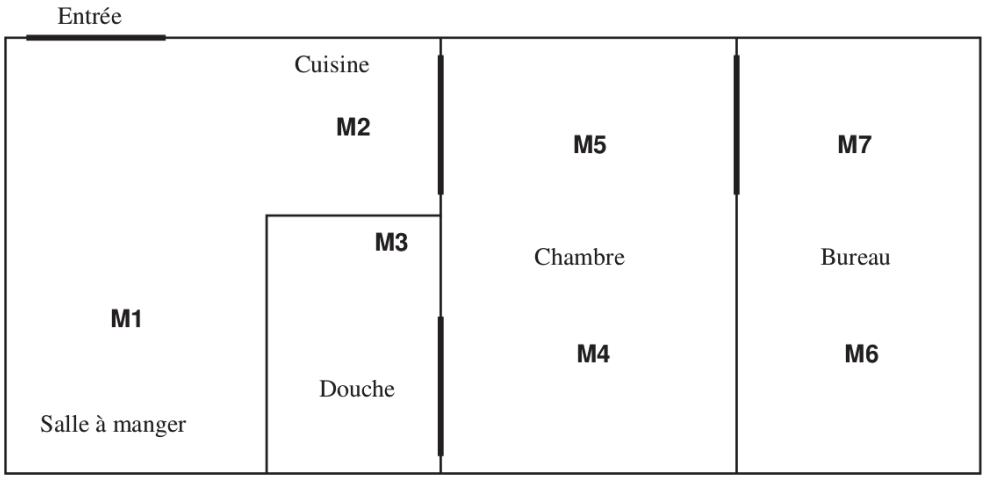

Figure 4.5: Position des microphones dans l'appartement DOMUS

Tableau 4.2: Scénarios d'enregistrement (phase 1) qui composent le corpus du projet SWEET-HOME

Activité	**Durée**	**Consignes**
Début		
Entrer dans l'appartement et fermer la porte Actionner la gâche électrique Faire le tour des pièces en fermant les portes derrière soit Poser le manteau ou la veste Revenir à la cuisine	~30 sec.	vous pouvez allumer la lumière s'il fait trop sombre
Habillage		
Aller dans la chambre et fermer la porte derrière soi Sortir les vêtements du tiroir (tee-shirt et pantalon) Enfiler tee-shirt et pantalon par dessus ses vêtements Ouvrir les volets et les fenêtres de la chambre	~3 min.	vous pouvez allumer la lumière s'il fait trop sombre
Hygiène		
Aller dans la salle de bain, éclairer la lumière Se laver les mains, s'essuyer, éteindre la lumière	~3 min.	
Manger		
Aller à la cuisine Ouvrir le volet de la cuisine Préparer les ustensiles de cuisine pour faire une boisson chaude Préparer une boisson chaude en utilisant une casserole Prendre une tasse du placard du pain, confitures, fruits, etc S'asseoir et manger sur la table de la cuisine Débarrasser la table Nettoyer la table Faire la vaisselle	~15 min.	 thé, café, tisane, chocolat
Non disponibilité lors d'un appel extérieur		
Recevoir un appel telephonique pendant la vaisselle et ne pas répondre		
Faire la vaisselle		
Essuyer la vaisselle Ranger la vaisselle		
Hygiène - se laver les dents après le repas		
Aller dans la salle de bain, allumer la lumière Tousser Se racler la gorge Se brosser les dents Cracher de l'eau Se rincer la bouche Boire de l'eau Claquer les mains Éteindre la lumière avant de sortir	~3 min.	
Sieste		
Aller dans la chambre Fermer les portes Fermer les volets et les fenêtres de la chambre Éteindre tous les lumières en restant dans la chambre Se coucher Dormir au moins 1 minute	~10 min.	

Suite du tableau à la page suivante

Tableau 4.2 – *(Suite)*

Activité	**Durée**	**Consignes**
Se lever et aller à la cuisine pour se servir un verre d'eau Se recoucher Dormir au moins 2 min.		
Se lever		
Se lever et allumer la lumière Ouvrir les fenêtres et les volets Faire le lit	~10 min.	
Ménage		
Prendre l'aspirateur dans le placard de la cuisine Passer l'aspirateur dans la cuisine et la chambre Aller dans la cuisine Sortir les assiettes une à une et les poser sur la table Prendre tous les couverts et les poser sur la table Prendre les produits de nettoyage dans le placard de la cuisine Laver l'évier de la cuisine Fermer les fenêtres Ranger la vaisselle dans le placard	~10 min.	
Détente		
Aller dans le bureau et chercher un livre sur l'étagère Lire le livre dans le bureau Rester dans le bureau, mettre en route la radio puis écouter Aller à la cuisine pour prendre une boisson froide Continuer à lire dans le bureau	~10 min.	verre d'eau
Appel téléphonique		
Recevoir un appel, étendre la télé/radio et avoir une conversation téléphonique	~5 min.	lire le texte 1, respecter une pause entre chaque phrase
Détente		
Remettre en route la radio Écouter la radio Éteindre la radio	~5 min.	
Sortie		
Fermer tous les fenêtres Éteindre toutes les lumières Prendre son manteau ou sa veste Sortir de l'appartement et fermer à clé	~3 min.	
Retour de sortie		
Ouvrir avec la clef Entrer dans l'appartement Poser son manteau ou son pull dans la chambre Ranger les courses dans la cuisine	~3 min.	
Détente/communication		
Aller dans le bureau Allumer l'ordinateur Utiliser l'ordinateur pour consulter le web Éteindre l'ordinateur	~10 min.	
Communication téléphonique		
Effectuer un appel téléphonique	~5 min.	lire le texte 2, respecter une pause entre chaque phrase

Suite du tableau à la page suivante

Tableau 4.2 – *(Suite)*

Activité	Durée	Consignes
Remplir le sac de courses		
Aller à la cuisine Boire un verre d'eau Remettre les courses dans le sac		opération inverse de la précédente
Déshabillage		
Aller dans la chambre Allumer le radiateur Se déshabiller	~3 min.	
Hygiène		
Prendre une douche Éteindre la lumière Sortir de la salle de bain en fermant la porte	~5 min.	faire couler la douche et simuler aller dans la chambre
Habillage		
S'habiller pour la nuit	~3 min.	
Vérification des issues		
Vérifier les volets, les fenêtres Verrouiller la porte d'entrée Éteindre le radiateur	~3 min.	
Se coucher pour la nuit		
Se coucher Lire quelques pages d'un livre Éteindre la lumière Dormir	~10 min.	

4.3.2.3 Indexation du corpus

Les différentes activités des participants ont été la source d'un nombre considérable d'événements acoustiques. Reconnaître tous les sons à l'oreille a donc été totalement impossible. Pour nous aider dans cette tâche, cruciale pour les évaluations, nous nous sommes servis des enregistrements vidéo des scénarios. Pour avoir une vue globale de l'appartement, nous avons utilisé des vidéos sous forme de mosaïque, créées au laboratoire Imag à Grenoble (figure 4.6).

Les vidéos n'étaient, toutefois, accompagnées que d'un seul canal audio ; les activités ayant lieu loin du canal retenu étaient à peine audibles. Pour avoir tous les événements acoustiques d'un scénario dans un seul canal, nous avons créé un canal « fusionné », contenant l'information acoustique des sept canaux. Les échantillons du canal fusionné sont la somme pondérée des échantillons des autres canaux. Les coefficients de pondération sont les RSBs (rapport signal sur bruit) normalisés des différents canaux. Le RSB est calculé sur une fenêtre de 2048 échantillons (128 ms).

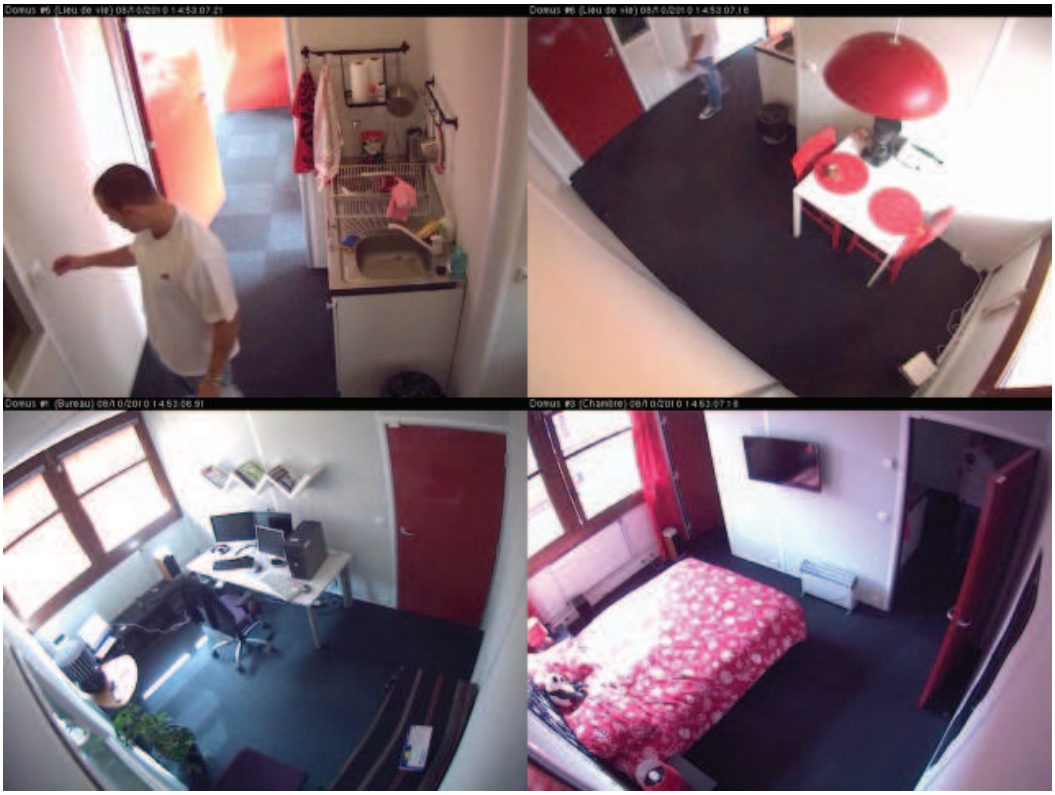

Figure 4.6: Mosaïque de 4 caméras de l'appartement DOMUS

Pour annoter les enregistrements, nous avons utilisé Transcriber [1] [Barras et al., 2001], un outil notamment utilisé pour transcrire la parole. Un algorithme de détection est utilisé pour détecter les activités acoustiques dans le canal fusionné. Un fichier compatible avec Transcriber est généré. Chacune des activités détectées est initialement étiquetée « inconnu » (*Unknown*). Le rôle de l'annotateur est de remplacer cette étiquette par le son correspondant (figure 4.7) en écoutant le canal fusionné et en regardant, si besoin, la vidéo. Pour la parole, l'annotateur en effectue également la transcription.

L'annotateur doit également corriger les erreurs commises par l'algorithme de détection. En effet, certains événements ne sont pas détectés, d'autres découpés en deux ou plusieurs morceaux, d'autres encore, détectés avec un intervalle de silence avant ou après l'événement, qui doit être éliminé (l'intervalle de silence).

1. http ://trans.sourceforge.net

Figure 4.7: Capture d'écran montrant un exemple d'annotation avec Transcriber

L'annotation fut une tâche particulièrement longue et difficile. Ne connaissant pas toutes les classes de sons que contiennent les enregistrements, il nous ne restait plus que à les découvrir au fur et à mesure de l'annotation. La liste des classes évoluait d'un sujet à l'autre et il nous fallait revenir plusieurs fois sur des parties déjà annotées. Enfin, en dépit de l'existence de sept canaux et des vidéos, certains sons nous paraissaient méconnaissables. Un nombre important de sons (pour la plupart très courts) ont ainsi conservé leur étiquette initiale, *Unknown*.

4.4 Résultats expérimentaux

4.4.1 Expérimentations avec la base de données de l'ESIGETEL

La base de données est subdivisée en trois parties. La première partie est utilisée pour créer les modèles GMM des différentes classes ainsi que le modèle UBM utilisé dans le noyau SVM-GSL. La seconde partie est utilisée pour créer les super vecteurs, afin d'éviter d'utiliser les mêmes données que celles du modèle UBM. La troisième partie est utilisée pour tester les deux méthodes.

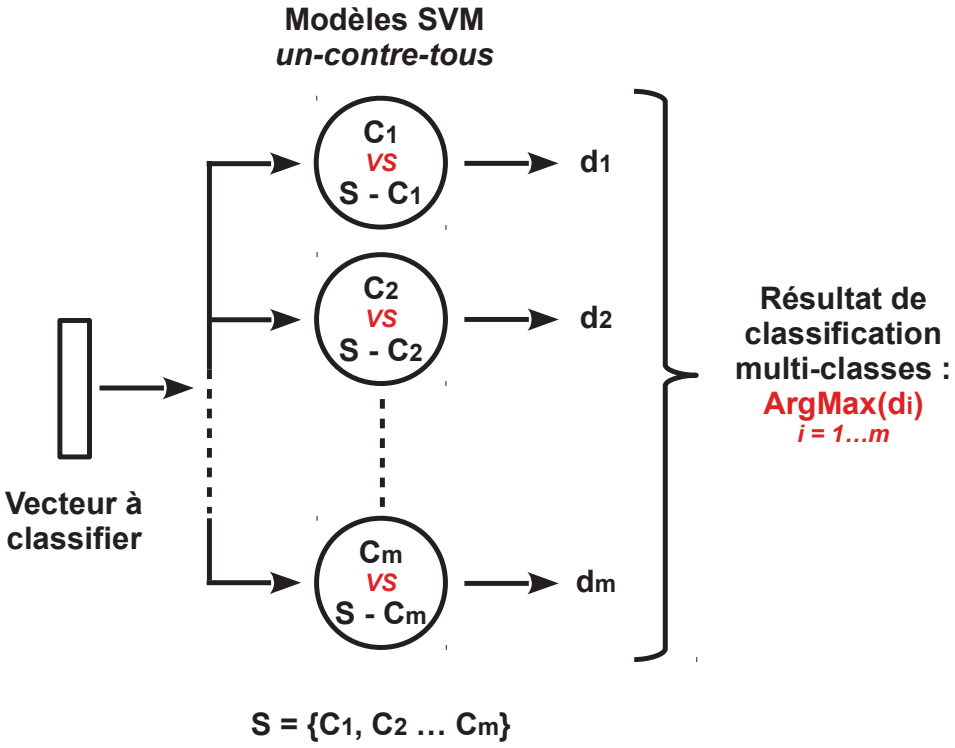

Figure 4.8: Classification SVM multi-classes en utilisant la stratégie un-contre-tous

Les SVMs sont une méthode de classification binaire, c'est-à-dire qu'un classifieur est entraîné pour discriminer deux classes. Bien qu'il existe des variantes de SVMs pouvant trouver une marge entre plusieurs classes [Vapnik, 1998] [Weston and Watkins, 1999], et de ce fait, effectuer une classification multi-classes avec un seul modèle, l'utilisation de plusieurs SVMs binaires restent la plus communément employée. Pour effectuer une classification multi-classes en se servant de SVMs binaires, on peut employer plusieurs stratégies.

La stratégie la plus simple consiste à créer, pour chaque classe de données, un modèle SVM qui la sépare de toutes les autres classes [Schölkopf et al., 1995]. Pour chaque modèle SVM_i $(i = 1, \ldots, m)$, tel que m est le nombre de classes, les données de la classes i sont étiquetées $+1$ et celles des classes restantes -1. Un vecteur dont la classe est inconnue est présenté en entrée de chacun des m modèles obtenus. Le vecteur est ensuite attribué à la classe dont le modèle (qui la sépare des autres classes) retourne la plus grande valeur d_i. La figure 4.8 illustre ce principe, aussi baptisé **un-contre-tous** (*One-Against-All*).

Cette stratégie, plutôt facile à mettre en œuvre, peut présenter un problème si la quantité de données utilisées est importante. En effet, pour chaque modèle SVM créé, les données de toutes les classes sont utilisées (avec un étiquetage différent, bien entendu), ce qui peut rendre le processus d'apprentissage et celui d'évaluation très lents. D'autre part, les valeurs d_i à comparer représentent des distances retournées par des modèles SVM différents et ne sont, de ce de fait, pas tout à fait comparables. Cette stratégie n'a pas été retenue par les évaluations présentées dans cette thèse.

Un alternative à la stratégie un-contre-tous et celle du **un-contre-un** (*One-Against-One*). Comme l'indique son nom, cette stratégie consiste à créer des modèles SVM binaires pour séparer chaque couple de deux classes distinctes. Pour m classes, le nombre de modèles créés est donc de $m(m-1)/2$.

Pour effectuer une classification multi-classes en utilisant les modèles binaires, une des solutions consiste à utiliser un système de compétition sous forme d'arbre binaire (figure 4.9). L'avantage de cette approche est d'éviter de faire toutes les évaluations possibles (du nombre de $m(m-1)/2$) pour décider de la classe du vecteur en entrée. Son inconvénient majeur vient du fait que deux ou plusieurs arbres différents peuvent donner des résultats très différents. Autrement dit, le résultat final peut être fonction du choix des classes mises en compétition à chaque niveau. Si, pour un problème de classification à huit classes, par exemple, l'évaluation d'un vecteur en entrée avec les sept modèles $\text{SVM}_{1,j}$ $(j = 2, \ldots, m)$ s'avère à l'avantage de la classe C_1, mise en compétition contre les classes C_3 à C_8, et à celui de la classe C_2 d'après la sortie du modèle $\text{SVM}_{1,2}$, la classe C_1 sera écartée au premier niveau selon l'arbre de la figure 4.9. Si, par contre, en modifiant légèrement la structure de l'arbre, la classe C_2, l'unique classe qui « battrais » C_1, est mise au premier niveau contre la classe C_3, et que le modèle $\text{SVM}_{2,3}$ attribue le vecteur à la classe C_3, la classe C_1 sera au final élue comme la meilleure classe.

Pour contourner ce problème, une autre stratégie est utilisée. Le vecteur en entrée est évalué avec tous les modèles $m(m-1)/2$ et est par la suite attribué à la classe qui aura reçu le plus de votes, c'est-à-dire, plus de victoires lors des évaluations binaires. Cette approche est retenue pour les expérimentations effectuées sur le corpus du projet SWEET-HOME (section 4.4.2).

Pour les expérimentations avec la base de données de l'ESIGETEL, en revanche, nous utilisons une variante ce cette stratégie : un vecteur est attribué à la classe qui aura $m-1$ votes (qui battra toutes les autres classes). Si une telle classe n'existe pas, ou si elle existe mais ne correspond, finalement, pas à la vraie classe du vecteur, la classification est alors fausse.

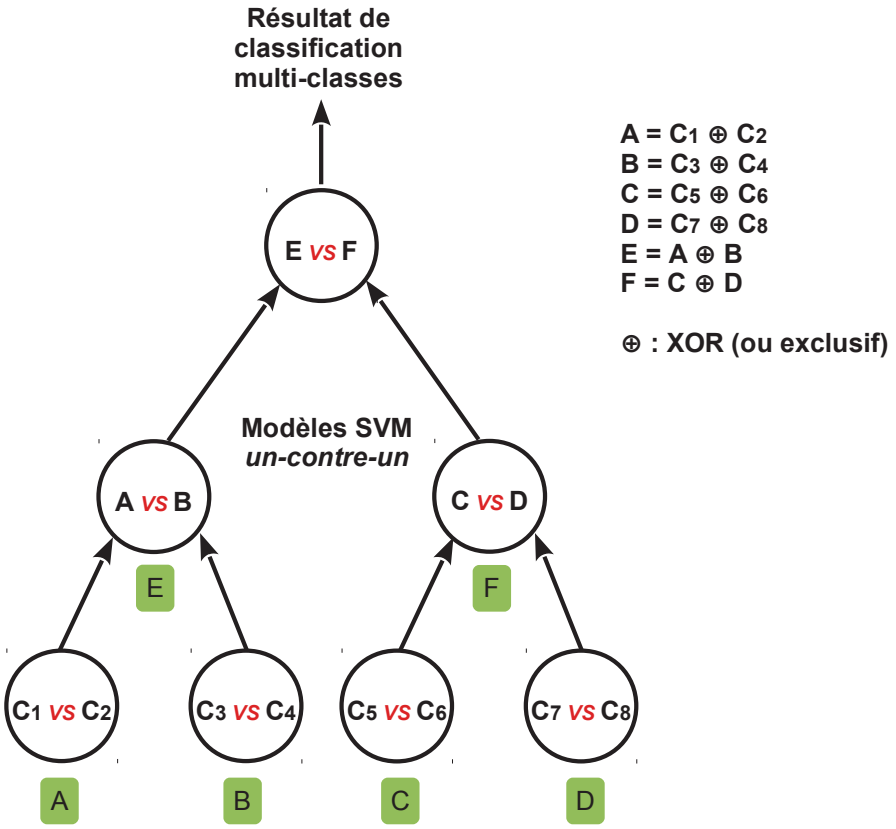

Figure 4.9: Classification SVM multi-classes en utilisant des modèles SVM dans une structure d'arbre binaire

D'autres stratégies de classification multi-classes basées sur l'utilisation de plusieurs modèles SVM un-contre-un existent également [Hsu and Lin, 2002] [Platt et al., 1999] [Duan and Keerthi, 2005]. On peut également mentionner les méthodes utilisant des SVMs dont les sorties se présentent sous forme de probabilité [Hastie and Tibshirani, 1998] (voir les travaux de Platt pour ce type de SVMs [Platt, 1999]).

Pour les expérimentations présentées ici, des vecteurs acoustiques de 16 coefficients *MFCC* sont utilisés. Ils sont extraits depuis des fenêtres de signal de 16ms, avec 50% (8ms) de recouvrement. Pour SVM-GSL, trois modèles UBM, de 256, 512 et 1024 composantes respectivement sont testés. Cela donne des super vecteurs de $256 \times 16 = 4096$, $512 \times 16 = 8192$ et $1024 \times 16 = 16384$ dimensions respectivement. Le nombre de composantes pour les modèles GMM varie d'une classe à l'autre. Il est initialement fixé à 50. Pour certaines classes dont le nombre de vecteurs

Tableau 4.3: Performances des GMMs et de SVM-GSL

Méthode	**Taux de Reconnaissance (%)**
GMM	71.1
SVM-GSL (taille UBM = 256)	74.0
SVM-GSL (taille UBM = 512)	75.4
SVM-GSL (taille UBM = 1024)	75.1

est particulièrement petit (claquements de porte, bris de glace), ce nombre était très élevé et l'algorithme d'apprentissage ne convergeait pas. Il a dû donc (le nombre de composantes) être réduit progressivement jusqu'à obtenir un nombre approprié pour chaque classe. Le nombre final de composantes varie de 25 à 50, selon la classe.

Pour créer les différents modèles GMM (y compris le modèle UBM), nous avons utilisé la bibliothèque Alize [Bonastre et al., 2005]. Pour les SVMs nous avons utilisé la bibliothèque LibSVM [Chang and Lin, 2011].

Les résultats obtenus (tableau 4.3) montrent une amélioration des performances apportée par SVM-GSL par rapport aux GMMs. Les résultats pour les différentes tailles du modèle UBM sont assez comparables. Cependant, au-delà de 1024 composantes, nous avons constaté une baisse des performances.

Pour une analyse plus détaillée des résultats, observons la matrice de confusion des classes de sons obtenue avec SVM-GSL pour un UBM de 1024 composantes. De prime abord, on remarque que les trois classes enregistrées dans les mêmes conditions acoustiques (*DoorOpening*, *Electrical-Shaver* et *HairDryer*) sont très bien reconnues. Par ailleurs, certaines classes dont les conditions d'enregistrement varient (*DoorClapping*, *Keys*, *Sneeze*, etc.) présentent tout de même de bons taux de reconnaissance. Pour les classes aux taux de confusion plus ou moins élevés, on observe un phénomène intéressant : chacune de ces classes est majoritairement confondue avec une seule autre classe. Par exemple, la classe *Breathing* est confondue avec la classe *DoorOpening*, *Dishes* avec *HandsClapping*, *Laughter* avec *Cough*, etc. Pour le couple de classes *Dishes–HandsClapping*, par exemple, l'observation de l'enveloppe d'amplitude (figures 4.10 et 4.11) permet de voir les similitudes qui existent entre leurs enregistrements respectifs. Pour les deux classes, on peut en effet se rendre compte de l'existence de plusieurs maxima espacés de fenêtres de silence.

Tableau 4.4: Matrice de confusion pour le noyau SVM-GSL (taille du modèle UBM = 512). L'intensité du vert ou du rouge indique le taux de reconnaissance ou celui d'erreur pour les cellules concernées, respectivement

	Breath	Cough	Dishes	DClapp	DOpen	EShaver	FemCry	FScrm	GBreak	HDryer	HClap	Keys	Laughter	MScrm	Paper	Sneeze	Water	Yawn
Breath	9/17	1/17	0/17	0/17	7/17	0/17	0/17	0/17	0/17	0/17	0/17	0/17	0/17	0/17	0/17	0/17	0/17	0/17
Cough	0/21	15/21	0/21	0/21	0/21	0/21	0/21	0/21	1/21	0/21	0/21	0/21	3/21	0/21	1/21	1/21	0/21	0/21
Dishes	1/33	1/33	14/33	0/33	1/33	0/33	0/33	3/33	0/33	0/33	12/33	0/33	0/33	0/33	1/33	0/33	0/33	0/33
DClapp	0/38	1/38	0/38	36/38	0/38	0/38	0/38	0/38	0/38	0/38	0/38	0/38	0/38	0/38	0/38	0/38	1/38	0/38
DOpen	0/7	0/7	0/7	0/7	7/7	0/7	0/7	0/7	0/7	0/7	0/7	0/7	0/7	0/7	0/7	0/7	0/7	0/7
EShaver	1/21	0/21	0/21	0/21	0/21	20/21	0/21	0/21	0/21	0/21	0/21	0/21	0/21	0/21	0/21	0/21	0/21	0/21
FemCry	0/12	2/12	0/12	0/12	0/12	0/12	8/12	1/12	0/12	0/12	0/12	0/12	0/12	0/12	0/12	1/12	0/12	0/12
FScrm	0/24	1/24	2/24	0/24	0/24	0/24	0/24	19/24	0/24	0/24	0/24	0/24	0/24	2/24	0/24	0/24	0/24	0/24
GBreak	0/34	4/34	0/34	5/34	0/34	0/34	0/34	0/34	25/34	0/34	0/34	0/34	0/34	0/34	0/34	0/34	0/34	0/34
HDryer	0/14	0/14	0/14	0/14	0/14	0/14	0/14	0/14	0/14	14/14	0/14	0/14	0/14	0/14	0/14	0/14	0/14	0/14
HClap	0/18	1/18	0/18	0/18	0/18	0/18	0/18	0/18	0/18	0/18	17/18	0/18	0/18	0/18	0/18	0/18	0/18	0/18
Keys	0/12	0/12	0/12	0/12	0/12	0/12	0/12	0/12	1/12	0/12	1/12	10/12	0/12	0/12	0/12	0/12	0/12	0/12
Laughter	0/17	12/17	1/17	0/17	0/17	0/17	0/17	0/17	0/17	0/17	0/17	0/17	2/17	1/17	0/17	1/17	0/17	0/17
MScrm	0/29	2/29	0/29	0/29	0/29	0/29	0/29	0/29	0/29	0/29	0/29	0/29	0/29	27/29	0/29	0/29	0/29	0/29
Paper	0/21	1/21	0/21	0/21	0/21	0/21	0/21	0/21	1/21	0/21	0/21	0/21	0/21	0/21	19/21	0/21	0/21	0/21
Sneeze	0/11	1/11	0/11	0/11	0/11	0/11	0/11	0/11	0/11	0/11	0/11	0/11	0/11	0/11	0/11	10/11	0/11	0/11
Water	0/18	2/18	0/18	4/18	0/18	0/18	0/18	0/18	0/18	0/18	0/18	0/18	0/18	0/18	0/18	0/18	12/18	0/18
Yawn	3/7	0/7	0/7	0/7	0/7	0/7	0/7	0/7	1/7	0/7	0/7	0/7	0/7	0/7	0/7	0/7	0/7	3/7

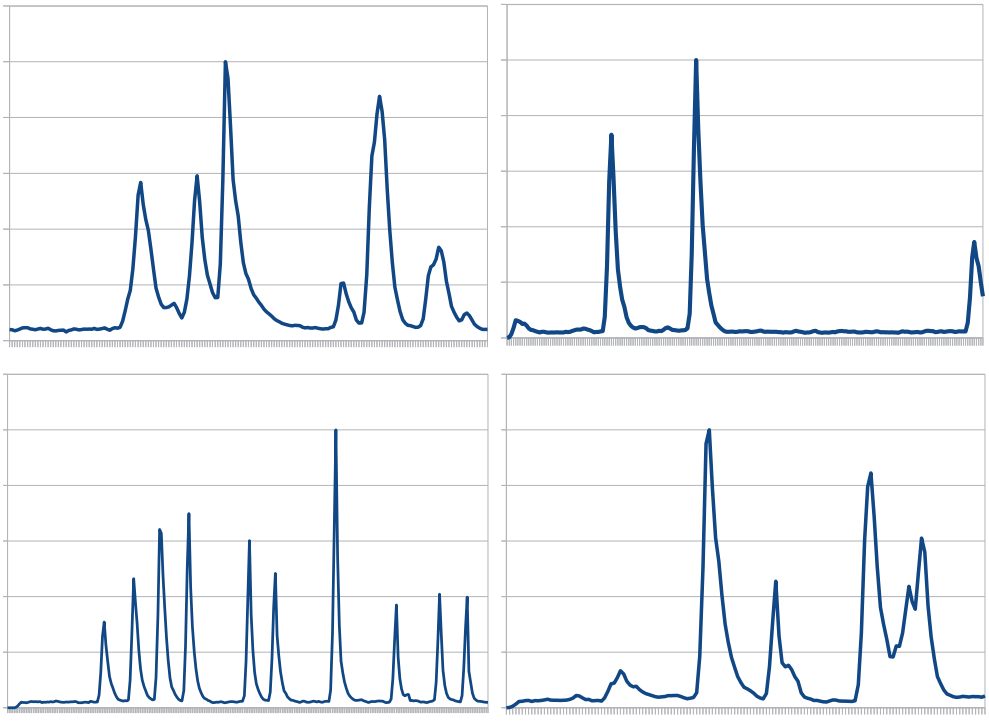

Figure 4.10: Enveloppe d'amplitude d'enregistrements de bruits d'assiettes

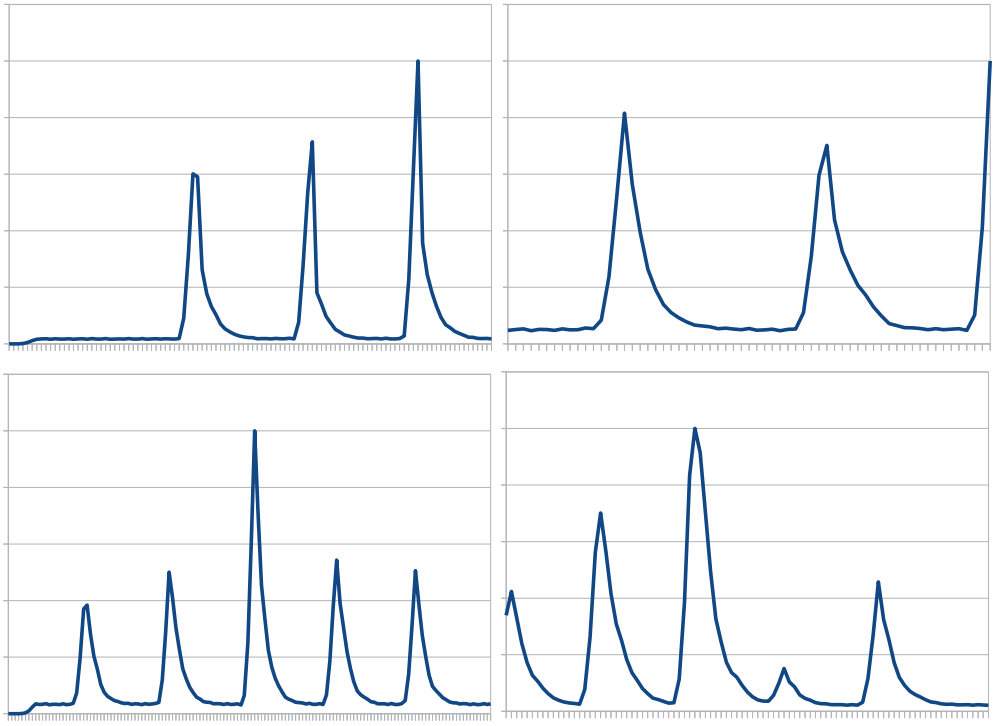

Figure 4.11: Enveloppe d'amplitude d'enregistrements de claquements de mains

4.4.2 Expérimentations avec le Corpus du projet SWEET-HOME

Le but de ces expérimentations est de tester les composantes d'un système typique de reconnaissance des événements acoustiques, ainsi que celles d'un système de reconnaissance automatique de la parole (partie réalisée par le laboratoire LIG à Grenoble) en conditions réalistes. L'architecture du système est illustrée à la figure 4.12.

Comme le montre la figure, les sept canaux installés dans l'appartement sont utilisés. Le système doit détecter les événements acoustiques sur tous les canaux excités et les confier au module de séparation entre la parole et les autres sons de la vie courante. Il aurait été possible de considérer la parole comme toute autre classe de sons et se passer complètement de cette couche. Le but de cette étape supplémentaire est de reconnaître les signaux contenant de la parole au plus vite et de les transmettre au module de RAP dont les traitements effectués peuvent être longs. Un autre argument en faveur de cette architecture est le fait de minimiser les erreurs de classification et, de ce fait, détecter le plus de phrases possibles. Un système de classification binaire, parole *versus* autres sons,

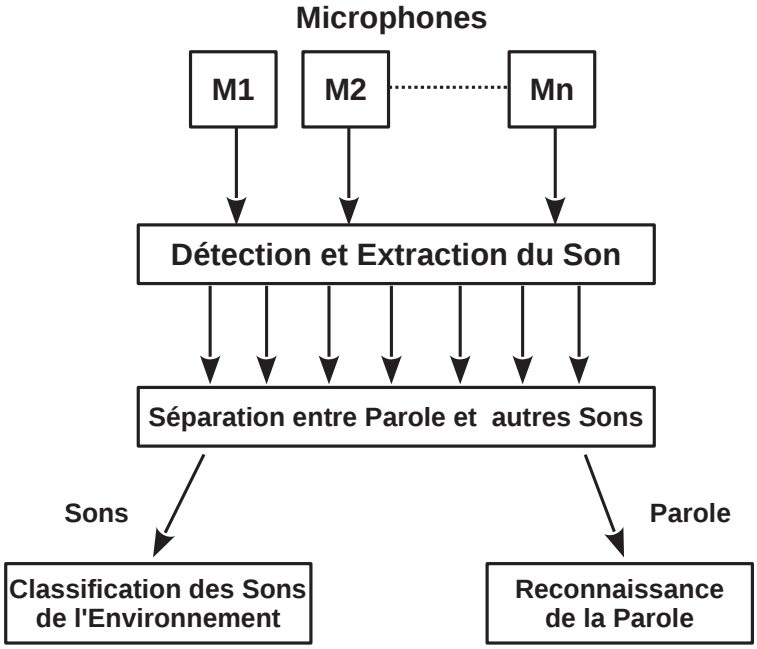

Figure 4.12: Architecture du système de reconnaissance du son de SWEET-HOME

est partiellement plus performant qu'un système qui considérerait la parole comme une classe de sons parmi un nombre important de classes.

Au vu des résultats obtenus sur la base de données de l'ESIGETEL, un système basé sur le noyau SVM-GSL est retenu pour toutes les expérimentations sur le corpus de SWEET-HOME. Il est utilisé pour la séparation entre la parole et les autres sons ainsi que pour la classification des sons de l'environnement.

Des scénarios de 4 sujets différents (S01, S03, S07 et S09 pour une durée de 88, 50, 72 et 63 minutes respectivement) sont utilisés dans ces les expérimentations. En raison d'une annotation incomplète, le sujet S13 n'est pas utilisé pour évaluer la détection et le reconnaissance des sons de l'environnement. Il est en revanche utilisé pour évaluer la séparation entre la parole et les autres sons, ainsi que pour la reconnaissance de la parole.

Les données utilisées en apprentissage proviennent de trois autres sujets (S02, S08 et S10) du corpus. Tous les sujets utilisés en apprentissage et en évaluation ont été annotés à l'ESIGETEL, pour la plupart par une seule personne (l'auteur de cette thèse). Certains sujets ont été annotés

partiellement ou complètement par un second annotateur. Ces sujets ont fait l'objet d'un second passage que effectué par l'auteur. Ils ont été corrigés pour correspondre à l'annotation réalisée pour les autres sujets. Enfin, tous les sujets utilisés dans ces expérimentations ont été rigoureusement vérifiés (et corrigés en cas de besoin) pour s'assurer que le même vocabulaire est utilisé pour se référer aux classes de sons identiques. Étant annotés sur une période de plusieurs mois, certaines étiquettes qui devraient se référer aux mêmes classes de sons ont, en effet, dû subir un changement plus ou moins important, sans parler des erreurs humaines. Des outils standard de recherche et d'édition de texte (grep, sed etc.) ont été utilisés pour détecter et corriger ces étiquettes.

4.4.2.1 Détection du son

La détection du son ne fait pas partie de ce travail mais ses performances affectent toutes les composantes suivantes du système. Nous allons donc donner une brève description de l'approche utilisée et en rapporter les résultats.

L'algorithme de détection utilisé [Rougui et al., 2009] est basé sur la **transformée en ondelettes discrète** (DWT ou *Discrete Wavelet Transform*). Il calcule l'énergie des trois coefficients de haute fréquence de la transformée en ondelettes et adapte dynamiquement le seuil de détection, en fonction du RSB de la fenêtre en cours. L'estimation du RSB de l'événement détecté est très importante pour les modules suivants. L'algorithme estime le RSB d'un événement acoustique en partant de l'hypothèse que le bruit dans l'événement est similaire au bruit des quelques fenêtres précédant l'événement.

Pour évaluer les performances de ce module, nous considérons qu'un événement est correctement détecté si l'algorithme le détecte sur au moins un canal, avec un taux de recouvrement τ entre la détection automatique et la détection de référence. Autrement dit, la détection automatique doit couvrir au moins $\tau\%$ de la détection de référence. Le tableau 4.5 montre les résultats de détection pour différents taux de recouvrement. Un faible taux de recouvrement (20%) permet de détecter plus d'événements mais le signal utile peut alors être très court. Avec un taux de recouvrement de 100% beaucoup d'événements utiles sont rejetés. Pour la méthode de reconnaissance qu'on utilise, l'information quant au début et à la fin d'un événement acoustique n'est pas prise en considération ; une partie du signal peut être utilisée. La figure 4.13 montre quelques exemples de détections automatiques et les erreurs qui peuvent êtres commises par l'algorithme de détection. Pour la suite des évaluations, nous retenons un taux de 50% pour trouver un compromis entre le nombre d'événements détectés et le recouvrement. Le tableau 4.6 montre les résultats moyens de détection

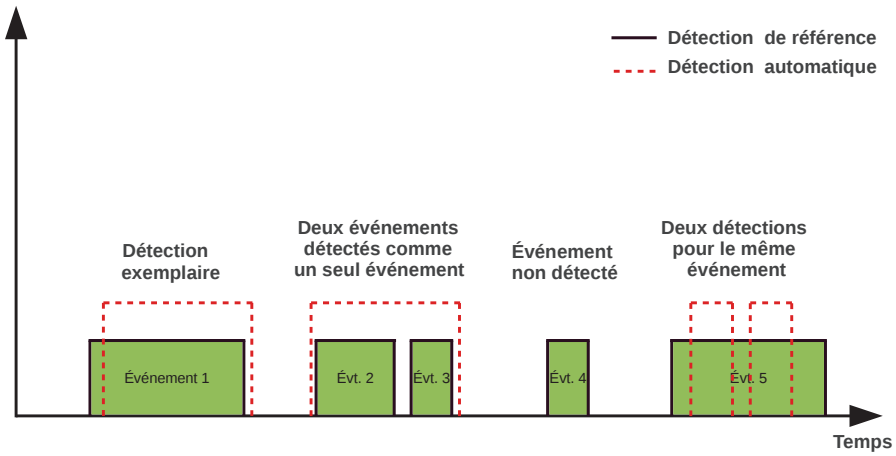

Figure 4.13: Exemple montrant un détection automatique exemplaire (par rapport à la référence) ainsi que des erreurs de détections possibles

Tableau 4.5: Sensibilité de détection pour différentes valeurs du taux de recouvrement τ.

τ (%)	20	50	80	100
Sensibilité (%)	96.1	93.4	90.3	88.6

Tableau 4.6: Sensibilité de détection pour 4 participants avec $\tau = 50\%$.

Sujet	S01	S03	S07	S09
Sensibilité (%)	93.2	93.1	93.0	94.4
Moyenne (%)	93.4			

pour les 4 sujets, en utilisant différents taux de recouvrement avec les détéection de référence.

4.4.2.2 Séparation parole/autres sons

Après détection, un modèle SVM-GSL est utilisé pour séparer la parole des sons de la vie courante. À l'instar des expérimentations avec la base de l'ESIGETEL, nous utilisons des vecteurs de 16 coefficients *MFCC*, calculés sur des fenêtres de 16ms avec 8ms de recouvrement. Le tableau 4.7 présente les résultats obtenus. D'après le scénario d'enregistrement, les conversations téléphoniques ont eu lieu au même endroit pour tous les participants. Les deux canaux C6 et C7 étaient les plus excités et présentaient, de ce fait, les meilleurs RSB. Les résultats pour les deux canaux sont illustrés au tableau 4.7. Les performances de reconnaissance sont évaluées par rapport aux phrases détectées (colonne **FN Reco.** pour *False Negative Recognition* ou fautes de reconnaissance). La colonne **TP D+R** (*True Positive Detection and Recognition* qui représente les événements détectés et reconnus

correctement) indique les performances du système en prenant en considération les deux modules impliqués jusque-là (détection et séparation paroles/sons de la vie courante). Les résultats sont plutôt bons, la plupart des phrases ayant été détectées et reconnues comme de la parole. De plus, aucun son de la vie courante n'a été identifié comme de la parole (non indiqué dans le tableau).

Tableau 4.7: Performances de la séparation entre parole et autres sons. **FN Det** : détection manquées. **FN Reco.** fausses reconnaissances par rapport aux références détectées. **TP D+R** : phrases détectées et reconnues correctement.

Sujet	# de phrases	Canal	FN Det.	FN Reco.	TP D+R
S01	44	C6	0	4	40
		C7	1	2	41
S03	41	C6	0	2	39
		C7	1	7	33
S07	45	C6	4	5	36
		C7	6	3	36
S09	40	C6	0	0	40
		C7	0	0	40
S13	42	C6	0	4	38
		C7	1	0	41

4.4.2.3 Reconnaissance des sons de la vie courante

Outre la parole, 18 classes de sons sont retenues comme des sons d'intérêt (SOI ou *Sound of Interest*) pour les expérimentations (*BrushingTeeth*, *Coughing*, *HandsClapping*, *DoorClapping*, *DoorOpening*, *ElectricDoorLock*, *WindowShutters*, *Curtains*, *WindowShutters +Curtains*, *Vacuum-Cleaner*, *PhoneRing*, *MusicRadio*, *SpeechRadio*, *Speech+MusicRadio*, *Paper*, *Keys*, *Kitchenware* et *Water*). Une liste plus complète de classes contiendrait une trentaine de classes mais beaucoup d'entre celles-ci sont rares ou bien absentes dans certains scénarios. Si un événement est détecté sur plusieurs canaux, plusieurs stratégies pourraient être utilisées par un algorithme de reconnaissance. On pourrait, par exemple, reconnaître l'événement sur tous les canaux excités et, en utilisant une stratégie de fusion, décider de l'étiquette finale à lui attribuer. Dans ce travail, nous partons de l'hypothèse que le meilleur canal en terme de RSB devrait être utilisé pour la reconnaissance, sans tenir compte des autres canaux. Pour tester cette hypothèse, nous la comparons à une stratégie qu'on appelle **oracle**. Elle consiste à utiliser l'information disponible *a posteriori* quant au meilleur canal qui devrait être utilisé pour la reconnaissance. Les performances de reconnaissance oracle sont calculées comme suit : une reconnaissance est correcte s'il existe un au moins un canal dont la reconnaissance automatique correspond à l'annotation de référence, quel que soit son RSB. l'hypothèse du meilleur canal peut être posée comme suit : une reconnaissance est correcte si et seulement si la reconnaissance du signal du canal au meilleur RSB correspond à la référence. Le tableau 4.8 montre les résultats de reconnaissance par rapport aux événements détectés (colonne

TP R), ainsi que les performances des deux composantes du système (détection et reconnaissance) par rapport à tous les événements de référence (colonne **TP D+R**). L'utilisation de la stratégie oracle n'apporte pas de gain (**Oracle TP D+R**) par rapport à l'utilisation du canal au meilleur RSB, l'utilisation de celui-ci en reconnaissance donnant toujours des performances supérieures ou égales à celles des autres canaux. Bien entendu, des stratégies de fusion seraient également utilisables.

Tableau 4.8: Performances du système. Un seul canal (celui du meilleur RSB) est utilisé pour reconnaître le signal détecté. **TP D** : performances de détection par rapport aux références. **TP R** : performances de reconnaissance par rapport aux références détectées. **TP D+R** : performances de reconnaissance par rapport à toutes les références (performances globales du système). **Oracle TP D+R** : performances globales si, pour la reconnaissance, le canal (s'il en existe) qui *donnerait* une reconnaissance correcte est utilisé, quel que soit son RSB. **Oracle TP D+R = TP D+R** signifie le meilleur canal en terme de RSB est meilleur qu'on puisse utiliser pour la reconnaissance.

Sujet	**# Occur. SOI**	**TP D(%)**	**TP R(%)**	**TP D+R(%)**	**Perf. Oracle TP D+R(%)**
S01	230	83.9	69.4	58.2	58.2
S03	175	80.6	65.2	52.6	52.6
S07	245	82.4	68.8	56.7	56.7
S09	268	91.8	74.8	68.7	68.7

Le tableau 4.9 montre les performances moyennes de reconnaissance par canal. Vu la surface de l'appartement et le scénario suivi, aucun canal n'était en mesure de capter tous les événements acoustiques et remplacer ainsi tous les autres canaux. L'utilisation de tous les canaux en reconnaissance améliore substantiellement les résultats. Certains canaux sont bien meilleurs que d'autres. Cela s'explique aisément par le fait que les sujets passaient plus de temps à proximité de ceux-ci (les canaux C1 et C2, par exemple, sont situés dans la salle à manger et le coin cuisine respectivement, endroit où, selon notre expérience d'annotation, les sujets passaient le plus de temps).

Tableau 4.9: Performance moyenne par canal, tous sujets confondus

Canal	C1	C2	C3	C4	C5	C6	C7	Fusion de tous les canaux
Moy. TP D+R(%)	31.3	33.3	14.9	24.1	21.8	13.3	9.6	59.1

4.5 Conclusions et Perspectives

Dans ce chapitre, nous avons testé trois méthodes pour la classification des sons de l'environnement, toutes issues de la reconnaissance automatique du locuteur.

Comme en RAL, l'utilisation des SVMs avec des vecteurs acoustiques en entrée n'est pas appropriée. Les temps d'apprentissage sont très longs pour des performances inférieures à celles des GMMs. Sans affirmer que, pour la REA, les GMMs auront le même succès qu'en RAL, cette méthode simple d'implémentation et rapide d'exécution, reste très intéressante. Elle constituerait une bonne

solution pour des problèmes de REA où le nombre de classes de sons est faible, les classes sont faciles à discriminer ou bien les ressources sont limitées.

Le noyau SVM-GSL, bien que plus exigeant en ressources, donne de meilleurs résultats que les GMMs, en dépit de la quantité limitée de données utilisées pour créer le modèle UBM. Par ailleurs, des classes dont les enregistrements sont très courts (*DoorClapping* et *Sneeze*, par exemple) sont très bien reconnues, malgré une taille relativement importante du modèle UBM (512 ou 1024).

L'analyse de la matrice de confusion nous a permis de réaliser que les classes enregistrées entièrement dans les mêmes conditions acoustiques seraient plus faciles à reconnaître. Or cela reste à confirmer car, d'une part, le nombre des classes concernées est petit (trois classes) et d'autre part, au moins deux (*ElectricalShaver* et *HairDryer*) seraient faciles à distinguer (nous verrons le spectrogramme de quelques enregistrements de ces deux classes au chapitre suivant). Certaines classes, par contre, sont souvent confondues avec d'autres classes, plus précisément, avec une seule autre classe la plupart du temps.

Pour les expérimentations sur le corpus du projet SWEET-HOME, la méthode SVM-GSL a été retenue. La séparation entre la parole et les autres sons a donné de très bonnes performances, la plupart des phrases étant en effet détectées et reconnues comme de la parole. Pour la reconnaissance des sons de la vie quotidienne, un taux de rappel (nombre de sons d'intérêt détectés et reconnus correctement, par rapport à tous les sons d'intérêt dans les scénarios) très intéressant a été obtenu. Pour les quatre scénarios, plus de 50% des événements appartenant aux classes de sons d'intérêt (dix-huit classes) ont été détectés et reconnus correctement.

L'utilisation de plusieurs canaux est indispensable pour capter les événements dans tout l'appartement. Pour la reconnaissance, l'utilisation du canal au meilleur RSB donne de meilleurs résultats que les canaux aux RSBs inférieurs.

Le rejet des sons sans intérêt n'a pas été pris en considération dans ces expérimentations. Il constituera l'une des points intéressants pour les futurs tests sur le corpus du projet. Des méthodes comparables à celles utilisées en vérification du locuteur sont en perspectives. Pour simplifier cela, l'idée de base est de considérer les sons d'intérêt comme des locuteurs et les autres sons de l'appartement comme des « imposteurs ». La quantité de données disponibles dans le corpus semble suffire pour créer les modèles nécessaires.

Les méthodes de modélisation de la variabilité intra-locuteur sont souvent utilisées en amont des

méthodes de classification. Nous les considérons comme une piste particulièrement intéressante pour modéliser la variabilité entre enregistrements appartenant à une même classe.

Enfin, outre les *MFCC*, des coefficients acoustiques modélisant d'autres caractéristiques du signal pourraient également être utilisés pour la classification du son. Nous traitons ce point au chapitre suivant.

CHAPITRE 5

CLASSIFICATION DU SON AVEC PLUSIEURS FAMILLES DE COEFFICIENTS

Pour toutes les expérimentations que nous avons effectuées jusque-là, nous n'avions considéré l'utilisation que d'un seul type de paramètres acoustiques : *MFCC*. Comme mentionné précédemment, les coefficients *MFCC* sont couramment utilisés en reconnaissance de la parole et en reconnaissance et vérification du locuteur. La raison en est que les filtres triangulaires utilisés dans le calcul de ces coefficients correspondent au système auditif humain plus que les filtre linéaires [Picone, 1993] [Schroeder, 1977]. Toutefois, il existe un nombre non négligeable de coefficients qui peuvent être extraits d'un signal audio, chaque famille de coefficients pouvant mettre en exergue une ou plusieurs caractéristiques du signal. Certaines familles de coefficients sont effectivement utilisées dans d'autres domaines audio voisins tels que la reconnaissance de genre ou d'instruments musicaux [Peeters, 2004] [West and Cox, 2004] [Jang et al., 2008] [Duxbury et al., 2003] [Eronen and Klapuri, 2000] [Mierswa and Morik, 2005].

Ce chapitre propose une étude d'un certain nombre de coefficients acoustiques pour les exploiter en reconnaissance des sons de l'environnement. Comme dans le chapitre précédent, les méthodes de classification utilisées sont basées sur les GMMs et les SVMs. L'utilisation des GMMs est semblable à celle que nous avons appliquée au chapitre précédent. Pour les SVMs, en revanche, nous testons deux méthodes pour la transformation de séquences de vecteurs. Nous faisons une comparaison entre les performances obtenues avec chaque famille de coefficients ainsi que celles obtenues en utilisant toutes les familles conjointement. Les sections suivantes expliquent les motivations de cette démarche et les techniques mises en œuvre pour la transformation de séquences de vecteurs.

5.1 Motivations de l'utilisation de plusieurs familles de coefficients

En reconnaissance ou vérification du locuteur, la plupart des techniques d'extraction de caractéristiques utilisent des informations spectrales de bas niveau qui véhiculent les caractéristiques du conduit vocal [Kinnunen and Li, 2010b]. Les informations spectrales sont extraites depuis des fenêtres d'une durée de 20 à 30 ms de signal de parole en utilisant le carré de l'amplitude de la **transformée de Fourier discrète** (DFT pour *Discrete Fourier transform*). Étant donnée la lente

variabilité du conduit vocal, le signal est presque stationnaire sur la fenêtre d'analyse. Ceci rend appropriée une analyse par blocs de points basée sur la DFT [Sahidullah and Saha, 2012].

Les coefficients *MFCC* ont initialement été proposés pour une tâche bien particulière, la reconnaissance de la parole. Ils ont par la suite trouvé leur utilisation auprès de la communauté de la reconnaissance du locuteur bien que les deux tâches soient de nature différente. Ils sont même les coefficients les plus utilisés pour les deux tâches du fait de l'existence de méthodes de calcul rapides et d'une certaine robustesse au bruit [Sahidullah and Saha, 2012].

D'après Kinnunen [Kinnunen, 2003] [Kinnunen, 2005], le fait que les *MFCC* soient l'une des familles de coefficients les plus utilisées dans les deux domaines peut s'avérer quelque peu « ironique » étant données les natures différentes des deux problèmes. En effet, l'un des problèmes les plus gênants en reconnaissance de la parole est la variabilité des locuteurs, alors qu'en reconnaissance du locuteur, c'est justement cette variabilité que l'on cherche à exploiter pour discriminer les locuteurs.

Nous restons tout de même sceptiques quant à ces observations car, d'une part, les *MFCC* rencontrent un grand succès en reconnaissance du locuteur comme en témoignent les bons résultats obtenus avec ces coefficients depuis de très nombreuses années. En étudiant la littérature récente en matière de reconnaissance du locuteur, on peut constater que tous les efforts se sont concentrés sur les méthodes de classification plutôt que sur les coefficients acoustiques utilisés. En effet, tandis que les *MFCC* constituent souvent le choix standard en matière de coefficients acoustiques, beaucoup d'algorithmes de classification ont été étudiés (VQ, GMMs, ANNs, SVMs, etc.). D'autre part, les *MFCC* peuvent s'avérer très utiles pour la différentiation de certains classes de sons. En effet, il est vrai que ces coefficients sont conçus pour modéliser la parole (d'où l'utilisation de filtres triangulaires qui sont plus étroits pour les basses fréquences, c'est à dire les fréquences où se situe majoritairement le signal de la parole), mais cela peut s'avérer bien utile pour différencier, par exemple, certains sons humains d'autres sons dont les plages de fréquences les plus importantes se trouvent dans une autre partie du spectre, ou bien remplissent tout le spectre.

Pour illustrer ce dernier point, examinons les spectrogrammes de trois classes de sons très différents : des cris humains, le bruit d'un moteur électrique (rasoir), et des claquements de porte. Les figures de 5.1 à 5.3 montrent les spectrogrammes de 4 enregistrements de chacune de ces 3 classes respectivement. On peut y observer que, pour les cris, les basses fréquences (entre 800Hz et 1500Hz environ) sont toujours d'une intensité élevée, contrairement aux hautes fréquences (de plus de 5500Hz) qui sont quasiment absentes.

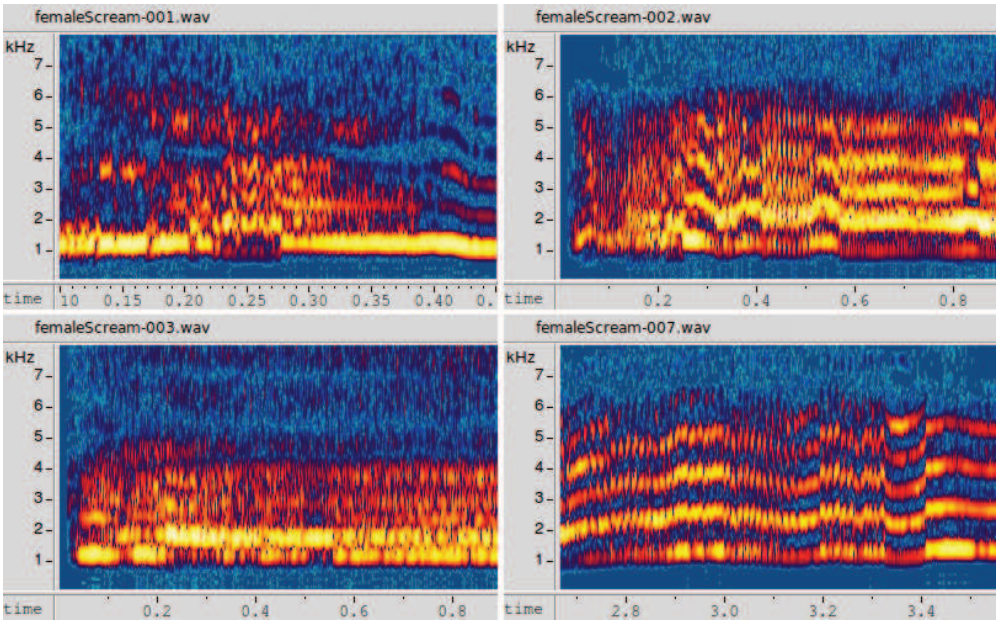

Figure 5.1: Spectrogrammes de cris humains

Cela constitue une différence exploitable pour séparer cette classe de celle des claquements de porte, pour lesquels on peut observer une saturation du spectre (quoique d'une courte durée) mettant en avant une plus large plage de fréquences (incluant les fréquences de 5500Hz à 8000Hz, à l'opposé des cris). De même, on remarque que les fréquences de moins de 1500Hz sont quasiment absentes pour le son de moteur, le différenciant ainsi des cris.

Toutefois, la différence entre certaines autres classes de sons peut parfois s'avérer moins évidente en se basant uniquement sur des coefficients cepstraux. Les deux figures 5.4 et 5.5 illustrent les spectrogrammes pour des enregistrements de toux et de bris de glace respectivement. Bien que la variation dans le temps soit visible et pourrait, en utilisant une méthode de classification appropriée, être utilisée pour différencier les deux classes, les parties où le spectre est saturé, communes aux deux classes, pourraient être une source de confusion pour les GMMs ou les SVMs, du moins pour les variantes des SVMs présentées au chapitre 3, dans lesquelles l'information temporelle n'est pas prise en considération.

L'utilisation de coefficients basés sur l'échelle de Mel (dont les *MFCC*) pour la reconnaissance du locuteur est sous tendue par l'hypothèse que le système auditif humain constitue le meilleur appareil pour reconnaître un locuteur. Or, en réalité, cela n'a pas été confirmé et des résultats

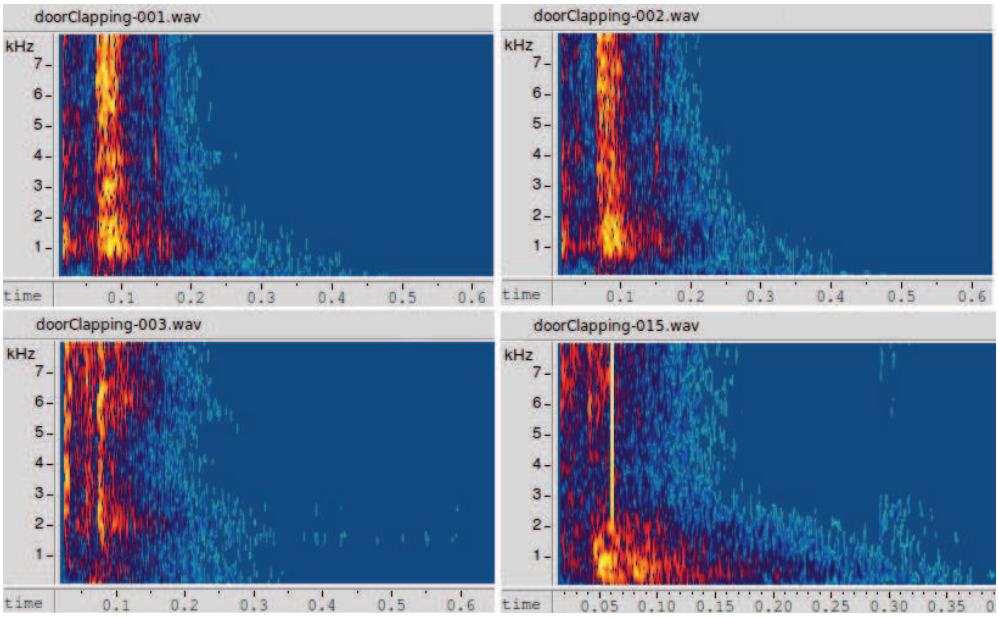

Figure 5.2: Spectrogrammes de claquements de porte

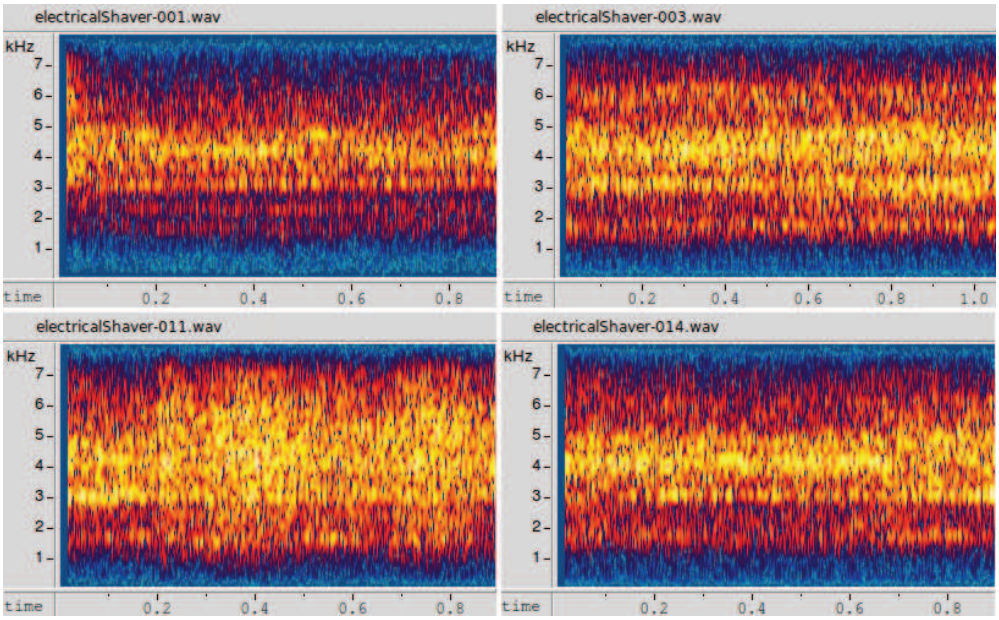

Figure 5.3: Spectrogrammes de bruits de moteur électrique (rasoir)

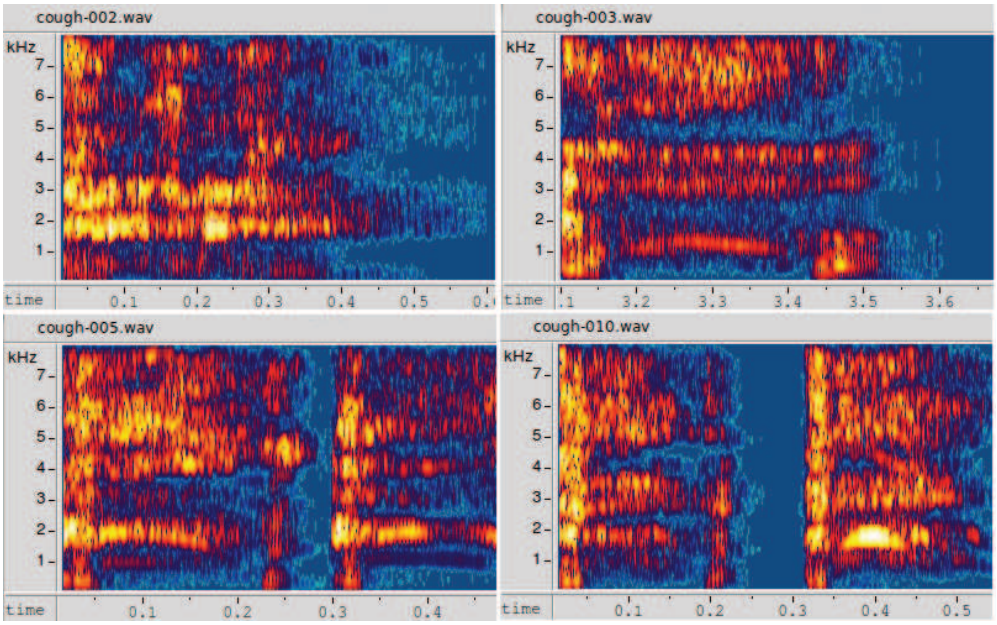

Figure 5.4: Spectrogrammes de toux

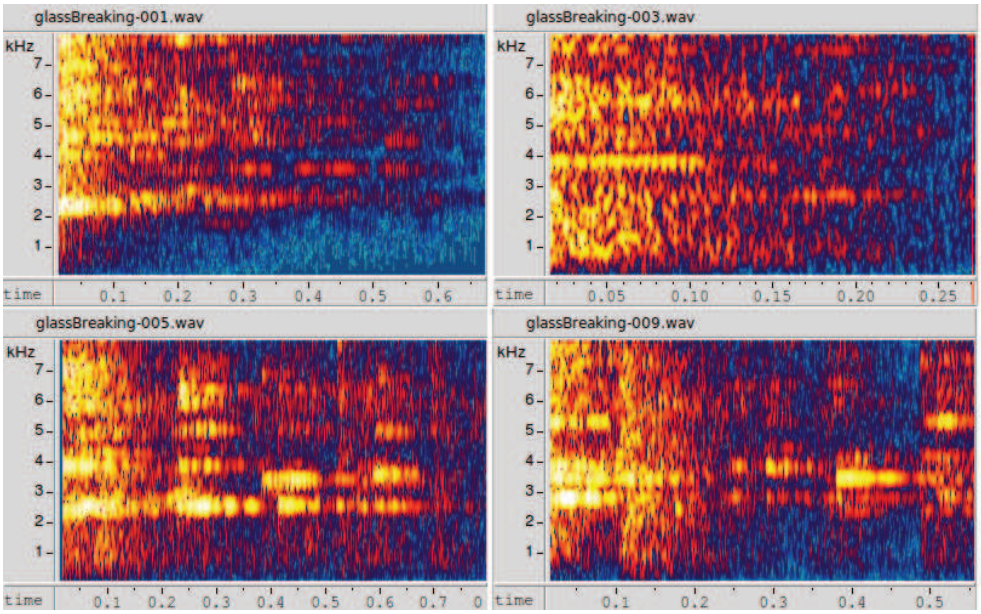

Figure 5.5: Spectrogrammes de bris de glace

opposés existent [Kinnunen, 2003].

Ce constat reste aussi pertinent lorsqu'il s'agit de reconnaître certains sons de la vie quotidienne. En effet, et d'après notre expérience d'annotation du corpus du projet Sweet-Home [Sehili et al., 2012b] (plusieurs personnes ont participé au processus d'annotation), nombre de sons étaient très difficiles, voire impossibles à reconnaître à l'oreille humaine en se basant uniquement sur les enregistrements audio. L'utilisation des vidéos associées nous a été indispensable pour annoter le corpus.

Par ailleurs, et paradoxalement par rapport à ce que l'on vient de mentionner, certains sons sont si aisément reconnaissables que toute confusion avec un ou plusieurs autres sons s'avère peu probable. Dans cet esprit, nous allons étudier le comportement d'un système de classification des sons de l'environnement utilisant en entrée différents types de paramètres acoustiques. Les objectifs de cette étude sont les suivants :

— Étudier les performances de reconnaissance par rapport à chaque « classe de son/famille de coefficients » dans la perspective de trouver les meilleurs modèles appropriés à une classe donnée.
— Simplifier le système en utilisant les coefficients les plus discriminants et les moins complexes qui permettent de distinguer deux classes de sons données ou, au mieux, une classe de toutes les autres.

5.1.1 Familles de coefficients retenues

L'un des objectifs que nous nous sommes donnés est l'étude des caractéristiques de chaque famille de coefficients. Pour cela, nous analysons l'information qu'elle permet d'extraire du signal et ce qui justifie son utilisation pour une classe de sons particulière, tels que les sons humains, sons produits par des objets métalliques ou sons stationnaires, etc. Cela étant dit, davantage de données sont indispensables si l'on veut être capable de tirer des conclusions pertinentes quant à l'adéquation d'une famille de coefficients pour distinguer une classe de sons d'une autre. L'approche retenue dans ce travail consiste à mener des expérimentations en utilisant chacune des familles de coefficients et d'observer par la suite celles qui permettent de discriminer au mieux deux classes de sons.

Les coefficients diffèrent par leur complexité de calcul, leur nombre de caractéristiques et leur portée par rapport à la longueur du signal. De ce fait, ils diffèrent également par l'information

qu'ils permettent d'extraire du signal. d'après Peeters [Peeters, 2004], plusieurs critères peuvent être utilisés pour placer les coefficients dans différentes catégories. Les 4 critères suivants y sont mentionnés :

1. La **stabilité** *(Steadiness)* ou la **dynamicité** *(Dynamicity)* du coefficient : l'extraction du coefficient peut s'appliquer au signal à un instant donné dans le temps ou bien prendre en considération la variation du signal dans le temps (moyenne, écart-type, dérivée du modèle de Markov d'un paramètre, etc.).
2. L'**étendue dans le temps** *(Time Extent)* de l'information que contient le coefficient : certains coefficients ne s'appliquent qu'à une partie bien particulière du signal (ex. l'attaque du signal) alors que d'autres concernent tout le signal (ex. le coefficient de *Loudness*).
3. Le niveau d'**abstraction** *(Abstractness)* du coefficient : les coefficients modélisant le système auditif humain (les *MFCC* par exemple), ont un plus haut niveau d'abstraction en comparaison avec les coefficients qui représentent des caractéristiques physiques du signal (le *ZCR*, par exemple).
4. Le **processus** d'extraction lui-même. Partant de ce critère, on peut distinguer les catégories de coefficients suivantes :
   — Coefficients extraits directement depuis le signal (c'est à dire, depuis les échantillons du signal, sans aucune transformation préalable), comme, par exemple, le *ZCR* qui représente simplement le taux de transitions entre les valeurs positives et négatives du signal.
   — Coefficients extraits après avoir appliqué une transformée au signal (DFT, transformée en ondelettes, etc.). Des exemples de ces coefficients sont les *MFCC*, le *Spectral Roll-Off*, etc.
   — Coefficients en relation avec un modèle donné du signal comme, par exemple, le modèle sinusoïdal de la source ou du filtre.
   — Coefficients essayant de d'imiter le système auditif humain (ex. filtres de Bark).

Les familles de coefficients présentées au chapitre 2 sont retenues pour des expérimentations utilisant les GMMs et les SVMs. Quatorze familles (tableau 5.1) sont utilisées avec des GMMs. Les premières expérimentations en utilisant plusieurs familles de coefficients ensemble ont été faites avec des GMMs.

Les expérimentations suivantes ont été réalisées avec des SVMs. Pour cela, nous avons retenu toutes les familles de coefficients que nous utilisées avec des GMMs, auxquelles nous avons également ajouté les statistiques de l'enveloppe d'amplitude du signal. De plus, nous calculons les

caractéristiques du *Temporal Shape Statistics* pour tous les échantillons du signal au lieu de le faire sur plusieurs fenêtres successives de courte durée. Cela permettra d'obtenir, pour ce coefficient, un vecteur de 4 caractéristiques quelle que soit la durée du signal.

Tableau 5.1: Coefficients retenus pour les expérimentations

Famille de coefficients	**Nombre de caractéristiques**
MFCC	16
Loudness	24
Spectral Crest Factor Per Band	19
Spectral Flatness Per Band	19
Complex Domain Onset Detection	1
Perceptual Sharpness	1
Perceptual Spread	1
Spectral Roll-Off	1
Spectral Decrease	1
Spectral Flatness	1
Spectral Variation	1
Spectral Slope	1
Spectral Shape Statistics	4
Temporal Shape Statistics	4
Total	94

Ces familles de coefficients ont été retenues suite à une étude que nous avons effectuée sur les classes de sons de la base de l'ESIGETEL. En effet, pour chaque famille de coefficients, nous avons observé qu'il existe des couples de classes séparables (en se basant sur cette famille). Plusieurs exemples seront donnés tout au long de ce chapitre. Pour l'extraction des coefficients acoustiques utilisés dans ce chapitre, nous avons utilisé la bibliothèque yaafe [1].

Comme nous l'avons vu au chapitre précédent, une des utilisations efficaces des SVMs pour des problèmes semblables au notre, requiert l'utilisation de techniques pour la transformation d'une séquence de vecteurs en un seul vecteur. Si le calcul du *Temporal Shape Statistics* sur la durée totale du signal au lieu de l'utilisation de fenêtres de courte durée ne pose pas de vrais problèmes, le calcul des coefficients nécessitant une DFT doit se faire sur des fenêtres de très courte durée, durant lesquelles le signal est supposé être stationnaire. Pour ce type de coefficients, nous obtenons en principe une séquence de vecteurs. Nous aurons donc besoin de moyens pour transformer ces séquences de longueur variable en un seul vecteur de taille fixe.

Dans nos précédentes expérimentations, le noyau SVM-GSL a été utilisé pour résoudre ce pro-

1. http ://yaafe.sourceforge.net

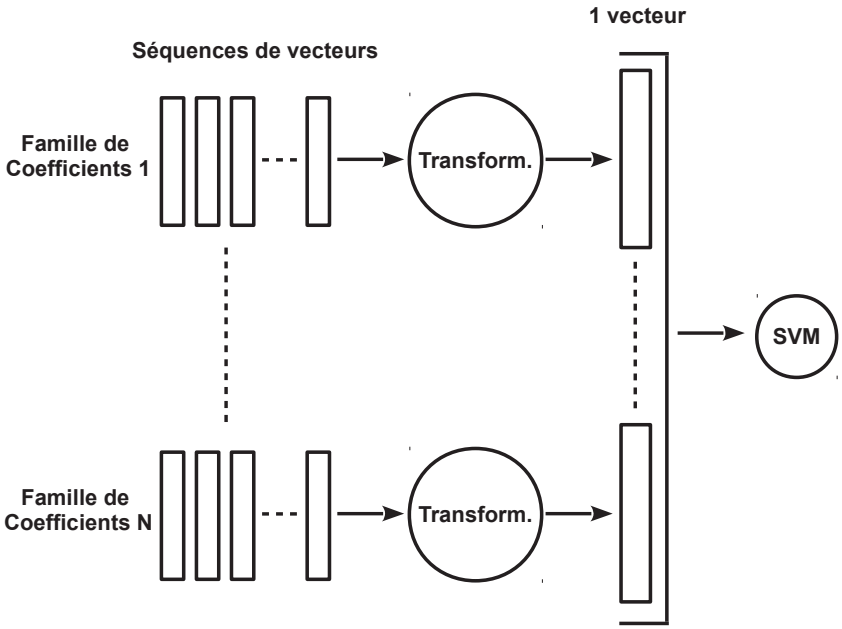

Figure 5.6: Construction d'un seul vecteur à partir de plusieurs familles de coefficients

blème. Dans ce chapitre, nous présentons deux autres méthodes pour transformer une séquence de vecteurs en un seul vecteur. Elles seront appliquées à tous les coefficients, sauf, bien entendu, le *Temporal Shape Statistics* pour lequel on peut déjà extraire un seul vecteur par enregistrement. La transformation est appliquée à chaque famille de coefficients. Les vecteurs obtenus, représentant chacun une famille, seront utilisés conjointement pour former un seul grand vecteur contenant des informations de toutes les familles de coefficients (figure 5.6).

Nombre de ces familles de coefficients sont basées sur l'information spectrale et pourraient, ainsi, présenter des similitudes ou des informations redondantes. De surcroît, nous partons de l'hypothèse que, pour certaines paires de classes, un nombre réduit de coefficients permettra de les discriminer. De ce fait, nous aurons recours aux méthodes de sélection de paramètres, qui seront utilisées afin de réduire le nombre de caractéristiques des vecteurs utilisés en entrée d'un SVM.

Avant d'expliciter la mise en oeuvre que nous avons réalisée pour l'emploi de plusieurs familles de coefficients avec les GMMs et les SVMs, ainsi que les méthodes de sélection de caractéristiques, nous présentons quelques exemples montrant les possibilités de séparation qu'offrent certaines familles de coefficients pour des couples de classes de sons donnés.

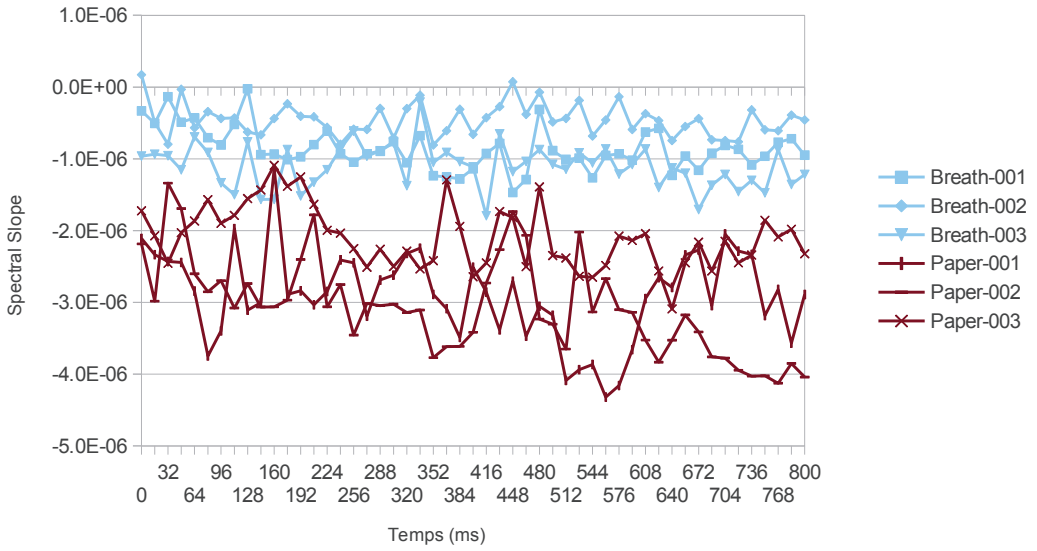

Figure 5.7: *Spectral Slope* pour la respiration *(Breath)* et la manipulation de papier *(Paper)*

Les figures de 5.7 à 5.9 montrent la variation dans le temps d'un coefficient pour deux classes de sons différentes (3 enregistrements, d'une durée de 800 ms chacun, par classe sont utilisés). Pour des raisons de lisibilité, seules des familles de coefficients contenant une seule caractéristique par fenêtre (*Spectral Slope*, *Perceptual Sharpness* et *Spectral Roll-Off* respectivement) sont choisies. Sur chacune des figures, on peut observer que, pour chaque coefficient, les intervalles de valeurs occupés par les fenêtres d'une classe donnée, sont bien distincts de ceux de l'autre classe. Par ailleurs, il y a très peu de variations dans le temps pour certaines classes par rapport à un coefficient donné ; à l'image des valeurs du *Spectral Roll-Off* pour le bruit du rasoir électrique (figure 5.8).

Notre approche consiste à utiliser tous les coefficients ensemble et à mesurer l'apport qui peut en résulter par rapport à l'utilisation des *MFCC* seuls. Pour cela nous utiliserons des GMMs et des SVMs. Nous testerons également chacune des familles séparément et analyserons ses performances pour chaque classe de sons. Ces dernières expérimentations seront faites en utilisant des SVMs.

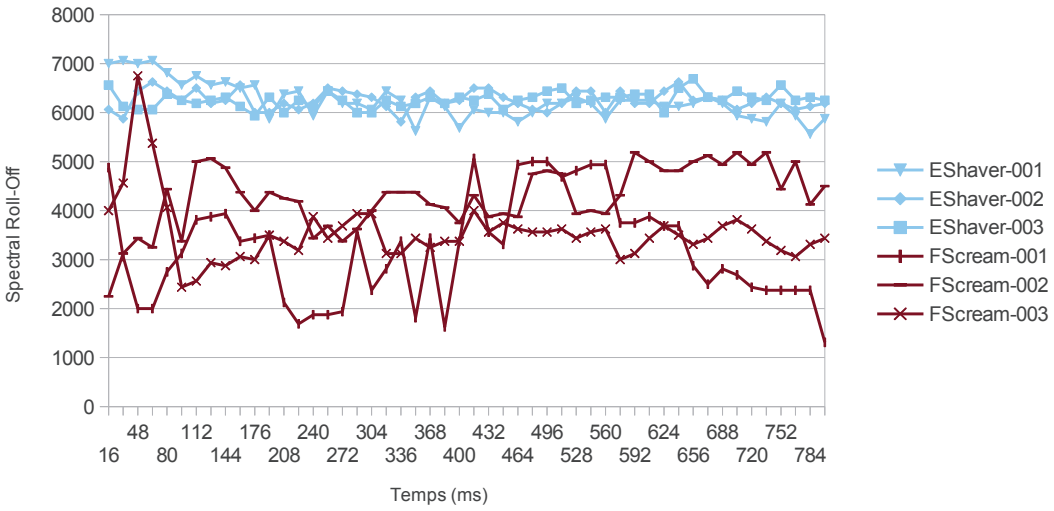

Figure 5.8: *Spectral Roll-Off* d'un rasoir électrique *(EShaver)* et des cris d'une personne de sexe féminin *(FScream)*

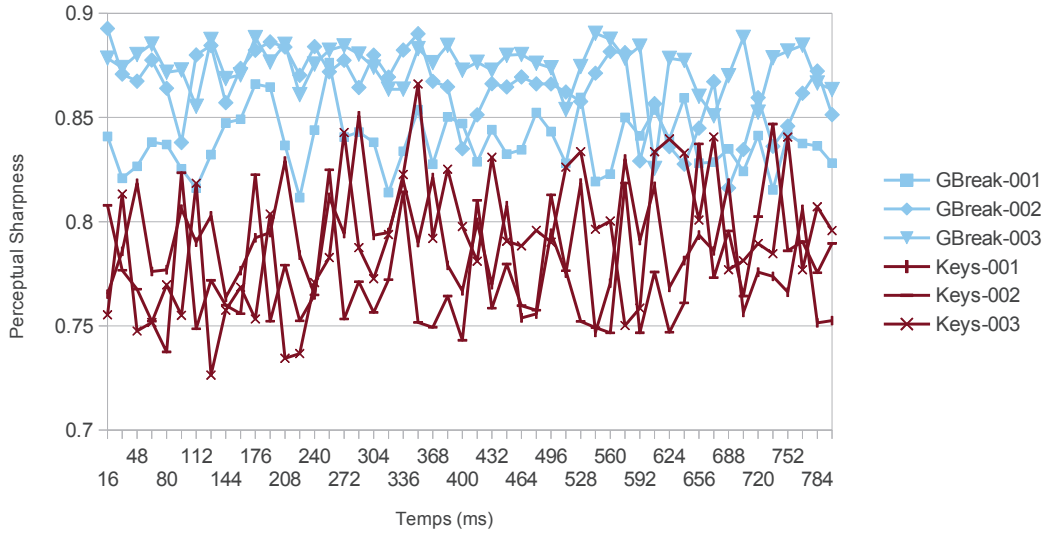

Figure 5.9: *Perceptual Sharpness* pour le bris de glace *(GBreak)* et le bruit de clés *(Keys)*

5.2 Approches pour utiliser plusieurs familles de coefficients ensemble

Les sections suivantes traitent de l'utilisation de ces familles de coefficients avec des GMMs et des SVMs. La mise en oeuvre des GMMs, présentée ci-après, ne diffère pas de celle utilisée jusqu'à maintenant pour les *MFCC* (chapitre 4). La seule différence réside dans le fait que les vecteurs en entrée contiennent des coefficients de toutes les familles présentées au tableau 5.1. La description en sera donc assez brève.

En revanche, pour l'application des SVMs, le problème de classification de séquences de vecteurs se pose de nouveau ; cette fois avec un nombre beaucoup plus élevé de caractéristiques. Comme nous l'avons vu au chapitre précédent, le noyau SVM-GSL est basé sur le calcul d'un modèle UBM. L'efficacité de ce noyau est donc étroitement liée à la qualité du modèle UBM. Le modèle UBM est un modèle GMM. Les modèles GMMs ne sont pas efficaces avec des données d'un nombre élevé de caractéristiques. En fait, le nombre de vecteurs nécessaire pour estimer la densité de probabilité augmente exponentiellement avec le nombre de caractéristiques. Ce problème est appelé **fléau de la dimension** (*Curse of Dimensionality*) [Jain et al., 2000].

Par ailleurs, l'une des caractéristiques intéressantes des coefficients *MFCC* est l'absence de corrélation entre les différentes caractéristiques d'un vecteur. Cela est d'une importance cruciale car, lors de l'estimation d'un modèle UBM, il est supposé qu'il n'y a pas de corrélation entre les caractéristiques du vecteur en entrée, et de ce fait, on utilise des matrices de covariance dont les éléments hormis ceux de la diagonale sont nuls.

Beaucoup de familles de coefficients du tableau 5.1 sont calculées en se basant sur l'information spectrale et certaines de leurs caractéristiques sont, de ce fait, corrélées. En effet, en effectuant une analyse en composantes principales sur les données, nous avons constaté que près de la moitié des caractéristiques (environ 40 sur 94) ne sont pas significatives. La somme des valeurs propres qui leurs sont associées constitue moins de 5% de la somme totale des valeurs propres. Pour une éventuelle projection des données dans l'espace des vecteurs propres, on pourrait tout simplement se passer de ces caractéristiques. Par ailleurs, comme nous le verrons dans la section 5.3, l'utilisation de toutes les familles de coefficients ensemble avec des GMMs donne de moins bons résultats que ceux obtenus avec les *MFCC* seuls.

Pour les expérimentations réalisées en utilisant plusieurs familles de coefficients, nous n'envisageons donc pas l'utilisation du noyau SVM-GSL. De ce fait, l'un de nos objectifs, ici, est de trouver

d'autres moyens pour transformer une séquence de vecteurs en un seul vecteur. Dans cet esprit, deux méthodes sont développées dans ce chapitre. La première consiste à calculer des statistiques de haut niveau pour chaque caractéristique. Ainsi obtient-on, depuis une séquence de vecteurs, un vecteur unique contenant les coefficients statistiques issus des coefficients acoustiques en entrée. La seconde méthode consiste à utiliser un algorithme de **discrétisation** de valeurs continues [Dougherty et al., 1995] [Liu et al., 2002] [Kotsiantis and Kanellopoulos, 2006] qui, pour chaque caractéristique d'un vecteur de coefficients acoustiques, détermine un certain nombre d'intervalles. Les intervalles sont déterminés de sorte que, dans chacun, une classe soit plus probable que l'autre. Pour toutes les caractéristiques, et pour tous les intervalles, la probabilité de la classe de sons est estimée. Les différentes probabilités sont utilisées en guise de caractéristiques pour construire un vecteur utilisé avec des SVMs. Pour faire la distinction entre les deux méthodes, nous les appelons **SVM-StatVect** et **SVM-ProbVect** respectivement. Elles sont expliquées en détail dans les sections qui suivent.

5.2.1 Plusieurs familles de coefficients avec des GMMs

La façon la plus simple d'utiliser tous les types de coefficients avec des GMMs est probablement d'utiliser des vecteurs contenant tous les paramètres (figure 5.10). Pour cela, il faudra utiliser la même taille de fenêtre lors de l'extraction de chaque type de coefficient. Pour les familles du tableau 5.1, cela donnera des vecteurs de 94 éléments.

Par ailleurs, il serait intéressant de tester séparément et comparer certaines familles de coefficients, notamment celles contenant plusieurs caractéristiques par fenêtre (*MFCC*, *Loudness* et *Spectral Crest Factor Per Band* et *Spectral Flatness Per Band* qui contiennent respectivement 16, 24, 19 et 19 caractéristiques par fenêtre).

5.2.2 SVM avec vecteurs de coefficients statistiques (SVM-StatVect)

L'idée de base de cette méthode consiste à remplacer une séquence de valeurs par un nombre fixe, N, de valeurs quelle que soit la taille de la séquence. Ce processus pourra ainsi s'appliquer individuellement à chacune des caractéristiques des familles de coefficients retenues tant que la caractéristique se présente sous forme d'une séquence d'observations. Pour ce faire, on calcule un certain nombre de coefficients statistiques pour chaque caractéristique. Les coefficients retenus sont : la moyenne, l'écart-type, le rapport écart-type/moyenne, ainsi que des statistiques de haut

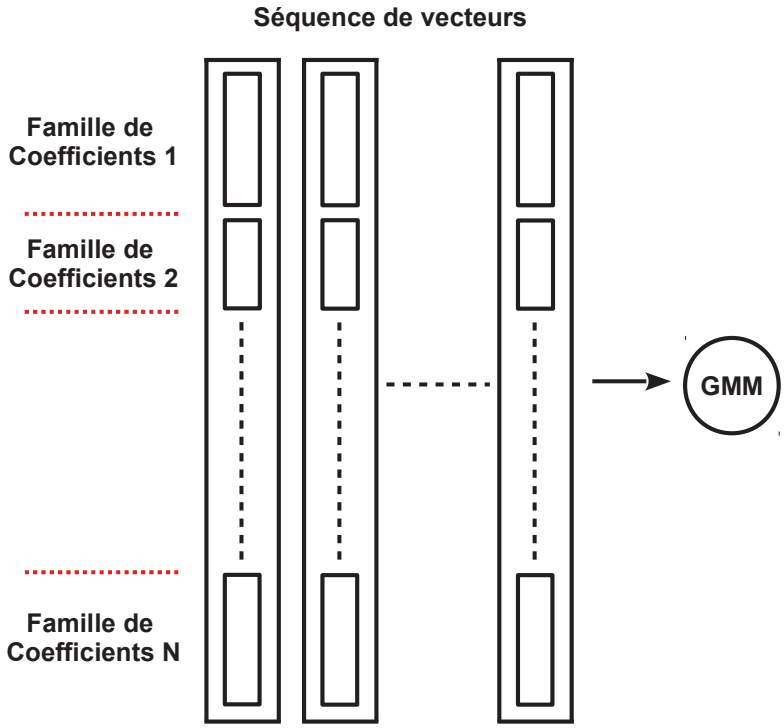

Figure 5.10: Utilisation de plusieurs familles de coefficients avec des GMMs

niveau : le *Centroid*, le *Spread*, le *Skewness* et le *Kurtosis* (équations de 5.1 à 5.4) [Gillet and Richard, 2004].

$$Centroid = \mu_1 \tag{5.1}$$

$$Spread = \sqrt{\mu_2 - \mu_1^2} \tag{5.2}$$

$$Skewness = \frac{2\mu_1^3 - 3\mu_1\mu_2 + \mu_3}{Spread^3} \tag{5.3}$$

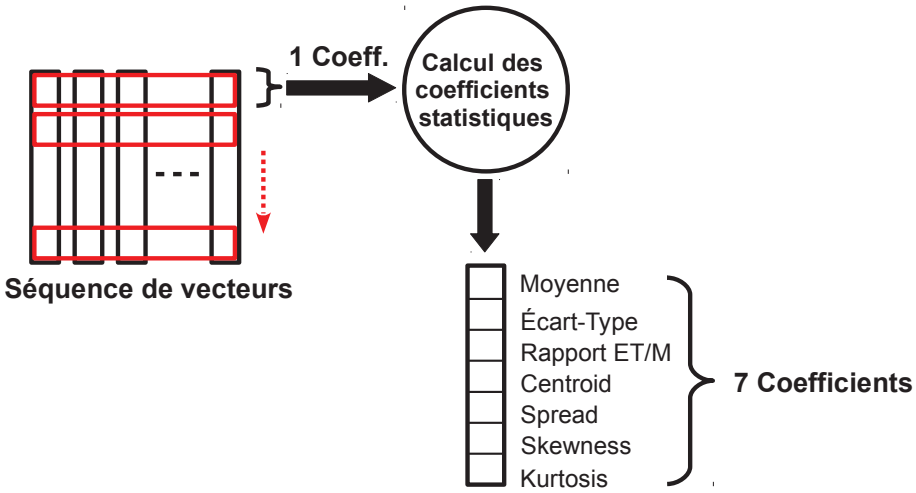

Figure 5.11: Transformation des valeurs d'une caractéristique en un vecteur de coefficients statistiques

$$Kurtosis = \frac{-3\mu_1^4 + 6\mu_1\mu_2 - 4\mu_1\mu_3 + \mu_4}{Spread^4} - 3 \tag{5.4}$$

où $\mu_i = \frac{\sum_{k=0}^{N-1} k^i \cdot A_k}{\sum_{k=0}^{N-1} A_k}$, et A_k est la k-ième valeur de la caractéristique A.

La figure 5.11 montre le processus de transformation. Ainsi, si on calcule ces coefficients statistiques pour une séquence de vecteurs *MFCC* de 16 caractéristiques, on obtient 7 valeurs pour chacune des caractéristiques *MFCC*. Pour 16 coefficients *MFCC*, cela donnera un vecteur de coefficients statistiques de 7 $\times$ 16 = 112 dimensions.

Le tableau 5.2 montre le type de transformation appliqué à chaque famille de coefficients et le nombre de coefficients statistiques qui en résulte. Trois ensembles de coefficients statistiques sont utilisés, Stat-1, Stat-2 et Stat-3 (tableau 5.3). Pour la plupart des familles, tous les coefficients statistiques sont calculés (ensemble Stat-3). Les *Spectral Shape Statistics* représentent déjà le *Centroid*, le *Spread*, le *Skewness* et le *Kurtosis* des amplitudes spectrales, calculés par fenêtre. On en calcule donc la moyenne, l'écart-type et le rapport écart-type/moyenne (ensemble Stat-1). Enfin, pour le *Temporal Shape Statistics*, sont calculés le *Spread*, le *Skewness* et le *Kurtosis* (ensemble Stat-3). à la différence des autres coefficients, *Temporal Shape Statistics* peut être calculé soit par fenêtre, soit pour tout le signal. Pour une utilisation avec les GMMs, il est calculé par fenêtre

afin d'obtenir le même nombre d'observations que les autres coefficients (figure 5.10). Pour les SVMs, on le calcule sur l'intégralité du signal. Les coefficients obtenus sont ajoutés (concaténés) au vecteur contenant les coefficients statistiques des autres familles.

Tableau 5.2: Nombre de caractéristiques avant et après le calcul des coefficients statistiques

Famille de coeffcients	**# initial de caractéristiques**	**Transformation**	**# obtenu de caractéristiques**
MFCC	16	Stat-3	112
Loudness	24	Stat-3	168
Spectral Crest Factor Per Band	19	Stat-3	133
Spectral Flatness Per Band	19	Stat-3	133
Complex Domain Onset Detection	1	Stat-3	7
Perceptual Sharpness	1	Stat-3	7
Perceptual Spread	1	Stat-3	7
Spectral Rolloff	1	Stat-3	7
Spectral Decrease	1	Stat-3	7
Spectral Flatness	1	Stat-3	7
Spectral Variation	1	Stat-3	7
Spectral Slope	1	Stat-3	7
Spectral Shape Statistics	4	Stat-1	12
Temporal Shape Statistics	4	Stat-2	4
Envelope Shape Statistics	-	Stat-3	7
Total	94 utilisés avec GMMs		625 utilisés avec SVMs

Tableau 5.3: Types de transformations

Transformation	**Coefficients**
Stat-1	*Moyenne* *Écart-Type* *Ratio*
Stat-2	*Centroid* *Spread* *Skewness* *Kurtosis*
Stat-3	Stat-1 Stat-2

5.2.3 SVM avec vecteurs de probabilité (SVM-ProbVect)

Dans cette partie du chapitre, nous détaillons la seconde méthode utilisée pour la transformation de séquences de vecteurs. Avant de présenter la méthode de discrétisation retenue et expliquer comment nous nous en sommes servis pour le problème de transformation de séquences, nous présentons les étapes qui nous ont conduits vers ce choix.

Dans les figures 5.7-5.9, on peut observer que, hormis quelques exceptions, les valeurs du coefficient appartenant à une classe donnée, se répartissent sur un intervalle bien distinct de celui occupé par les valeurs de l'autre classe, et cela pour presque toutes les fenêtres d'analyse. Pour des données se présentant ainsi, une méthode pour classifier les différents points (ici les fenêtres d'analyse) consiste à choisir un seuil T qui divise le domaine de valeurs en deux intervalles. Ce choix de seuil peut ainsi être vu comme une *règle* permettant de distinguer une classe de l'autre.

5.2.4 Un sytème à base de règles ?

Le seuil T peut également être vu comme un séparateur linéaire. Les figures 5.12 et 5.13 montrent un exemple de séparateurs linéaires pour deux coefficients différents. Cela peut aussi être considéré comme une règle très simpliste permettant de distinguer une classe de l'autre. La règle exprimée dans la figure 5.12 est la suivante :

$$\text{Classe de son} = \begin{cases} \textit{Electric Shaver} & \textbf{Si } \textit{Spectral Roll-Off} \geq 5000 \\ \textit{Female Scream} & \textbf{Sinon} \end{cases}$$

Idéalement, on pourrait déterminer, en observant les données, l'ensemble de règles qui permettent de bien caractériser une classe de sons et retrouver ainsi les traits qui la distinguent des autres classes ou, du moins, de certaines classes. Toutefois, cette procédure requiert une analyse rigoureuse de toutes les combinaisons de classes qui existent, et cela pour chaque coefficient. Cette tâche peut malheureusement devenir le « goulot d'étranglement » du processus car, d'une part, le nombre de caractéristiques (94) et celui de combinaisons de classes (153 combinaisons pour 18 classes de sons) sont très élevés. D'autre part, pour des attributs à valeurs numériques, il n'est pas toujours évident de trouver la ligne qui sépare deux classes. Dans les figures 5.12 et 5.13, les deux seuils 5000 (pour le *Spectral Roll-Off*) et 0.825 (pour le *Perceptual Sharpness*) sont respectivement choisis. Ces deux

seuils ont été choisis dans l'objectif de minimiser le nombre d'erreurs de classification des deux classes, mais les valeurs choisies sont arbitraires. On aurait pu, en fait, choisir n'importe quelle autre valeur tout en respectant le critère fixé au départ. Il existe donc pratiquement un nombre infini de manières pour séparer les deux classes.

Pour nombre de problèmes, un système de décision pourrait effectivement être construit en se servant de règles établies par des experts. Les règles d'experts concernent souvent des attributs à valeurs nominales (couleur, sexe, fonction, etc.) ou des attributs discrets (nombre de pièces, nombre d'enfants, etc.), mais ils peuvent également s'appliquer aux attributs continus (salaire, prix, distance, etc.). Cette procédure a toutefois ses limites. D'après [Waterman, 1986], un expert a tendance à exprimer les conclusions qu'il a tirées en observant les données, en se servant de termes généraux, jugés trop vastes pour une exploitation efficace en utilisant une machine. Quinlan [Quinlan, 1987] met en évidence l'intérêt d'utiliser des arbres de décision pour remplacer les règles d'experts. L'accent y est mis sur l'efficacité et l'économie de la représentation sous forme d'arbre de décision. L'auteur n'exclut toutefois pas le recours aux experts pour définir un cadre pour les concepts importants, complétés éventuellement par une liste d'exemples. Les règles peuvent ainsi être automatiquement dérivées et exprimées via des arbres de décision. La sous-section suivante introduit les arbres de décision et discute une possible application à notre problème.

5.2.5 Arbres de Décision

Les arbres de décision [Breiman et al., 1984] [Quinlan, 1986] [Brodley and Utgoff, 1995] sont une méthode de classification supervisée largement utilisée en exploration de données, en statistiques et en reconnaissance des formes. Plusieurs algorithmes de classification populaires (C4.5, ID3, CART, GID3*, etc. [Quinlan, 1986] [Quinlan, 1993] [Breiman et al., 1984] [Fayyad, 1992] [Gelfand et al., 1991]) sont basés sur des arbres de décision, et ils diffèrent souvent dans leur procédure de construction d'arbre. L'objectif d'un arbre de décision est d'extraire automatiquement, à partir des données observées, les règles qui permettent de les représenter sous une forme compréhensible par les humains. Pour peu que l'arbre soit « lisible », il pourrait être vu comme une « représentation de la connaissance » [Quinlan, 1987]. Cela n'est pas toujours le cas, notamment pour des données ayant un nombre élevé d'attributs. Dans le texte qui suit, on se réfère indifféremment aux mots « caractéristique » et « attribut » ; les deux mots signifient un élément d'un vecteur de données. Un vecteur *MFCC* de 16 dimensions possède donc 16 caractéristiques ou attributs.

Un arbre de décision est construit en divisant les données en entrée en deux ou plusieurs sous-

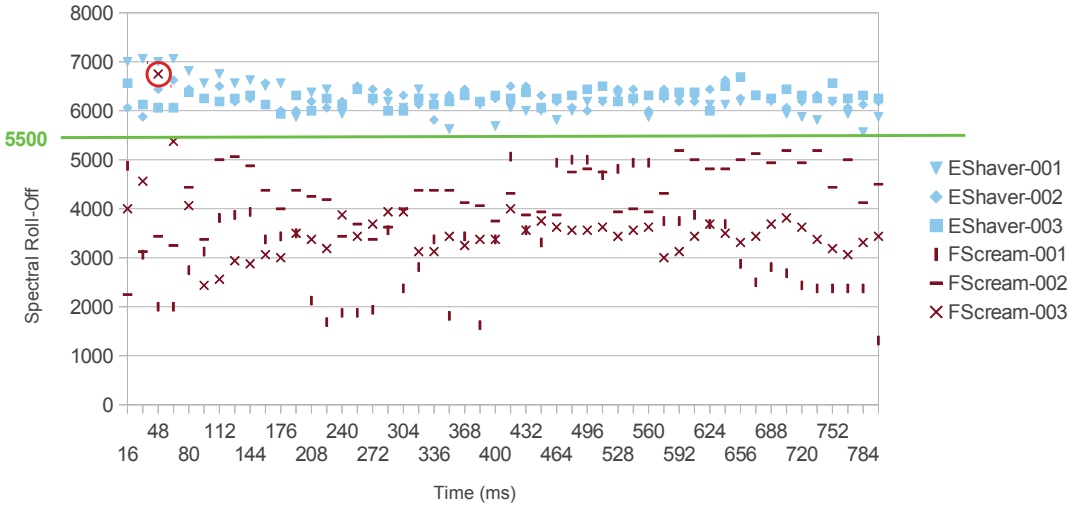

Figure 5.12: Une possible séparation linéaire entre les fenêtres des deux classes *Electric Shaver* et *Female Scream* en se basant sur le coefficient *Spectral Roll-Off*

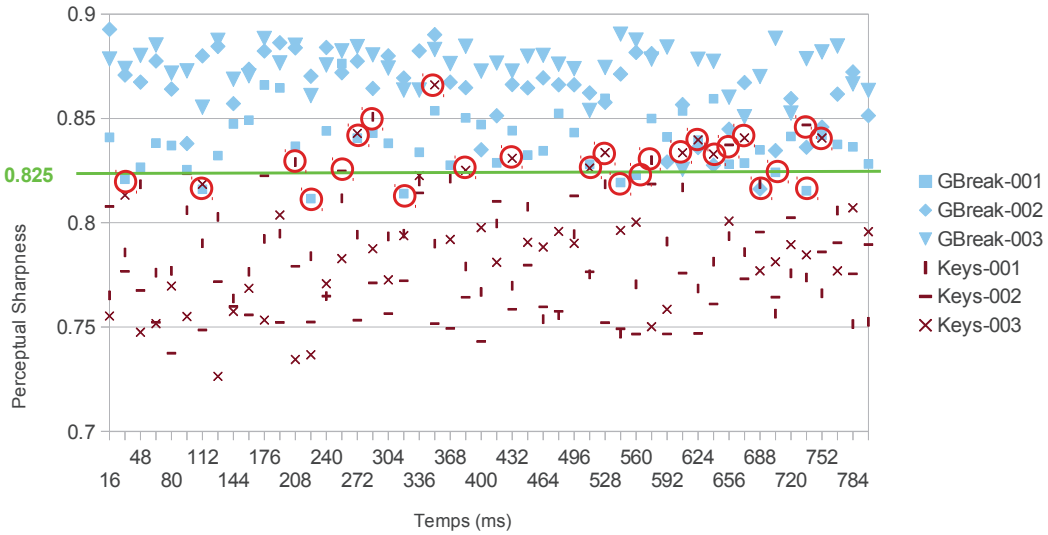

Figure 5.13: Une possible séparation linéaire entre les fenêtres des deux classes *GlassBreaking* et *Keys* en se basant sur le coefficient *Perceptual Sharpness*

ensembles, en se basant sur la valeur d'un attribut. Par la suite, chacun des sous-ensembles obtenus est à son tour divisé en se basant sur la valeur d'un autre attribut. Un sous-ensemble est considéré comme un nœud final si tous les exemples qu'il contient appartiennent à une seule classe, ou si, en utilisant une fonction d'évaluation, aucune autre division n'est plus possible. Ce processus est appelé partitionnement récursif [Breiman et al., 1984].

Il existe, en effet, plusieurs façons de créer un arbre de décision. Une fonction de gain est souvent utilisée pour décider, à chaque itération, du prochain attribut à utiliser pour le partitionnement. L'approche la plus utilisée est basée sur un **algorithme glouton** (*Greedy Algorithm*) qui consiste à choisir l'optimum local parmi les choix disponibles à chaque itération. Ce processus descendant est appelé TDIDT (*Top-Down Induction of Decision Trees*) [Quinlan, 1986].

Le tableau 5.4 contient des exemples de données appartenant à deux classes X et Y. Le nombre d'attributs est de 3, un attribut à valeurs nominales (A), un attribut à valeurs continues (B) et un attribut à valeurs discrètes (C). Un façon de construire un arbre de décision pour ces données est présentée à la figure 5.14. Dans cet exemple, la stratégie empruntée pour construire l'arbre a pour objectif d'arriver à des nœuds finaux qui ne contiennent des exemples que d'une seule classe. Un nœud final porte donc l'étiquette d'une seule classe. Cette stratégie, quelque peu simpliste, peut donner lieu à des arbres avec un nombre très élevé de nœuds finaux qui ne contiennent qu'un seul exemple. Une contrainte est souvent imposée quant au nombre minimum d'exemples par noeud final. Un nœud final représente, en principe, une seule classe. Mais dans beaucoup d'approches, chaque nœud final est associé à un vecteur de probabilités représentant la probabilité de chaque classe.

Pour les attributs à valeurs continues, les algorithmes utilisés pour créer les arbres de décision ont souvent recours à un ou plusieurs critères pour choisir les seuils. Les critères peuvent être la minimisation de l'entropie ou la maximisation de la corrélation entre l'étiquette d'une classe et un intervalle donné, etc.

Cela étant dit, la question qui se pose maintenant est la suivante : peut-on simplement utiliser des arbres de décision avec tous les coefficients pour trouver les règles qui permettent de distinguer deux classes ?

Pour des données numériques continues, un arbre de décision peut être interprété comme une collection d'hyperplans linéaires, chacun étant orthogonal sur un des axes [Brodley and Utgoff, 1995]. À la différence des SVMs, un arbre de décision incorpore un hyperplan séparateur par

Tableau 5.4: Exemples de données appartenant à deux classes

			Attribut		
			A	B	C
Classe	X	x_1	Bleu	11.8	43
		x_2	Bleu	12	50
		x_3	Rouge	10.5	54
		x_4	Rouge	9.8	47
	Y	y_1	Bleu	10.1	50
		y_2	Rouge	11.4	48
		y_3	Rouge	12.6	39
		y_4	Rouge	10.7	42

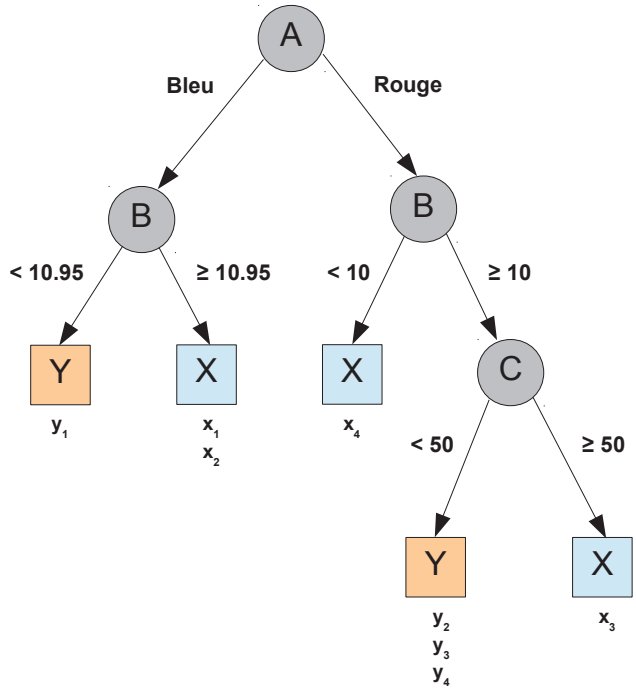

Figure 5.14: Un arbre de décision correspondant aux données du tableau 5.4

attribut ; il peut ainsi être vu comme un hyperplan « local » correspondant à un seul attribut, donc à une seule dimension. Avec les SVMs, l'hyperplan séparateur obtenu correspond à toutes les dimensions.

D'autre part, le critère utilisé pour les SVMs pour déterminer l'hyperplan est la maximisation de la marge entre les points de deux classes. Dans un arbre de décision, des critères différents sont

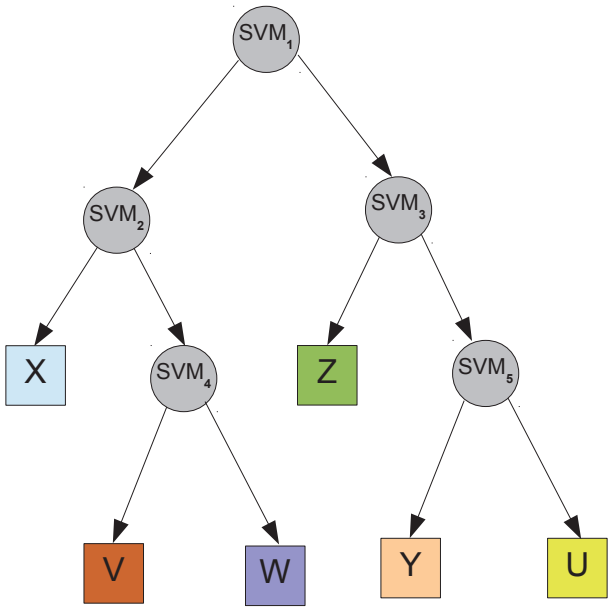

Figure 5.15: Un arbre de décision avec des SVMs incorporés

utilisés pour déterminer les hyperplans de séparation. L'approche utilisée dans les SVMs possède de très bonnes bases théoriques et a, de surcroît, prouvé son efficacité en pratique.

Enfin, les SVMs peuvent réaliser une séparation non-linéaire des données en utilisant une fonction noyau non-linéaire. Une fonction non-linéaire est vue comme une projection des données dans un espace vectoriel de très grande taille (probablement infinie), où une séparation linéaire serait possible. Cette caractéristique n'est pas disponible dans les arbres de décision. Tous ces facteurs ont donné lieu à plusieurs méthodes combinant les SVMs et les arbre de décision [Bennett and Blue, 1997] [Madjarov and Gjorgjevikj, 2009] [Madjarov and Gjorgjevikj, 2012]. L'architecture la plus répandue consiste à incorporer un modèle SVM dans chaque nœud intermédiaire d'un arbre de décision (figure 5.15). L'objectif de cette architecture étant de tirer avantage de la précision des SVMs et de la simplicité des arbres de décision [Madzarov and Gjorgjevikj, 2010], un SVM pour des données de dimension 1 étant plus simple.

À l'instar des SVMs, une possible application des arbres de décision pour notre problème de classification du son pourrait se faire de deux manières. La première consiste à les utiliser pour classifier les vecteurs acoustiques séparément et de faire appel à une stratégie d'agrégation afin de décider de la classe de la séquence toute entière. Tout comme dans SVM-frame-level (paragraphe

4.1), la stratégie d'agrégation peut très bien être un vote majoritaire. La seconde est de les utiliser après la transformation des séquences de vecteurs acoustiques en vecteurs de de grande taille, c'est-à-dire en super vecteurs comme ceux issus de la transformation du noyau SVM-GSL (paragraphe 4.1) ou en vecteurs de coefficients statistiques tels ceux vus dans la sous-section 5.2.2.

La seconde approche semble plus appropriée, d'autant plus que les SVMs que nous avons utilisés avec ces vecteurs de grande taille sont linéaires. La première, quant à elle, peut donner lieux à des temps d'exécution très élevés en raison du nombre important de coefficients que l'on envisage d'utiliser et du nombre de vecteurs. De surcroît, les vecteurs ne sont souvent pas linéairement séparables à ce niveau. Avec SVM-frame-level, il a fallu utiliser des noyaux non-linéaires pour séparer les données. Cela pourra constituer un vrai obstacle pour toute utilisation des arbres de décision.

Cependant, sur l'ensemble de nos 94 coefficients, on a pu voir que certains coefficients nous permettent très bien de faire une séparation linéaire de deux classes (figures 5.12 et 5.13). Dans les figures 5.12 et 5.13, on peut observer qu'une classe est plus représentée dans un domaine de valeurs particulier par rapport à la classe concurrente, et inversement. L'idée de base de la méthode proposée dans cette section est de trouver, pour chaque paire de deux classes, et pour chaque caractéristique (de n'importe quelle famille de coefficients), un ensemble d'intervalles de sorte que, dans chaque intervalle, une classe est plus probable que l'autre. Certes, les temps de calcul requis pour déterminer ces intervalles peuvent être relativement longs, mais cette procédure n'est nécessaire que pendant l'apprentissage. Une fois les intervalles déterminés, une séquence de valeurs est transformée en un seul vecteur de probabilités, dont chaque élément correspond à la probabilité de la classe dans un des intervalles.

Ce processus est réitéré pour chaque caractéristique. Les vecteurs de probabilités ainsi obtenus sont « assemblés » pour n'en faire qu'un seul vecteur, utilisé en entrée d'un SVM. Cela constitue une transformation d'une séquence de vecteurs en un seul vecteur. La probabilité est estimée en divisant le nombre d'observations de la classe appartenant à un intervalle donné par le nombre total d'observations de la classe. La figure 5.16 montre des données transformées d'après cette méthode pour le coefficient *Perceptual Sharpness*.

Dans les arbres de décision, on cherche souvent à trouver un seul point de séparation par attribut. Mais en réalité, la plage des valeurs appartenant à deux ou plusieurs classes pourra être divisée en plusieurs intervalles. Cette procédure, qui consiste à trouver plusieurs intervalles pour un attribut donné, est ce qu'on appelle **discrétisation**. Elle possède beaucoup d'applications intéressantes et

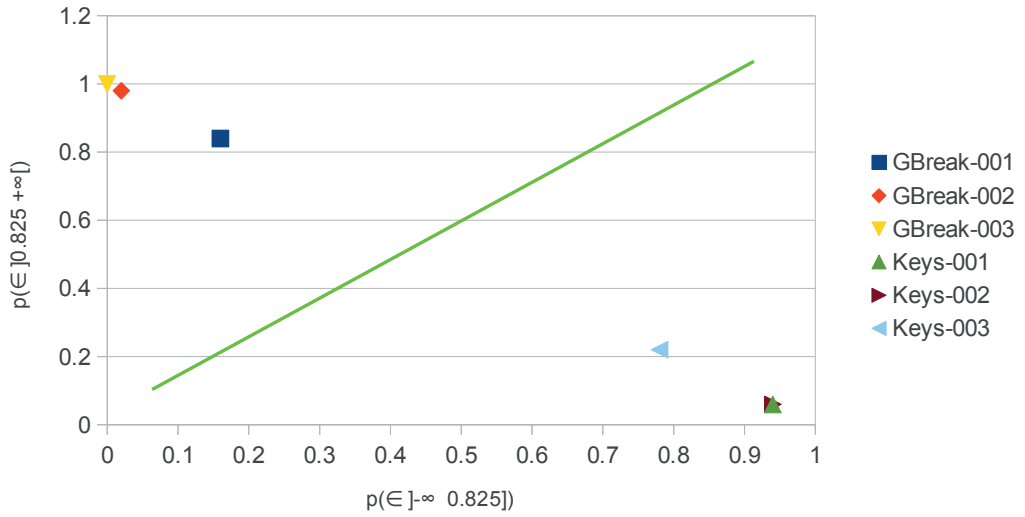

Figure 5.16: Calcul de vecteurs de probabilités basé sur le coefficient *Perceptual Sharpness*. Le domaine des valeurs est divisé en deux intervalles $(]-\infty\ 0.825]$ et $]0.825\ +\infty[)$ pour lesquels la probabilité d'appartenance d'une classe est calculée.

est, dans beaucoup d'algorithmes de classification, utilisée comme une étape de pré-traitement des données [Dougherty et al., 1995]. Les valeurs continues appartenant à un seul intervalle sont remplacées par une seule valeur nominale ou discrète. Pour beaucoup d'algorithmes, elle permet d'améliorer les performances et les temps de traitement.

Pour notre problème, et suivant l'approche proposée, mentionnée plus haut, l'intérêt de la discrétisation est de trouver les intervalles qui permettent de bien distinguer les domaines de valeurs d'une classe par rapport à une autre. Autrement dit, on cherche à trouver des intervalles dans lesquels une classe est mieux « représentée » que l'autre. Les paragraphes suivants traitent du problème de discrétisation, les catégories d'algorithmes et les critères d'évaluation.

5.2.6 Discrétisation des valeurs continues

Certains d'algorithmes d'apprentissage automatique sont conçus pour traiter des données ayant des attributs à valeurs nominales ou discrètes [Michalski and Stepp, 1983]. En revanche, pour un nombre important de problèmes réels, les données sont le plus souvent dans un espace continu. Ce facteur pourrait rendre difficile, ou pour le moins inefficace, l'utilisation de ces algorithmes. Une façon de contourner ce problème est de transformer les données continues en données discrètes, pour lesquelles chaque attribut ne peut avoir qu'un nombre limité de valeurs.

La façon la plus simple de transformer des valeurs continues en valeurs discrètes est de diviser la plage de valeurs en N intervalles. Les valeurs continues qui se trouvent dans un intervalle prennent alors toutes la même valeur (numéro de l'intervalle, son identifiant, etc.). Une telle approche ne prend pas en considération la classe des points en entrée et ne cherche, de surcroît, pas à trouver le meilleur nombre d'intervalles. Le nombre d'intervalles est souvent donné par l'utilisateur. Un bon algorithme de discrétisation se doit de trouver un bon compromis entre un nombre raisonnable d'intervalles et la perte d'information intrinsèquement liée à la transformation [Kotsiantis and Kanellopoulos, 2006]. Autrement dit, il est important d'éviter les approches qui donnent lieu à un nombre très élevé d'intervalles ne contenant que très peu de valeurs ; et celles qui, inversement, créent des intervalles contenant des exemples appartenant à plusieurs classes. Dans ce dernier cas, si les exemples sont étiquetés, il est intéressant d'avoir plusieurs intervalles qui aient chacun une bonne corrélation avec une seule classe.

Le tableau 5.5 illustre ces principes. La première ligne contient des exemples à valeurs continues appartenant à deux classes X et Y. La seconde ligne montre un choix possible des intervalles pour la discrétisation. L'objectif de ce choix est de créer des intervalles « purs » qui ne contiennent des exemples que d'une seule classe. Dans la troisième ligne, on cherche plutôt à réduire le nombre d'intervalles tout en faisant en sorte que dans chaque intervalle, une seule classe soit plus représentée. Suivant un critère ou l'autre, le nombre d'intervalles obtenus est de 9 et 2 respectivement.

Tableau 5.5: Exemple montrant deux façons différentes de choisir les intervalles. Les exemples, appartenant à deux classes X et Y, sont triés de la plus petite à la plus grande valeur.

y_1	x_1	x_2	y_2	x_3	x_4	x_5	y_3	x_6	y_4	y_5	y_6	x_7	y_7
0	1		2	3			4	5	6			7	8
0							1						

Le processus de discrétisation commence souvent par créer une liste triée des valeurs en entrée. Les

intervalles sont ensuite déterminés par une liste de seuils T_i ($i = 1 \ldots k-1$), tel que k est le nombre d'intervalles. Le nombre de seuils possibles étant en pratique infini, une fonction d'évaluation est souvent utilisée pour en déterminer les « meilleurs ». Avant de poursuivre avec avec le processus de discrétisation et les catégories d'algorithmes, nous allons donner quelques définitions qui seront utilisées tout au long des sections suivantes.

5.2.6.1 Définitions

Un attribut : (appelé aussi caractéristique, ou variable), est un élément d'un vecteur de données. Sa valeur peut être nominale, discrète ou continue. Un vecteur de dimension D en contient donc D. L'objectif de la discrétisation est de transformer les valeurs continues d'un attribut en valeurs discrètes, c'est à dire de limiter le nombre de valeurs que peut prendre l'attribut.

Une instance : (appelée aussi exemple, vecteur ou point), est une collection de D valeurs de D attributs différents. Une instance n'appartient qu'à une seule classe (un vecteur *MFCC*, par exemple), bien qu'il puisse y avoir deux instances égales appartenant à deux classes différentes. L'ensemble de toutes les instances est appelé S tel que $|S| = N$. La valeur d'un attribut A pour une instance s est notée : $\text{val}_A(s), s \in S$.

Un point de découpage : d'un attribut A, est toute valeur T_i appartenant à l'intervalle $I_A = [\text{val}_A(s); \text{val}_A(t)]$, tels que s et t sont les instances avec la plus petite et la plus grande valeur de A respectivement. Chaque point de découpage divise l'intervalle I_A en deux intervalles, $[\text{val}_A(s); T_i]$ et $]T_i; \text{val}_A(t)]$ qui contiennent chacun au moins une instance. Un attribut peut avoir plusieurs points de découpage.

L'arité : désigne le nombre total d'intervalles engendrés par les points de découpage d'un attribut A. L'arité vaut k pour un nombre de points de découpage égal à $k-1$. Il s'agit souvent d'une valeur difficile à déterminer. Plusieurs techniques peuvent être utilisées pour trouver une valeur appropriée.

Pour résumer cela, si $\biguplus_{i=1}^{k} S_i$ est un partitionnement de S en k ensembles disjoints et non-vides, en utilisant les points de découpages $T_1 \ldots T_{k-1}$, alors :

$$S_i = \begin{cases} \{s \in S \mid \text{val}_A(s) \leqslant T_1\} & \text{si } i = 1, \\ \{s \in S \mid T_{i-1} < \text{val}_A(s) \leqslant T_i\} & \text{si } 1 < i < k, \\ \{s \in S \mid \text{val}_A(s) > T_{k-1}\} & \text{si } i = k \end{cases}$$

5.2.6.2 Typologie des méthodes de discrétisation

Plusieurs critères sont utilisés pour catégoriser les méthodes de discrétisation [Dougherty et al., 1995] [Liu et al., 2002]. Dans cette section nous verrons les critères les plus courants, un aperçu de quelques fonctions d'évaluation de la qualité d'un intervalle et les critères d'arrêt les plus utilisés. Plus de détails sur ces deux derniers éléments sont donnés dans les sections suivantes, traitant des algorithmes de discrétisation. Les critères portent principalement sur les éléments suivants : étiquetage des données (classe des instances connue ou inconnue), la direction du traitement (descendant ou ascendant), la dynamicité de l'algorithme (statique ou dynamique), et sa portée (global ou local). D'après ces points de vue, les catégories de méthodes sont :

Supervisée *vs* non-supervisée : La classe des exemples en entrée peut être connue et utilisée par l'algorithme de discrétisation, on parle alors de méthode supervisée. Dans le cas opposé la méthode est dite non-supervisée. En pratique, les méthodes supervisées donnent de meilleurs résultats pour les algorithmes de classification [Dougherty et al., 1995], elles doivent donc être utilisées si l'information quant à la classe des exemples est connue. Si aucune information sur la classe n'est disponible, le seul choix est d'utiliser des méthodes non-supervisées. Les méthodes non-supervisées ne sont, de plus, pas très citées dans la littérature. Parmi les méthodes les plus connues on trouve celle qui consiste à créer des intervalles à largeur égale (*Equal Width Interval*) ou bien celle qui cherche à placer le même nombre d'instances par intervalle (*Equal Frequency Interval*). Dans la première approche, I_A est divisé en k intervalles de même largeur δ, tel que $\delta = \frac{\max(\text{val}_A) - \min(\text{val}_A)}{k}$ (k étant fourni par l'utilisateur et non pas déterminé par l'algorithme). Le grand inconvénient de cette méthode est sa sensibilité aux valeurs extrêmes. Si $\max(\text{val}_A)$ ou $\min(\text{val}_A)$ s'éloignent largement des autres valeurs, on pourrait avoir des intervalles qui ne contiennent aucun ou très peu d'exemples. Dans la seconde méthode, les points de découpage sont choisis de sorte que les intervalles qui en résultent contiennent le même nombre d'exemples. De ce fait, deux valeurs égales de A pourraient se trouver dans deux intervalles différents.

Directe *vs* incrémentale : Les méthodes de discrétisation directes divisent l'intervalle I_A en k intervalles en répondant à un ou plusieurs critères pour choisir les points de découpage. Le nombre d'intervalles k est connu à l'avance et n'est pas déterminé par l'algorithme. Les deux méthodes précédentes constituent deux exemples de méthodes directes. Les méthodes incrémentales, quant à elles, déterminent le nombre d'intervalles au cours du traitement. Elles commencent souvent par diviser l'intervalle I_A en deux intervalles puis, une fonction d'évaluation est récursivement invoquée pour décider de la division d'un intervalle donné ou de la fusion de deux intervalles adjacents. Ce processus est réitéré jusqu'à ce qu'un critère d'arrêt soit satisfait.

Statique *vs* dynamique Les méthodes statiques appliquent la discrétisation de chaque attribut A indépendamment des autres attributs. Le nombre d'intervalles obtenus est de ce fait différent d'un attribut à l'autre. Les méthodes dynamiques cherchent les valeurs possibles de k en utilisant toutes les valeurs de tous les attributs simultanément, prenant ainsi les corrélations entre les attributs en considération.

Locale *vs* globale : Les méthodes locales ne s'intéressent qu'à une partie des données en entrée à la fois. Autrement dit, toutes les instances ne sont pas prises en compte pour discrétiser un attribut. Les méthodes globales utilisent tout l'espace des données en entrée pour la discrétisation d'un attribut.

Descendante *vs* ascendante : Les méthodes descendantes commencent par une simple division de I_A en deux intervalles. D'autres points de découpage sont déterminés successivement jusqu'à ce que le nombre désiré d'intervalles soit atteint ou que plus aucun intervalle ne puisse être divisé. Elles sont considérées comme des méthodes de « division » d'intervalles. Les méthodes ascendantes, également dites méthodes de « fusion » d'intervalles, commencent par la création d'un nombre initial d'intervalles, qui seront ensuite fusionnés (deux ou plusieurs intervalles à la fois) en se basant sur une fonction d'évaluation. Le nombre initial d'intervalles peut très bien être celui des instances en entrée. Dans ce cas, chaque intervalle correspondra à une seule instance, et on pourrait ainsi obtenir plusieurs intervalles identiques. Une version améliorée de cette approche d'initialisation consiste à placer toutes les instances égales dans un seul intervalle.

Les méthodes de discrétisation ont souvent besoin d'une fonction d'évaluation pour évaluer les points de découpage, c'est-à-dire, pour savoir si un intervalle donné doit être divisé, ou si plusieurs

intervalles doivent plutôt être fusionnés pour répondre à un ou plusieurs critères fixé(s) au départ. D'autre part, pour contrôler le nombre d'intervalles obtenus, et éviter d'avoir des intervalles contenant très peu ou trop d'exemples, un ou plusieurs critères d'arrêt sont utilisés. Les deux paragraphes qui suivent traitent de ces deux notions très importantes.

Fonctions d'évaluation : Une fonction d'évaluation sert à évaluer la « qualité » d'un intervalle, et donc d'un point de découpage. Dans la méthode *Equal Width Interval* par exemple, seule importe la taille des intervalles obtenus, qui doit être uniforme. Pour l'autre méthode non-supervisée, *Equal Frequency Interval*, les intervalles doivent contenir le même nombre d'exemples, qu'elle que soit la différence en taille des intervalles qui en résulteront.

Des fonctions plus complexes sont souvent employées dans les méthodes supervisées. Parmi les critères les plus utilisés on trouve : la minimisation de l'entropie, la maximisation de la dépendance entre l'étiquette d'une classe et un intervalle donné et la performance (précision) de l'algorithme de classification qui utilisera les données discrétisées en entrée.

L'entropie de Shannon d'une variable X est donnée par l'équation 5.5 [Shannon, 1948] [Shannon and Weaver, 1949] :

$$\text{Ent}(X) = -\sum_{x} p_x \log(p_x) \tag{5.5}$$

où x est une valeur de X et p_x sa probabilité estimée. $\text{Inf}(x)$ est la quantité moyenne d'information par événement, donnée par l'équation 5.6 :

$$\text{Inf}(x) = -\log(p_x) \tag{5.6}$$

L'information est élevée pour les événements rares et petite pour les événements les plus probables. De ce fait, l'entropie $\text{Ent}(X)$ est la plus élevée si les événements possibles sont équiprobables ($p_{x_i} = p_{x_j}$ pour tout i, j) et la plus petite si $p_{x_i} = 1$ et $p_{x_j} = 0$ pour tout $j \neq i$. Des exemples de méthodes utilisant une fonction d'évaluation basée sur l'entropie sont : C4.5, ID3, D2, Fayyad et Irani, Mantaras [Quinlan, 1993] [Quinlan, 1986] [Catlett, 1991] [Fayyad and Irani, 1993] [Cerquides and Màntaras, 1997].

L'objectif de certaines méthodes est d'augmenter la corrélation entre une classe et la valeur d'un attribut. Le but est donc de produire des intervalles dont les valeurs correspondent plus à une classe donnée qu'aux autres classes. Dans le méthode Zeta [Ho and Scott, 1997] [Ho and Scott, 1998], la dépendance entre une classe et une valeur discrète de l'attribut est définie par le meilleur taux de précision qu'on peut avoir si une valeur de l'attribut est utilisée pour « prédire » la classe de l'exemple. D'autres méthodes (ChiMerge, Chi2, ConMerge etc. [Kerber, 1992] [Liu and Setiono, 1995] [Wang and Liu, 1998]) sont basées sur la mesure du χ^2. Un test de signification statistique est effectué entre la valeur de l'attribut et une classe donnée. La représentation de la classe dans un intervalle doit montrer une certaine cohérence, sinon l'intervalle est divisée pour mieux exprimer cette cohérence. De surcroît, deux intervalles adjacents ne doivent pas avoir les mêmes fréquences de classes car, dans ce cas, ils doivent être fusionnés.

Enfin, dans les méthodes basées sur la précision, l'objectif est d'améliorer les résultat obtenus par un algorithme donné. Plusieurs partitionnements sont alors testés et le meilleur est retenu. Ce processus peut être très long car, à chaque itération, un nouvel apprentissage de l'algorithme doit être effectué. [Chan et al., 1991] présentent une exemple de méthode basée sur la précision.

Critères d'arrêt : Le nombre d'intervalles obtenus après la discrétisation, k, doit en principe être beaucoup plus petit que le nombre des exemples en entrée. Plusieurs critères d'arrêt sont alors utilisés par les méthodes de discrétisation pour éviter de produire un nombre élevé d'intervalles. La façon la plus simple de contrôler k est de le choisir de manière statique, avant le lancement du processus de discrétisation. Cette approche peut s'avérer inefficace car, d'une part, si k n'est pas suffisamment grand, certains des intervalles obtenus pourraient contenir un nombre élevé d'exemples de plusieurs classes. Cela augmentera l'entropie et supprimera toute corrélation entre un intervalle donné et une des classes. D'autre part, si k est trop grand, les intervalles obtenus pourraient contenir un nombre très réduit d'exemples. Par ailleurs, certains intervalles adjacents pourraient avoir la même distribution de classes ; ils devraient, en principe, être fusionnés.

D'autres méthodes sont basées sur une **fonction de gain**. Tant qu'il y a un gain à diviser un nouvel intervalle, l'algorithme se poursuit. Pour éviter de créer des intervalles à un nombre très réduit d'instances, certaines méthodes imposent un nombre minimal d'instances par intervalle [Holte, 1993]. Plusieurs méthodes utilisent le principe de Longueur de Description Minimale (*MDLP* ou *Minimum Description Length Principle*). Ces deux concepts sont expliqués dans les sous-sections suivantes traitant de quelques méthodes de discrétisation.

La figure 5.17 (adaptée de [Liu et al., 2002]) illustre les étapes d'un processus de discrétisation typique.

5.2.7 Méthodes de discrétisation

Dans cette section nous présentons quelques méthodes de discrétisation parmi les plus utilisées. Comme nous venons juste de le voir, plusieurs critères peuvent être utilisés pour classer les différentes méthodes. L'utilisation d'un critère mène à une catégorisation différente de celle qu'on pourrait obtenir avec un autre critère. Autrement dit, deux méthodes peuvent être dans la même catégorie selon le critère C_i et dans deux catégories différentes d'après un autre critère C_j. Par exemple la méthode Zeta et ChiMerge ont toutes les deux une fonction d'évaluation avec une tendance à augmenter la dépendance entre une classe et un intervalle. Elles seront, en revanche, placées dans deux catégories différentes si on s'intéresse à la direction du traitement. La méthode Zeta est basée sur la division successive d'intervalles, elle est de ce fait une approche descendante. ChiMerge est une approche ascendante car elle est basée sur la fusion d'intervalles.

Dans ce qui suit on s'appuiera sur les fonctions d'évaluation pour présenter les différentes méthodes étudiées. De surcroît, et compte tenu de la nature de notre problème, dans lequel les données sont étiquetées, on ne s'intéressera qu'aux méthodes supervisées. Nous présentons, dans cet esprit, une méthode basée sur un simple *binning* supervisé ; c'est à dire, contrairement aux méthodes *Equal Width Interval* et *Equal Frequency Interval*, elle utilise la classe des exemples pour pour créer les intervalles. Les autres méthodes présentées sont soit basées sur la minimisation de l'entropie, soit sur la maximisation de dépendance entre une classe et la valeur d'un attribut.

5.2.7.1 Méthodes de *binning*

1Rule Discretizer : L'idée de base de la méthode *1Rule Discretizer* (ou 1RD) [Holte, 1993] est de construire, pour un attribut donné, des intervalles purs, qui ne contiennent des exemples que d'une seule classe. Cette stratégie peut, toutefois, donner lieu à des intervalles qui ne contiennent qu'un seul exemple. Pour éviter ce problème, un nombre minimum d'exemples par intervalle est imposé. En se basant sur une analyse empirique de plusieurs tâches de classification, [Holte, 1993] suggère de fixer ce nombre à 6 (le tout dernier intervalle pourra tout de même contenir moins de 6 exemples). Cela implique que certains intervalles contiendront probablement des exemples venant de plusieurs classes. Ces intervalles sont alors attribués à la classe « dominante », c'est à dire la

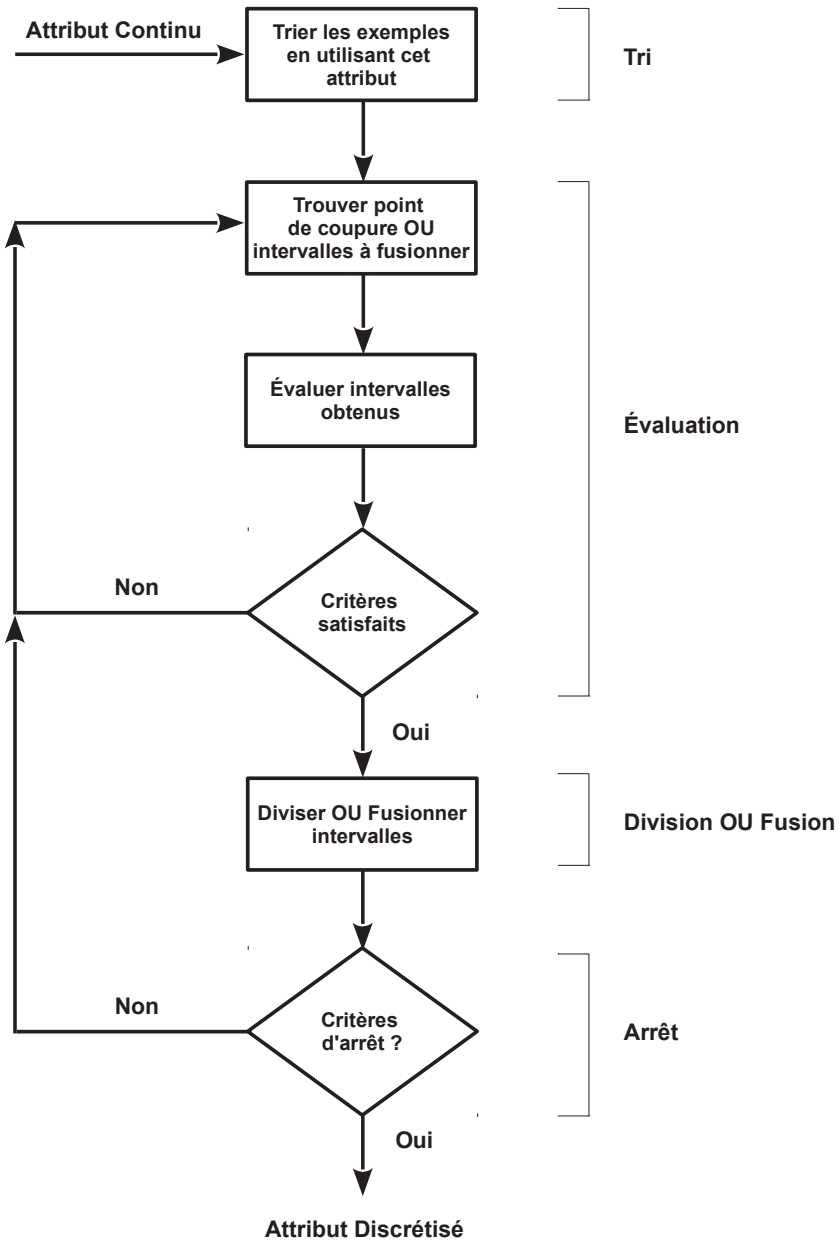

Figure 5.17: Un processus de discrétisation typique

classe la plus représentée dans l'intervalle. 1RD est vu comme une méthode de discrétisation, aussi bien qu'une méthode classification pouvant être assimilée à un arbre de décision.

5.2.7.2 Méthodes basées sur l'entropie

ID3 : ID3 [Quinlan, 1986] est une méthode utilisée pour la création d'arbres de décision. Elle utilise la minimisation de l'entropie comme mesure pour choisir un point de découpage. à chaque itération, le meilleur attribut est utilisé pour faire une discrétisation binaire. L'algorithme s'arrête lorsque tous les nœuds finaux ne contiennent des exemples que d'une seule classe. Ce critère peut toutefois être relâché pour accepter plus d'une classe par nœud final.

D2 : D2 [Catlett, 1991] est comparable à ID3 en ce qu'elle utilise également l'entropie pour choisir un point de découpage. Contrairement à ID3 qui trouve un point de découpage par nœud, D2 détermine récursivement les points de découpage potentiels tant que le critère d'arrêt ne s'est pas réalisé. Le critère d'arrêt peut être le nombre de valeurs minimal par intervalle, le nombre final d'intervalles à obtenir, ou bien le gain obtenu après discrétisation.

Méthode Fayyad et Irani : Cette méthode utilise aussi une heuristique basée sur la minimisation l'entropie. Après avoir trié les exemples de S ($|S| = N$) en se basant sur un attribut A, tous les points de découpage potentiels ($N-1$ points, chacun se trouvant à mi-distance entre deux valeurs de A) sont examinés. Pour chaque point de découpage T, S est divisé en deux sous-ensembles S_1 et S_2. Si le nombre de classes est m, $P(C_i, S_j)$ est la proportion des exemples de la classe C_i dans S_j, alors l'entropie de S_j est donnée par (voir équation 5.5) :

$$\text{Ent}(S_j) = \sum_{i=1}^{m} P(C_i, S) \log(P(C_i, S_j))$$

$\text{Ent}(S_j)$ est la quantité d'information (en bits) pour décrire les classes dans S_j. L'entropie de S, résultat du partitionnement de A par le point T, est la moyenne pondérée de l'entropie de S_1 et S_2 :

$$E(A, T; S) = \frac{|S_1|}{|S|}\text{Ent}(S_1) + \frac{|S_2|}{|S|}\text{Ent}(S_2) \tag{5.7}$$

Le point de découpage T_A^S retenu est celui qui minimise $E(A, T; S)$:

$$T_A^S = \arg\min_{i=1...k} E(A, T_i; S) \tag{5.8}$$

Ce processus est par la suite répété pour les sous-ensembles qui en résultent jusqu'à ce que le critère d'arrêt soit satisfait. Le critère d'arrêt utilisé ici est le *MDLP*. Le partitionnement d'un sous-ensemble S_j s'arrête si et seulement si :

$$Gain(A, T_i; S_j) < \frac{\log_2(N-1)}{N} + \frac{\Delta(A, T_i; S_j)}{N} \tag{5.9}$$

où

$$Gain(A, T_i; S_j) = \text{Ent}(S_j) - E(A, T_i; S_j)$$

et

$$\Delta(A, T_i; S_j) = \log_2(3^{m_j} - 2) - [m_j \cdot \text{Ent}(S_j) - m_j^1 \cdot \text{Ent}(S_{j2}) - m_j^2 \cdot \text{Ent}(S_{j2})]$$

5.2.7.3 Méthodes basées sur la dépendance entre une classe et la valeur d'attribut

ChiMerge : Dans cette méthode [Kerber, 1992], une initialisation est faite en créant autant d'intervalles qu'il y a de valeurs distinctes de l'attribut. Chaque deux intervalles adjacents sont par la suite évalués en utilisant la mesure du χ^2. Si deux intervalles sont indépendants de toute classe, ils sont fusionnés, autrement ils doivent rester séparés. L'algorithme s'arrête si, pour toute paire d'intervalles adjacents, la valeur de χ^2 est en dessous d'un certain seuil.

Chi2 : Chi2 [Liu and Setiono, 1995] est une version automatisée de ChiMerge. Les intervalles adjacents sont fusionnés tant que cela ne mène pas à une incohérence. L'incohérence signifie, ici, que deux instances égales sont placées dans deux intervalles différents. En se basant sur ce principe, cette méthode peut aussi être utilisée pour la sélection d'attributs. Les attributs pour lesquels il n'est

pas possible de faire une discrétisation sans observer une incohérence étant écartés.

5.2.8 Méthode retenue et application pour la transformation de séquences

Après avoir présenté quelques une des méthodes de discrétisation les plus connues, nous présentons dans cette section l'application de la méthode que nous avons retenue pour transformer une séquence de vecteurs en un seul vecteur, dans le but de l'utiliser en entrée d'un SVM.

Il n'est pas facile, en partant des définitions ci-dessus, de juger l'efficacité d'une méthode par rapport aux autres, à moins qu'on les ait toutes testées. De plus, une méthode qui peut se montrer efficace pour un problème ou des données spécifiques, peut très bien ne plus l'être pour d'autres.

Dans la littérature, on ne trouve pas beaucoup de comparaisons entre méthodes de discrétisation. Dans [Dougherty et al., 1995] et [Liu et al., 2002], les meilleures performances ont été obtenues avec la méthode de Fayyad et Irani, en utilisant plusieurs bases de données pour différents problèmes. En général, les méthodes basées sur l'entropie se sont avérées meilleures que les autres. Nous allons donc, dans notre travail, retenir cette approche et l'utiliser pour trouver les points de découpage d'un attribut.

L'idée de base de cette méthode de transformation de séquence est de trouver, pour chaque paire de classes, et pour chaque attribut (coefficient acoustique), l'ensemble des points de découpage qui divisent le domaine des valeurs de l'attribut en plusieurs intervalles (figure 5.18). Une fois les intervalles connus, un vecteur de probabilités représentant la séquence est calculé. Chaque valeur représente la probabilité de la classe par rapport à un des intervalles. La probabilité d'une classe X par rapport à l'intervalle I_i de l'attribut j est estimée par :

$$P(X, I_i^j) = \frac{|X_i^j|}{\sum_i^k |X_i^j|} \tag{5.10}$$

où X_i^j sont les instances de X dont la valeur de l'attribut j appartient au i-ème intervalle de l'attribut j, I_i^j.

Ce processus est répété pour tous les attributs. Les différents vecteurs de probabilités qui en résultent sont joints l'un à l'autre pour n'en faire qu'un seul. Celui-ci représentera la séquence et est utilisé en entrée du SVM. Pour chaque attribut, et pour chaque paire de classes, des intervalles différents

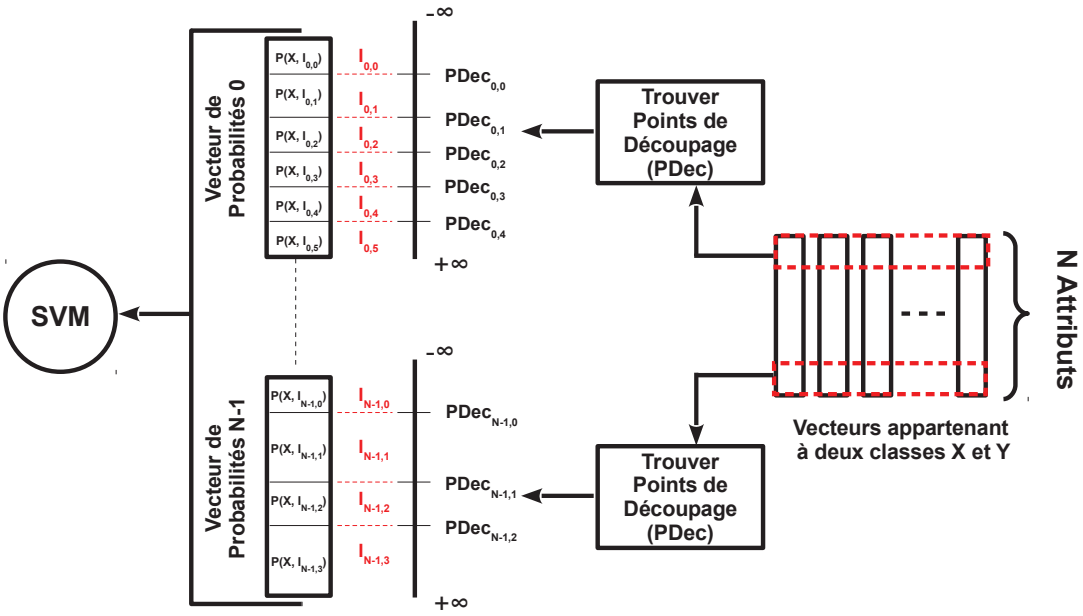

Figure 5.18: Transformation d'une séquence de vecteurs en vecteur de probabilités

sont déterminés. Pour classifier la séquence par rapport à une autre paire de classes, le vecteur de probabilité correspondant est créé et utilisé avec le modèle SVM binaire qui correspond aux deux classes.

En principe, les intervalles sont choisis par l'algorithme de sorte que, dans chacun, une classe soit plus représentée que l'autre. Cela n'est, toutefois, pas forcément le cas de toutes les classes et de tous les attributs.

5.3 Résultats expérimentaux

5.3.1 Comparaison entre les GMMs et SVM-StatVect

Cette section rapporte les résultats obtenus avec les méthodes et les familles de coefficients décrites dans ce chapitre. Pour les GMMs, quatorze familles de coefficients sont utilisées (coefficients du tableau 5.2 excepté *Envelope Shape Statistics*). Tous les coefficients, extraits depuis des fenêtres de

16ms, avec 50% de recouvrement, sont utilisés pour former un seul vecteur de 94 coefficients.

Pour les SVMs, les 15 familles de coefficients sont utilisées. Avant de calculer leurs coefficients statistiques, les valeurs de l'enveloppe d'amplitude du signal sont normalisées pour être entre 0 et 1. Enfin, comme mentionné précédemment, les coefficients du *Temporal Shape Statistics* peuvent être calculés soit sur des fenêtre de courte durée, ou bien sur la totalité du signal. Nous utilisons la première configuration pour les GMMs et la seconde pour SVM-StatVect.

La même base de données que celle utilisée dans le chapitre précédent est utilisée pour les expérimentations de ce chapitre. Ici également, un tiers de la base est utilisé pour l'apprentissage et un second tiers pour le test. Le tableau 5.6 montre les résultats obtenus avec les GMMs et SVM-StatVect. Pour les GMMs, et pour les expérimentations faites en utilisant une seule famille de coefficients, seules les familles comptant plusieurs caractéristiques par vecteur on été retenues.

Lorsque seuls les coefficients *MFCC* sont utilisés, on obtient de meilleurs résultats avec les GMMs en comparaison avec SVM-StatVect. Il en va de même pour les deux familles (*Loudness* et *Spectral Crest Factor Per Band*) utilisées seules, mais avec une différence moins importante avec SVM-StatVect. Lorsque les deux familles de coefficients *Spectral Flatness Per Band* et *Spectral Shape Statics* sont utilisées seules, SVM-StatVect se montre meilleur que les GMMs. Il n'est pas facile d'interpréter ces résultats mais les moindres performances des GMMs pour ces deux dernières familles ont probablement pour raison la corrélation entre les caractéristiques dans le cas du *Spectral Flatness Per Band* et le nombre réduit de caractéristiques pour *Spectral Shape Statics*.

Par ailleurs, le fait que les meilleures performances soient obtenues avec les GMMs par rapport au SVM-StatVect, pour les premières familles de coefficients (*MFCC*, *Loudness* et *Spectral Crest Factor Per Band*), revient certainement à la perte d'information due à la réduction de la séquence de vecteurs en un vecteur de coefficients statistiques, un modèle GMM permettant de mieux modéliser la séquence.

Cela étant dit, l'utilisation de plusieurs familles de coefficients avec des GMMs a eu pour effet, comme prévu, une baisse des performances. L'utilisation de toutes les familles de coefficients avec des GMMs ne semble pas très appropriée. Cette approche (vecteurs dont les caractéristiques proviennent de plusieurs familles de coefficients en entrée d'un GMM) ne sera pas utilisée dans le futur, notamment si d'autres familles de coefficients sont ajoutées. À l'encontre des GMMs, les SVMs, mis en oeuvre avec toutes les familles de coefficients, donnent toujours de meilleurs résultats que ceux obtenus avec une seule famille (voir la section suivante).

Tableau 5.6: Comparaison entre les performances des GMMs et de la méthode SVM-StatVect en utilisant individuellement certaines familles de coefficients, puis en utilisant plusieurs familles ensemble

Famille de Coefficients	**GMM**	**SVM-StatVect**
MFCC	**75.9**	63.2
Loudness	**70.3**	66.9
Spectral Crest F. P. B.	**67.7**	66.6
Spectral Flatness P. B.	64.1	**65.5**
Spectral Shape Stat.	46.8	**55.3**
14 Familles	72.3	-
15 Familles	-	**76.5**

5.3.2 Performances de SVM-StatVect par famille de coefficients

Pour observer les performances de chaque famille de coefficients, nous effectuons un test par famille en utilisant SVM-StatVect. Le tableau 5.7 présente les taux de reconnaissance des différentes familles, triés du meilleur au moins bon, et rappelle les dimensions des vecteurs utilisés. Comme nous l'avions déjà constaté dans le tableau 5.6, certaines familles de coefficients offrent de meilleurs résultats que les *MFCC*. Aucune famille, par contre, ne semble pouvoir remplacer toutes les autres. Autrement dit, l'utilisation de toutes les familles ensemble semble être toujours meilleure que l'utilisation d'une seule famille. Les meilleures performances ont été obtenues avec les coefficients de *Loudness*, alors que les moins bonnes sont celles obtenues avec *Temporal Shape Statistics*. Les coefficients du *Temporal Shape Statistics* sont calculés en utilisant directement en entrée les échantillons du signal, sans aucune transformation.

Les résultats présentés au tableau 5.7 sont obtenus pour la base de données, toutes classes de sons confondues. Pour une analyse plus détaillée, la section suivante rapporte les performances de chaque famille de coefficients pour chacune des classes de sons séparément.

5.3.3 Performances par famille de coefficients pour chaque classe de sons

Le tableau 5.8 présente le taux de reconnaissance de chaque classe de sons par rapport à chaque famille de coefficients, ainsi que pour toutes les familles utilisées ensemble (dernière colonne). Pour certaines classes, on peut observer que des performances égales sont obtenues avec des familles différentes. Les classes qui présentent peu de variations intra-classes (*DoorOpening*, *ElectricalShaver* et *HairDryer*) sont particulièrement bien reconnues. Certaines classes présentant

Tableau 5.7: Performances triées de SVM-StatVect utilisée séparément avec chaque famille de coefficients

Famille de Coefficients	**Dimension**	**Taux de Reco. (%)**
Loudness	168	66.9
Spectral Crest F. P. B.	133	66.6
Spectral Flatness P. B.	133	65.5
MFCC	112	63.27
Spectral Shape Stat.	12	55.3
Spectral Roll-Off	7	48.0
Spectral Flatness	7	48.0
Spectral Slope	7	45.4
Complex D. O. D.	7	44.6
Spectral Variation	7	43.7
Perceptual Sharpness	7	43.5
Perceptual Spread	7	41.8
Envelope Shape Stat.	7	38.1
Spectral Decrease	7	36.7
Temporal Shape Stat.	4	33.3
Tous les Coefficients	625	**76.5**

des variations intra-classes plus ou moins importantes (*DoorClapping*, *HandsClapping*, *Keys* et *Water*) sont également bien reconnues. Pour toutes ces classes (avec ou sans variations intra-classes), de surcroît, plusieurs familles (une seule utilisée à la fois), semblent appropriées. Bien entendu, cela reste à confirmer avec des bases de données plus riches.

Par ailleurs, pour beaucoup de classes, une seule famille de coefficients permet d'obtenir des performances supérieures ou égales à celles obtenues en utilisant tous les coefficients ensemble. Ceci suggère qu'au risque de créer des confusions, certaines familles de coefficients ne doivent pas être utilisées pour reconnaître certains types de sons. Certaines familles de coefficients qui permettent de reconnaître parfaitement certaines classes (*Spectral Slope* pour les claquements de portes ou bien *Loudness* pour le bruit de clés, par exemple) ne sont , en fait, pas efficaces pour d'autres classes.

5.3.4 Résultats de la sélection de caractéristiques avec SVM-StatVect

D'après le tableau 5.8, les taux de reconnaissance de certaines classes de sons, plutôt bons avec une seule famille de coefficients, baissent dès lors qu'on utilise toutes les familles. L'objectif de cette sélection est de choisir automatiquement, pour chaque couple de deux classes de sons, les coefficients qui permettent de mieux les discriminer.

Tableau 5.8: Performances de SVM-StatVect pour chaque famille de coefficients/classes de sons. Pour une meilleure lisibilité, l'intensité du vert indique le taux de bonne reconnaissance

	Famille de Coefficients															
Son ↓	MFCC	Loudness	Spectral Crest F. P. B	Spectral Flatness P. B.	Complex D. O. D.	Perceptual Sharpness	Perceptual Spread	Spectral Roll-Off	Spectral Decrease	Spectral Flatness	Spectral Variation	Spectral Slope	Spectral Shape Stat.	Temporal Shape Stat.	Envelope Shape Stat.	Toutes les Familles
Breath	0.47	0.71	0.65	0.53	0.71	0.53	0.06	0.59	0.24	0.41	0.59	0.18	0.12	0	0	0.76
Cough	0.38	0.24	0.52	0.71	0	0	0.1	0.1	0	0	0.1	0.1	0.1	0	0	0.48
Dishes	0.52	0.61	0.64	0.64	0.55	0.36	0.24	0.48	0.48	0.45	0.27	0.39	0.48	0.24	0.36	0.85
DClapp	0.92	0.97	0.92	0.92	0.79	0.92	0.84	0.79	0.76	0.74	0.74	1	0.92	0.79	0.66	1
DOpen	0.57	0.86	0.86	0.86	0.86	0.43	0	0	0	0	0.57	0	0.43	0	0	0.86
EShaver	0.67	0.67	0.71	0.71	0.67	0.67	0.67	0.57	0.67	0.62	0.67	0.67	0.95	0.62	0.67	0.67
FemCry	0.42	0.42	0.5	0.58	0	0.17	0.17	0.25	0	0.42	0.17	0.17	0.58	0	0	0.5
FScrm	0.58	0.62	0.5	0.62	0.21	0.17	0.25	0.21	0.04	0.04	0.21	0.04	0.46	0.17	0	0.71
GBreak	0.68	0.74	0.74	0.68	0.68	0.44	0.62	0.74	0.74	0.74	0.62	0.53	0.68	0.68	0.53	0.79
HDryer	1	1	1	0.93	0.29	0.93	0.79	1	0	0.71	0.79	0.93	1	0	0.07	1
HClapp	0.83	0.94	0.61	0.44	0.5	0.72	0.72	0.83	0.56	0.78	0.5	0.72	0.89	0	0.89	0.94
Keys	0.83	1	0.92	0.75	0	0.08	0	0	0	0.75	0	0.08	0.83	0	0	0.92
Laughter	0.59	0.41	0.53	0.47	0.35	0.29	0.06	0.24	0	0.35	0.29	0.24	0.18	0	0.12	0.65
MScrm	0.72	0.59	0.72	0.59	0.48	0.52	0.55	0.66	0.38	0.59	0.62	0.52	0.52	0.79	0.72	0.72
Paper	0.67	0.71	0.71	0.62	0.14	0.05	0.19	0.19	0.14	0.33	0.43	0.14	0.48	0	0.57	0.81
Sneeze	0.18	0.36	0.18	0.36	0.09	0	0	0.09	0	0.45	0	0.36	0.36	0	0	0.73
Water	0.44	0.61	0.5	0.67	0.72	0.67	0.94	0.56	0.94	0.44	0.39	0.94	0.28	0.94	0.78	0.61
Yawn	0.29	0.14	0.29	0.29	0	0	0	0	0	0	0.14	0	0	0	0	0.29

Les algorithmes de sélection de caractéristiques employés ici (F-Score, SVM-Wrapper et CFS, section 2.6) permettent d'évaluer chacune des caractéristiques en se servant des données d'apprentissage. Après l'élimination d'un certain nombre de caractéristiques (les caractéristiques éliminées sont différentes d'une paire de classes à l'autre), des modèles SVM sont créés en utilisant ces mêmes données d'apprentissages, réduites des caractéristiques éliminées. Pour évaluer un vecteur inconnu (contenant donc toutes caractéristiques) par rapport un modèle SVM donné, seules les caractéristiques retenues dans le modèle sont prises en considération par la fonction noyau linéaire du SVM. De ce fait, seules les caractéristiques du vecteur inconnu jugées, d'après les données d'apprentissage, les « meilleures » pour discriminer deux classes spécifiques sont utilisées pour décider de l'appartenance du vecteur à l'une ou à l'autre classe.

Pour choisir des caractéristiques à conserver pour chaque paire de classes, on pourrait utiliser un sous-ensemble de données de développement pour évaluer les modèles obtenus. Cette méthode, quoique intéressante, requiert beaucoup de temps car après l'élimination d'une ou de plusieurs caractéristiques un nouveau modèle SVM doit être créé et évalué. Nous utilisons, dans ce travail, une approche plus simple qui consiste à fixer, au départ, le pourcentage des caractéristiques à retenir. Autrement dit, les caractéristiques sont évaluées en utilisant les données d'apprentissage mais le nombre exact à conserver pour chaque paire de classes n'est pas connu car aucune évaluation des modèles SVM qui en résultent n'est effectuée. Selon cette procédure, il est, certes, possible, avec un

pourcentage très faible, d'éliminer des caractéristiques importantes pour discriminer deux classes ou bien, d'en conserver d'autres moins discriminantes avec un pourcentage élevé. Notre objectif, ici, est de montrer qu'un nombre réduit de caractéristiques pourrait donner des performances supérieures ou égales à celles obtenues avec toutes les caractéristiques. Un procédure de sélection plus intelligente devra faire partie de nos futurs travaux.

Pour ce faire, nous réalisons, pour les deux méthodes F-Score et SVM-Wrapper, des expérimentations avec plusieurs pourcentages de sélection. L'algorithme CFS, quant à lui, permet de déterminer automatiquement le nombre de caractéristiques à conserver.

Le tableau 5.9 présente les résultats obtenus. Avec F-Score, le taux de reconnaissance baisse légèrement au fur et à mesure qu'on réduit le nombre de caractéristiques. À partir de 2% (12 coefficients environ sont retenus), la différence devient importante. Avec SVM-Wrapper, en revanche, on constate une amélioration des performances pour des taux de sélection entre 15% et 30%. Rappelons que, dans chaque ligne du tableau, le pourcentage de caractéristiques à retenir est le même pour toutes les paires de classes. Cette stratégie, loin d'être idéale, mène à quelques résultats qui méritent plus d'explications. En effet, on constate avec SVM-Wrapper qu'un taux de sélection de 10% donne de moins bons résultats qu'un taux de 15%, mais aussi qu'un taux de 5%. Cela s'explique par le fait que la réduction du pourcentage de 15% à 10% a eu pour effet une baisse de performances pour certaines classe, alors que la réduction du pourcentage de 10% à 5% a, au contraire, été bénéfique pour certaines classes sans affecter grandement les autres, d'où les meilleures performance obtenues avec un taux de 5%. Enfin, CFS permet de préserver le même taux de reconnaissance en utilisant un nombre moindre de coefficients. Avec cette algorithme, le nombre de coefficients retenus varie d'un couple de classes à l'autre.

Pour les deux algorithmes F-Score et SVM-Wrapper, différents pourcentages de sélections sont testés.

5.3.5 Performances de SVM-ProbVect

Cette section rapporte les résultats obtenus avec les SVMs en utilisant la seconde approche pour la transformation de séquences. Pour la sélection de caractéristiques, seule la méthode SVM-Wrapper est retenue, étant donnés les résultats du tableau 5.9. Enfin, nous testons la des coefficients issus des deux méthodes de transformation dans un seul vecteur. Le tableau 5.10 montre les résultats obtenus. Sans sélection de caractéristiques, les deux approches donnent des performances assez

Tableau 5.9: Résultats de SVM-StatVect avec sélection de caractéristiques

		Méthode de Sélection		
		F-Score	**SVM-Wrapper**	**CFS**
Taux de Sélection (%)	100	76.55		76.27
	95	76.5	76.5	
	90	75.7	76.5	
	80	75.9	76.2	
	50	75.7	76.5	
	30	74.8	77.6	
	20	75.9	77.9	
	15	74.8	**78.5**	
	10	74.0	75.9	
	5	74.0	76.5	
	2	72.0	70.9	
	1	67.2	62.1	
	0.5	66.3	57.9	

proches. Contrairement à SVM-StatVect, la méthode SVM-ProbVect est sensible à la sélection de caractéristiques. Pour des pourcentages de coefficients de 20% ou moins, l'écart entre les deux méthodes devient de plus en plus important. L'utilisation ensemble des coefficients en provenance des deux méthodes permet d'obtenir un gain par rapport à l'utilisation d'une seule méthode, et cela reste valable pour presque tous les pourcentages de coefficients utilisés. Notons que pour la fusion, les caractéristiques issues des deux méthodes sont normalisées pour être dans le même rang de valeurs.

Tableau 5.10: Comparaison de SVM-StatVect , SVM-ProbVect et la fusion des deux méthodes avec plusieurs taux de sélection de coefficients avec SVM-Wrapper

		Provenance des coefficients		
		SVM-StatVect	**SVM-ProbVect**	**SVM-StatVect + SVM-StatVect**
Taux de Sélection (%)	100	76.5	76.2	**79.0**
	50	76.5	76.5	77.6
	20	77.9	75.1	**79.0**
	15	78.5	72.5	78.5
	10	75.9	73.4	77.6
	5	76.5	70.0	77.6
	2	70.9	60.1	70.9
	1	62.1	49.7	62.1
	0.5	57.9	39.8	57.6

5.4 Conclusions et Perspectives

Dans ce chapitre, nous avons utilisé plusieurs types de coefficients acoustiques pour la reconnaissance des sons de l'environnement. L'objectif est d'exploiter les différences entre certaines classes de sons pour rendre la classification plus simple et/ou plus efficace que celle basée sur un seul type de coefficients (*MFCC*).

Pour utiliser conjointement plusieurs familles de coefficients, en se basant sur les SVMs comme méthode de classification, deux méthodes pour la transformation de séquences ont été expérimentées. La première méthode, SVM-StatVect, transforme une séquence en un seul vecteur en calculant plusieurs coefficients statistiques par attribut. La seconde méthode, SVM-ProbVect, se sert d'une méthode de discrétisation afin de trouver, pour chaque attribut, les intervalles de valeurs où une classe de sons donnée est plus probable que la classe en concurrence. La séquence est transformée en un seul vecteur en calculant sa probabilité par rapport à chaque intervalle.

Une première comparaison avec les *MFCC* utilisés seuls montre que SVM-StatVect donne des résultats bien inférieurs à ceux d'une utilisation classique des coefficients *MFCC* avec des GMMs. Toutefois les performances des GMMs baissent si plusieurs familles de coefficients sont utilisées ensemble. Les performances de SVM-StatVect, par contre, augmentent substantiellement avec l'utilisation de toutes les familles.

Les différentes familles de coefficients offrent des performances différentes selon les classes de sons mais, pour les classes de notre base, aucune famille ne semble en mesure de remplacer toutes les autres familles pour les différentes tâches de classification. L'utilisation de toutes les familles ensemble donne de meilleurs résultats. Cela se comprend en observant le comportement de chaque famille de coefficients par rapport à chaque classe de sons. Pour chaque classe, nous avons pu constater qu'il existe une ou plusieurs familles qui permet(tent) de mieux la reconnaître.

La sélection de caractéristiques issues de SVM-StatVect montre que peu de coefficients sont effectivement nécessaires pour obtenir des performances supérieures ou égales à celles obtenues avec toutes les caractéristiques. La méthode de sélection SVM-Wrapper a donné de meilleurs résultats que celles basées sur le F-Score ou CFS.

L'utilisation des vecteurs de probabilités avec des SVMs (méthode SVM-ProbVect) donne des résultats assez comparables à ceux de obtenus avec SVM-StatVect, mais elle est moins robuste

quant à la sélection de caractéristiques. Enfin, la fusion des coefficients en provenance des deux méthodes de transformation donne un gain par rapport aux deux méthodes utilisées seules.

Les résultats restent, bien évidemment, à confirmer avec des bases de sons plus importantes.

En perspective, étant donnés les résultats obtenus ici, nous croyons que des méthodes plus sophistiquées pour combiner plusieurs familles de coefficient, pourraient améliorer les performances de classification. Une combinaison de plusieurs classifieurs, chacun basé sur une famille de de coefficients, semblerait une voie intéressante.

D'autres types de familles acoustiques peuvent être étudiées, en utilisant probablement des bases de données plus riches en termes de classes de sons et de nombre de sources par classes.

Enfin, d'autres méthodes de transformation de séquences sont également à tester pour transformer les séquences de vecteurs contenant des coefficients de plusieurs familles.

CHAPITRE 6

CONCLUSIONS ET PERSPECTIVES

De tous les sons qui garnissent le paysage sonore qui nous entoure, la parole est celui qui a le plus été étudié et compris. La quantité des travaux et les résultats obtenus en Reconnaissance Automatique de la Parole (RAP) et en Reconnaissance Automatique du Locuteur (RAL) témoignent de d'intérêt dont bénéficie la parole auprès des chercheurs.

Cependant, il y a bien plus que de la parole dans le paysage sonore de la vie courante et les humains en sont pleinement conscients. Un éternuement dans un métro ou dans un bus, par exemple, ne passerait pas inaperçu (ou plutôt inaudible) pendant une période où les médias s'enthousiasment pour la moindre nouvelle concernant LA grippe du moment.

Le problème de reconnaissance des sons de l'environnement s'est posé depuis de nombreuses années mais il reste bien moins exploré que d'autres domaines de l'Intelligence Artificielle. Dans cette thèse, nous nous sommes intéressés à la Reconnaissance des Événements Acoustiques (REA) dans un contexte domotique, plus précisément dans la Maison Intelligente d'une personne âgée vivant seule.

La thèse a atteint son objectif principal en fournissant des solutions pour la séparation de la parole des autres sons de la vie courante, et la reconnaissance des sons de l'environnement. Ces solutions sont considérées comme une composante essentielle dans le système d'analyse audio du projet SWEET-HOME.

Les solutions fournies pour le projet SWEET-HOME s'inspirent largement des méthodes utilisées en reconnaissance du locuteur et sont, de surcroît, basées sur les coefficients *MFCC*, qui, eux, viennent de la reconnaissance de la parole.

Par la suite, nous nous sommes intéressés à l'étude de plusieurs familles de coefficients dans la perspective de trouver des coefficients plus efficaces et/ou moins complexes que les *MFCC*, qui soient plus appropriés à certaines classes de sons. Les résultats obtenus montrent que, effectivement, certaines classes sont plus discriminables d'une ou de plusieurs autres classes, en se basant sur une famille de coefficients spécifique, différente des *MFCC*.

Les deux sections suivantes récapitulent, respectivement, les travaux réalisés dans cette thèse ainsi que nos perspectives pour des travaux futurs.

6.1 Résumé des travaux et conclusions

Une étude de la littérature de la reconnaissance des événements acoustique nous a permis de constater un certain nombre de faits. En l'absence de méthodes de référence pour le domaine, un bon nombre de travaux sont basés sur les techniques utilisés en RAP, en RAL et en reconnaissance de la musique. D'autre part, certains techniques, en particulier les filtres auditifs, initialement proposés pour la RAP mais bien moins utilisés que les *MFCC*, ont été repris dans certains travaux de REA. Ils constituent l'une des pistes les plus prometteuses du domaine. Enfin, la classification du signal audio à partir de représentations sur deux dimensions, en utilisant les techniques du traitement d'image, a récemment donné des résultats très prometteurs.

L'un des éléments les plus embarrassants concernant la littérature de la REA est probablement le fait que les différentes applications s'intéressent à des types de sons différents et utilisent, de ce fait, des techniques et des bases de données différentes. Il n'y a donc pas eu beaucoup de comparaisons entre les différents travaux.

Trois méthodes, issues de la RAL, ont été retenues pour la REA dans cette thèse :

— SVM avec des vecteurs acoustiques (SVM-frame-level)
— GMM
— SVM avec un noyau de discrimination de séquences (SVM-GSL)

La technique SVM-frame-level consiste à utiliser les vecteurs acoustiques (vecteurs *MFCC* extraits du signal d'un événement acoustique) directement en entrée d'un SVM, à classifier chaque vecteur séparément et, en utilisant une stratégie d'agrégation, à décider de la classe de l'événement inconnu. Les deux stratégies d'agrégation utilisées sont le vote majoritaire (à quelle classe le classifieur a attribué le plus de vecteurs ?) ou la somme du résultat de classification des vecteurs (quelle est la classe dont la valeur absolue de la somme est la plus grande ?). Cette méthode est très coûteuse en temps de calcul et donne de moins bons résultats par rapport aux GMMs.

Les GMMs restent une des méthodes les plus utilisée en REA. Les systèmes utilisant des GMMs sont assez rapides et leurs performances restent intéressantes. Dans cette thèse, nous avons obtenus

de meilleurs résultats avec des GMMs qu'avec la méthode SVM-frame-level.

Une des solutions utilisées pour contourner les problèmes de l'utilisation des SVMs avec des vecteurs acoustiques en entrée est l'utilisation des noyaux de discrimination de séquences. Ces noyaux ont bien été explorés en RAL et ont souvent eu des performances meilleures que celles des GMMs. Dans cette thèse, nous avons retenu le noyau SVM-GSL pour la reconnaissance des sons de l'environnement. Les résultats obtenus sont très intéressants, vu que, contrairement à ce qui se passe en RAL, la quantité de données utilisées pour créer le modèle UBM (utilisé par le noyau SVM-GSL), ainsi que les données utilisées en adaptation sont très limitées.

Cette méthode est toutefois plus coûteuse en ressources (espace en mémoire centrale et temps processeur) que les GMMs. Nous croyons que, pour des problèmes de REA n'incluant qu'un nombre limité de classes, et qui sont, de plus, facilement discriminables, une solution basée sur les GMMs devrait être considérée.

En analysant la matrice de confusion des classes de la base de sons de l'ESIGETEL, nous avons pu tirer un certain nombre de conclusions. Les classes qui présentent très peu de variabilités intra-classe, du fait que les sons ont été enregistrés dans les mêmes conditions acoustiques (même microphone, même environnement, même source de son, etc.), sont très bien reconnues. D'autres classes avec des enregistrements réalisés dans des conditions acoustiques différentes sont également bien reconnues. Pour les classes aux taux d'erreurs les plus élevés, nous avons réalisé que la source de confusion est principalement une seule autre classe.

Tous ces résultats ont été obtenus avec des coefficients *MFCC*. Cela nous mène à plusieurs conclusions. Pour une application de REA où les événements acoustiques ont lieu dans des environnements qui risquent peu de changer, et dont les sons d'intérêt sont « faciles » à discriminer, des modèles « personnalisés » devraient être utilisés, en utilisant si possible les coefficients acoustiques et/ou les méthodes de classification les moins coûteux en ressources. De façon plus générale, cette suggestion reste valable pour les applications dont les classes de sons d'intérêt sont bien discriminables. Pour les classes dont la source de confusion est une seule ou un nombre limité de classes, il faudrait trouver de meilleurs coefficients que les *MFCC*. Bien entendu, ces conclusions sont à prendre avec beaucoup de précaution car la base de données utilisée reste très petite, tant en nombre de classes, qu'en nombre d'enregistrements par classe.

Au vu des performances du noyau SVM-GSL, nous avons retenu cette méthode pour les expérimentations faites sur le corpus du projet SWEET-HOME. Les scénarios utilisés sont enregistrés dans une

maison intelligente en utilisant sept canaux audio. Quatre scénarios d'une durée de 88, 50, 72 et 63 minutes respectivement ont été utilisés. Un algorithme de détection des événements acoustiques est utilisé en amont de l'algorithme de reconnaissance.

Le canal au meilleur RSB (rapport signal sur bruit) a été utilisé pour la reconnaissance. De très bonnes performances ont été obtenues pour la séparation entre la parole et les autres sons de la vie quotidienne. Pour la reconnaissance des autres sons, 18 classes de sons de la vie courante ont été retenues. Les résultats obtenus sont très prometteurs. Pour les quatre scénarios, plus de la moitié des sons d'intérêts dans l'appartement ont été détectés et reconnus correctement.

Par la suite nous avons fait une étude sur une quinzaine de familles de coefficients acoustiques pour les exploiter en reconnaissance des sons de l'environnement. Le but de cette étude est de trouver, pour certains couples de classes du moins, des familles de coefficients qui soient plus efficaces ou bien aussi efficaces mais moins complexes que les *MFCC*, utilisés dans toutes nos expérimentation précédentes.

Nous avons, pour ce faire, utilisé deux autres méthodes (SVM-StatVect et SVM-ProbVect) de transformation de séquences de vecteurs acoustiques en un seul vecteur, différentes du noyau SVM-GSL. Les résultats obtenus montrent que, pour la plupart des classes, on peut identifier une ou plusieurs familles de coefficients qui permettent de mieux les reconnaître. Pour certaines classes, les meilleurs résultats sont obtenus en utilisant toutes les familles de coefficients ensemble. Pour d'autres, en revanche, l'utilisation d'une seule famille donne les meilleurs résultats. L'ajout des autres familles ne procure aucun gain ou, bien au contraire, augmente les confusions avec les autres classes. En de conclusion de ces expérimentations, nous croyons qu'un système de REA pourrait être bien plus efficace en utilisant des familles de coefficients acoustiques plus appropriées aux classes de sons d'intérêt, au lieu d'utiliser une seule famille généraliste, tels que les *MFCC*.

6.2 Perspectives et futurs travaux

Bases de données : Les résultats obtenus dans nos différentes expérimentations sont certes très intéressants, mais il n'est pas facile de les généraliser. Pour cela, il faudra utiliser des bases de données beaucoup plus grandes. Un nombre important de classes de sons augmentera certainement le nombre de confusions, mais permet de tirer des jugements plus pertinents quant aux classes bien reconnues. Autrement dit, cela permettrait de mieux comprendre certaines classes de sons et les

coefficients qui permettent de les discriminer des autres classes.

Coefficients : La première perspective nous conduit à la seconde. En effet, plus le nombre de classes augmente, plus les confusions entres elles sont probables. Au vu des résultats obtenus dans cette thèse, l'utilisation d'autres familles de coefficients semble une voie très importantes. L'utilisation des coefficients issus des filtres auditifs (*Gammatone* par exemple), ainsi que ceux issus des représentations bidimensionnelles du signal entrent également partie de cette perspective.

Fusion : Dans nos expérimentations utilisant plusieurs familles de coefficients (chapitre 5), nous avons utilisé soit toutes les familles de coefficients ensemble, soit une seule famille à la fois. La fusion de plusieurs familles de coefficients constitue également une bonne perspective. Une manière de faire cette fusion serait d'utiliser plusieurs classifieurs, un par famille de coefficients, et de fusionner leur sorties (logique floue, coefficients de pondération pour les différents classifieurs, etc.).

Modélisation de la variabilité intra-classe : Il s'agit d'un point très important en RAL. En ce qui concerne ce travail, nous avons observé que les classes dont tous les enregistrements ont été réalisés dans les mêmes conditions acoustiques sont très bien reconnues. La variabilité intra-classe constitue l'une des sources de confusion les plus importantes. En RAL, des méthodes efficaces ont été développées pour faire face à ce problème. Elles sont souvent utilisées avec les GMMs ou les SVMs à noyau de classification de séquences. Après les résultats intéressants obtenus avec les GMMs et SVM-GSL, nous croyons que ces méthodes constituent une bonne perspective de recherche qui complète ce travail. Toutefois, il convient de noter que de telles méthodes requièrent souvent beaucoup de données, d'où notre première perspective concernant les bases de données.

Information temporelle : Les travaux comparant les GMMs aux HMMs que nous avons analysés, montrent l'efficacité des HMMs par rapport aux GMMs. De plus, en observant les représentations visuelles du signal de certaines classes, il semble que l'information temporelle pourrait constituer un élément utilisable pour discriminer certaines classes de sons.

Informations de haut niveau : Comme nous l'avons signalé au chapitre 2, l'une des différences importantes entre la RAP et la REA est l'existence d'un modèle de langage pour la RAP. Nous

avons également parlé de l'importance du contexte (d'après les travaux de Ballas et Howard) que les humains utilisent pour distinguer les sons de l'environnement. Nous croyons que cette source d'information pourrait, sur le moyen et le long terme, être très intéressante pour la REA.

Tests plus complets : Dans les expérimentations faites sur le corpus du projet SWEET-HOME, nous n'avons pas utilisé une stratégie de rejet ni fait de tests en présence du bruit. Pour des tests plus réalistes, ces deux points devront absolument être pris en considération dans les futurs travaux.

LISTE DES PUBLICATIONS DE L'AUTEUR

Conférences Internationales avec comité de lecture et actes

- Michel Vacher, Dan Istrate, François Portet, Thierry Joubert, Thierry Chevalier, Serge Smidtas, Brigitte Meillon, Benjamin Lecouteux, **Mohamed Sehili**, Pedro Chahuara, and Sylvain Meniard. The SWEET-HOME Project : Audio Technology in Smart Homes to improve Well-being and Reliance. *In 33rd Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC 2011)*, pages 5291–5294, Boston, USA, 2011.
- **M. A. Sehili**, D. Istrate, B. Dorizzi, and J. Boudy. Daily sound recognition using a combination of GMM and SVM for home automation. In Proceedings of the 20th European Signal Processing Conference (EUSIPCO), pages 1673–1677, 2012.
- P. A. Cavalcante, **M. A. Sehili**, M. Herbin, D. Istrate, F. Blanchard, J. Boudy, and B. Dorizzi. First steps in adaptation of an evidential network for data fusion in the framework of medical remote monitoring. *In 34th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC 2012)*, pages 2044–2047, San Diego, USA, 2012.
- **Mohamed A. Sehili**, Benjamin Lecouteux, Michel Vacher, François Portet, Dan Istrate, Bernadette Dorizzi, and Jérôme Boudy. Sound environment analysis in smart home. *In International Joint Conference on Ambient Intelligence*, volume 7683, pages 208–223, Pisa, Italy, 2012.
- Michel Vacher, Pedro Chahuara, Benjamin Lecouteux, Dan Istrate, François Portet, Thierry Joubert, **Mohamed El Amine Sehili**, Brigitte Meillon, Nicolas Bonnefond, Sébastien Fabre, Roux Camille, and Sybille Caffiau. The Sweet-Home project : Audio processing and decision making in smart home to improve well-being and reliance. *In 35th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC 2013)*, Osaka, Japan, 2013.

Journaux Nationaux

- **M. A. Sehili**, D. Istrate, and J. Boudy. Primary investigations of sound recognition for a domotic application using support vector. *Annals of the University of Craiova, Series : Automation, Computers, Electronics and Mechatronics*, 7(34)(2) :61–65, 2010.

Rapports Techniques

- Rapport de soutenance de mi-parcours, Mai 2011.
- Dan Istrate, **Mohamed A. Sehili**, Michel Vacher. Rapport pour le projet SWEET-HOME. Validation des algorithmes de reconnaissance sur plateforme, Mars 2012.

BIBLIOGRAPHIE

[Abdulla, 2002] Abdulla, W. H. (2002). Auditory based feature vectors for speech recognition systems. *Advances in Communications and Software Technologies*, pages 231–236.

[Abdulla and Zhang, 2010] Abdulla, W. H. and Zhang, Y. (2010). Voice biometric feature using gammatone filterbank and ica. *International Journal of Biometrics*, 2(4) :330–349.

[Alberti, 2001] Alberti, P. W. (2001). The anatomy and physiology of the ear and hearing. *Occupational exposure to noise : Evaluation, prevention, and control*, pages 53–62.

[Allegro et al., 2001] Allegro, S., Büchler, M., and Launer, S. (2001). Automatic sound classification inspired by auditory scene analysis. In *Eurospeech, Aalborg, Denmark*.

[Andersson et al., 2010] Andersson, M., Ntalampiras, S., Ganchev, T., Rydell, J., Ahlberg, J., and Fakotakis, N. (2010). Fusion of acoustic and optical sensor data for automatic fight detection in urban environments. In *Information Fusion (FUSION), 2010 13th Conference on*, pages 1–8.

[Anniés et al., 2007] Anniés, R., Hernandez, E. M., Adiloglu, K., Purwins, H., and Obermayer, K. (2007). Classification schemes for step sounds based on gammatone-filters. In *NIPS-Workshop Music, Brain, & Cognition*. Citeseer.

[Arthur and Vassilvitskii, 2007] Arthur, D. and Vassilvitskii, S. (2007). k-means++ : The advantages of careful seeding. In *Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms*, pages 1027–1035. Society for Industrial and Applied Mathematics.

[Bahlmann et al., 2002] Bahlmann, C., Haasdonk, B., and Burkhardt, H. (2002). On-line handwriting recognition with support vector machines - a kernel approach. In *In Proc. of the 8th IWFHR*, pages 49–54.

[Bahoura and Pelletier, 2004] Bahoura, M. and Pelletier, C. (2004). Respiratory sounds classification using cepstral analysis and gaussian mixture models. In *Engineering in Medicine and Biology Society, 2004. IEMBS'04. 26th Annual International Conference of the IEEE*, volume 1, pages 9–12. IEEE.

[Baker et al., 2009] Baker, J., Deng, L., Glass, J., Khudanpur, S., Lee, C.-H., Morgan, N., and O'Shaughnessy, D. (2009). Developments and directions in speech recognition and understanding, part 1 [dsp education]. *Signal Processing Magazine, IEEE*, 26(3) :75–80.

[Ballas and Howard, 1987] Ballas, J. A. and Howard, J. H. (1987). Interpreting the language of environmental sounds. *Environment and behavior*, 19(1) :91–114.

[Barras et al., 2001] Barras, C., Geoffrois, E., Wu, Z., and Liberman, M. (2001). Transcriber : development and use of a tool for assisting speech corpora production. *Speech Communication*, 33(1) :5–22.

[Bennett and Blue, 1997] Bennett, K. P. and Blue, J. (1997). A support vector machine approach to decision trees. In *Department of Mathematical Sciences Math Report No. 97-100, Rensselaer Polytechnic Institute*, pages 2396–2401.

[Bonastre et al., 2005] Bonastre, J.-F., Wils, F., and Meignier, S. (2005). Alize, a free toolkit for speaker recognition. In *ICASSP'05, IEEE*, Philadelphia, PA (USA).

[Bourlard et al., 1996] Bourlard, H., Hermansky, H., and Morgan, N. (1996). Towards increasing speech recognition error rates. *Speech Communication*, 18(3) :205 – 231.

[Bousquet et al., 2011] Bousquet, P.-M., Matrouf, D., and Bonastre, J.-F. (2011). Intersession compensation and scoring methods in the i-vectors space for speaker recognition. In *INTERSPEECH*, pages 485–488.

[Bregman, 1990] Bregman, A. S. (1990). *Auditory Scene Analysis*. MIT Press, Cambridge.

[Bregman, 1994] Bregman, A. S. (1994). *Auditory Scene Analysis : The perceptual organization of sound*. The MIT Press.

[Breiman et al., 1984] Breiman, L., Friedman, J., Olshen, R., and Stone, C. (1984). *Classification and Regression Trees*. Wadsworth and Brooks, Monterey, CA.

[Brodley and Utgoff, 1995] Brodley, C. E. and Utgoff, P. E. (1995). Multivariate decision trees. *Mach. Learn.*, 19(1) :45–77.

[Bronkhorst, 2000] Bronkhorst, A. W. (2000). The cocktail party phenomenon : A review of research on speech intelligibility in multiple-talker conditions. *Acta Acustica united with Acustica*, 86(1) :117–128.

[Brown and Cooke, 1994] Brown, G. J. and Cooke, M. (1994). Computational auditory scene analysis. *Computer speech and language*, 8(4) :297–336.

[Burges, 1998] Burges, C. J. (1998). A tutorial on support vector machines for pattern recognition. *Data mining and knowledge discovery*, 2(2) :121–167.

[Burton, 1987] Burton, D. (1987). Text-dependent speaker verification using vector quantization source coding. *Acoustics, Speech and Signal Processing, IEEE Transactions on*, 35(2) :133–143.

[Campbell and Greated, 1994] Campbell, M. and Greated, C. (1994). *The Musician's Guide to Acoustics*. OUP Oxford.

[Campbell et al., 2002] Campbell, W. M., Assaleh, K. T., and Broun, C. C. (2002). Speaker recognition with polynomial classifiers. *Speech and Audio Processing, IEEE Transactions on*, 10(4) :205–212.

[Campbell et al., 2006a] Campbell, W. M., Campbell, J. P., Reynolds, D. A., Singer, E., and Torres-Carrasquillo, P. A. (2006a). Support vector machines for speaker and language recognition. *Computer Speech & Language*, 20(2-3) :210–229.

[Campbell et al., 2006b] Campbell, W. M., Sturim, D. E., and Reynolds, D. A. (2006b). Support vector machines using gmm supervectors for speaker verification. *Signal Processing Letters, IEEE*, 13(5) :308–311.

[Campbell et al., 2006c] Campbell, W. M., Sturim, D. E., Reynolds, D. A., and Solomonoff, A. (2006c). Svm based speaker verification using a gmm supervector kernel and nap variability compensation. In *in Proceedings of ICASSP, 2006*, pages 97–100.

[Cannon et al., 1986] Cannon, R. L., Dave, J. V., and Bezdek, J. C. (1986). Efficient implementation of the fuzzy c-means clustering algorithms. *IEEE Trans. Pattern Anal. Mach. Intell.*, 8(2) :248–255.

[Catlett, 1991] Catlett, J. (1991). On changing continuous attributes into ordered discrete attributes. In Kodratoff, Y., editor, *Machine Learning — EWSL-91*, volume 482 of *Lecture Notes in Computer Science*, pages 164–178. Springer Berlin Heidelberg.

[Cerquides and Màntaras, 1997] Cerquides, J. and Màntaras, R. L. D. (1997). Proposal and empirical comparison of a parallelizable distance-based discretization method. In *3d Int. Conference on Knowledge Discovery and Data Mining*, pages 139–142.

[Chan et al., 1991] Chan, C.-C., Batur, C., and Srinivasan, A. (1991). Determination of quantization intervals in rule based model for dynamic systems. In *Systems, Man, and Cybernetics, 1991. 'Decision Aiding for Complex Systems, Conference Proceedings., 1991 IEEE International Conference on*, pages 1719–1723 vol.3.

[Chan et al., 2009] Chan, M., Campo, E., Estève, D., and Fourniols, J.-Y. (2009). Smart homes—current features and future perspectives. *Maturitas*, 64(2) :90–97.

[Chang and Lin, 2011] Chang, C.-C. and Lin, C.-J. (2011). LIBSVM : A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2 :27 :1–27 :27.

[Chen et al., 2006a] Chen, C.-Y., Abdallah, A., and Wolf, W. (2006a). Audiovisual gunshot event recognition. In *Systems, Man and Cybernetics, 2006. SMC'06. IEEE International Conference on*, volume 6, pages 4807–4812. IEEE.

[Chen et al., 2005a] Chen, J., Kam, A. H., Zhang, J., Liu, N., and Shue, L. (2005a). Bathroom activity monitoring based on sound. In *Pervasive Computing*, pages 47–61. Springer.

[Chen et al., 2005b] Chen, J., Zhang, J., Kam, A. H., and Shue, L. (2005b). An automatic acoustic bathroom monitoring system. In *Circuits and Systems, 2005. ISCAS 2005. IEEE International Symposium on*, pages 1750–1753. IEEE.

[Chen et al., 2006b] Chen, L., Gunduz, S., and Ozsu, M. T. (2006b). Mixed type audio classification with support vector machine. In *Multimedia and Expo, 2006 IEEE International Conference on*, pages 781–784. IEEE.

[Chen and Lin, 2006] Chen, Y.-W. and Lin, C.-J. (2006). Combining svms with various feature selection strategies. In *Feature Extraction*, pages 315–324. Springer.

[Chen and Maher, 2006] Chen, Z. and Maher, R. C. (2006). Semi-automatic classification of bird vocalizations using spectral peak tracks. *The Journal of the Acoustical Society of America*, 120 :2974.

[Chu et al., 2009] Chu, S., Narayanan, S., and Kuo, C.-C. (2009). Environmental sound recognition with time–frequency audio features. *Audio, Speech, and Language Processing, IEEE Transactions on*, 17(6) :1142–1158.

[Clavel et al., 2008] Clavel, C., Vasilescu, I., Devillers, L., Richard, G., and Ehrette, T. (2008). Fear-type emotion recognition for future audio-based surveillance systems. *Speech Communication*, 50(6) :487–503.

[Cook, 2001] Cook, P. (2001). *Music, Cognition, and Computerized Sound : An Introduction to Psychoacoustsics*. MIT Press.

[Cowling and Sitte, 2002] Cowling, M. and Sitte, R. (2002). Recognition of environmental sounds using speech recognition techniques. In Wysocki, T., Darnell, M., and Honary, B., editors, *Advanced Signal Processing for Communication Systems*, volume 703 of *The International Series in Engineering and Computer Science*, pages 31–46. Springer US.

[Cowling and Sitte, 2003] Cowling, M. and Sitte, R. (2003). Comparison of techniques for environmental sound recognition. *Pattern Recognition Letters*, 24(15) :2895–2907.

[Davis and Mermelstein, 1980] Davis, S. and Mermelstein, P. (1980). Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. *Acoustics, Speech and Signal Processing, IEEE Transactions on*, 28(4) :357–366.

[Dehak, 2009] Dehak, N. (2009). *Discriminative and generative approaches for long-and short-term speaker characteristics modeling : application to speaker verification*. Ecole de Technologie Superieure (Canada).

[Dehak and Chollet, 2006] Dehak, N. and Chollet, G. (2006). Support vector gmms for speaker verification. In *Speaker and Language Recognition Workshop, 2006. IEEE Odyssey 2006 : The*, pages 1–4. IEEE.

[Dehak et al., 2011] Dehak, N., Kenny, P. J., Dehak, R., Dumouchel, P., and Ouellet, P. (2011). Front-end factor analysis for speaker verification. *Audio, Speech, and Language Processing, IEEE Transactions on*, 19(4) :788–798.

[Dellaert et al., 1996] Dellaert, F., Polzin, T., and Waibel, A. (1996). Recognizing emotion in speech. In *Spoken Language, 1996. ICSLP 96. Proceedings., Fourth International Conference on*, volume 3, pages 1970–1973. IEEE.

[Dennis et al., 2012] Dennis, J., Dat, T. H., and Chng, E. (2012). Overlapping sound event recognition using local spectrogram features with the generalised hough transform. In *INTERSPEECH*. ISCA.

[Dennis et al., 2013a] Dennis, J., Tran, H., and Chng, E. (2013a). Overlapping sound event recognition using local spectrogram features and the generalised hough transform. *Pattern Recognition Letters*, 34(9) :1085 – 1093.

[Dennis et al., 2013b] Dennis, J., Tran, H. D., and Chng, E.-S. (2013b). Image feature representation of the subband power distribution for robust sound event classification. *Audio, Speech, and Language Processing, IEEE Transactions on*, 21(2) :367–377.

[Dennis et al., 2011] Dennis, J., Tran, H. D., and Li, H. (2011). Spectrogram image feature for sound event classification in mismatched conditions. *Signal Processing Letters, IEEE*, 18(2) :130–133.

[Deutsch, 1999] Deutsch, D. (1999). *The Psychology of Music*. Academic Press Series. Academic Press.

[Devillers and Vidrascu, 2006] Devillers, L. and Vidrascu, L. (2006). Real-life emotions detection with lexical and paralinguistic cues on human-human call center dialogs. In *Interspeech*.

[Devillers et al., 2005] Devillers, L., Vidrascu, L., and Lamel, L. (2005). Challenges in real-life emotion annotation and machine learning based detection. *Neural Networks*, 18(4) :407–422.

[Doddington, 2001] Doddington, G. (2001). Speaker recognition based on idiolectal differences between speakers. In *Proc. Eurospeech*, volume 1, pages 2521–2524.

[Dougherty et al., 1995] Dougherty, J., Kohavi, R., and Sahami, M. (1995). Supervised and Unsupervised Discretization of Continuous Features. In Prieditis, A. and Russell, S., editors, *Proceedings of the Twelfth International Conference on Machine Learning*, pages 194–202.

[Duan and Keerthi, 2005] Duan, K.-B. and Keerthi, S. S. (2005). Which is the best multiclass svm method ? an empirical study. In *Multiple Classifier Systems*, pages 278–285. Springer.

[Duan et al., 2011] Duan, S., Towsey, M., Zhang, J., Truskinger, A., Wimmer, J., and Roe, P. (2011). Acoustic component detection for automatic species recognition in environmental monitoring. In *intelligent sensors, sensor networks and information processing (ISSNIP), 2011 seventh international conference on*, pages 514–519. IEEE.

[Dubnov, 2004] Dubnov, S. (2004). Generalization of spectral flatness measure for non-gaussian linear processes. *Signal Processing Letters, IEEE*, 11(8) :698–701.

[Dufaux et al., 2000] Dufaux, A., Besacier, L., Ansorge, M., and Pellandini, F. (2000). Automatic sound detection and recognition for noisy environment. In *Proc. of the X European Signal Processing Conference*. Citeseer.

[Duxbury et al., 2003] Duxbury, C., Bello, J. P., Davies, M., Sandler, M., and S, M. (2003). Complex domain onset detection for musical signals. In *In Proc. Digital Audio Effects Workshop (DAFx*.

[Eronen and Klapuri, 2000] Eronen, A. and Klapuri, A. (2000). Musical instrument recognition using cepstral coefficients and temporal features. In *Acoustics, Speech, and Signal Processing, 2000. ICASSP '00. Proceedings. 2000 IEEE International Conference on*, volume 2, pages II753–II756 vol.2.

[Eronen et al., 2006] Eronen, A. J., Peltonen, V. T., Tuomi, J. T., Klapuri, A. P., Fagerlund, S., Sorsa, T., Lorho, G., and Huopaniemi, J. (2006). Audio-based context recognition. *Audio, Speech, and Language Processing, IEEE Transactions on*, 14(1) :321–329.

[Espy-Wilson et al., 2006] Espy-Wilson, C. Y., Manocha, S., and Vishnubhotla, S. (2006). A new set of features for text-independent speaker identification. In *Proc. Interspeech*, pages 1475–1478.

[Everitt and Hand, 1981] Everitt, B. and Hand, D. (1981). *Finite Mixture Distributions*. Monographs on Applied Probability and Statistics. Chapman and Hall.

[Fauve et al., 2007] Fauve, B., Matrouf, D., Scheffer, N., and Bonastre, J.-F. (2007). State-of-the-art performance in text-independent speaker verification through open-source software. In *IEEE Transactions on Audio, Speech, and Language Processing*, volume 15, pages 1960–1968.

[Fayyad, 1992] Fayyad, U. M. (1992). *On the induction of decision trees for multiple concept learning*. PhD thesis, University of Michigan, Ann Arbor, MI, USA. UMI Order No. GAX92-08535.

[Fayyad and Irani, 1993] Fayyad, U. M. and Irani, K. B. (1993). Multi-interval discretization of continuousvalued attributes for classification learning. In *Thirteenth International Joint Conference on Articial Intelligence*, volume 2, pages 1022–1027. Morgan Kaufmann Publishers.

[Freund and Schapire, 1997] Freund, Y. and Schapire, R. E. (1997). A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of computer and system sciences*, 55(1) :119–139.

[Furui, 1981] Furui, S. (1981). Cepstral analysis technique for automatic speaker verification. *Acoustics, Speech and Signal Processing, IEEE Transactions on*, 29(2) :254–272.

[Gaunard et al., 1998] Gaunard, P., Mubikangiey, C. G., Couvreur, C., and Fontaine, V. (1998). Automatic classification of environmental noise events by hidden markov models. In *Acoustics, Speech and Signal Processing, 1998. Proceedings of the 1998 IEEE International Conference on*, volume 6, pages 3609–3612. IEEE.

[Gauvain and Lee, 1994] Gauvain, J.-L. and Lee, C.-H. (1994). Maximum a posteriori estimation for multivariate gaussian mixture observations of markov chains. *IEEE Transactions on Speech and Audio Processing*, 2 :291–298.

[Gazzaniga et al., 2000] Gazzaniga, M. S., Ivry, R. B., and Mangun, G. R. (2000). *Neurosciences cognitives : la biologie de l'esprit*. De Boeck.

[Gelfand, 2004] Gelfand, S. (2004). *Hearing : An Introduction to Psychological and Physiological Acoustics*. Marcel Dekker Incorporated.

[Gelfand et al., 1991] Gelfand, S. B., Ravishankar, C. S., and Delp, E. J. (1991). An iterative growing and pruning algorithm for classification tree design. *IEEE Trans. Pattern Anal. Mach. Intell.*, 13(2) :163–174.

[Gerhard, 2003] Gerhard, D. (2003). *Audio signal classification : History and current techniques*. Department of Computer Science, University of Regina, Regina, Saskatchewan, CANADA.

[Gersho and Gray, 1992] Gersho, A. and Gray, R. M. (1992). *Vector quantization and signal compression*, volume 159. Kluwer Academic Pub.

[Ghiurcau and Rusu, 2010] Ghiurcau, M. V. and Rusu, C. (2010). About classifying sounds in protected environments. In *Electrical and Electronics Engineering (ISEEE), 2010 3rd International Symposium on*, pages 84–87. IEEE.

[Gillet and Richard, 2004] Gillet, O. and Richard, G. (2004). Automatic transcription of drum loops. In *Acoustics, Speech, and Signal Processing, 2004. Proceedings. (ICASSP '04). IEEE International Conference on*, volume 4, pages iv–269–iv–272 vol.4.

[Glasberg and Moore, 1990] Glasberg, B. R. and Moore, B. C. (1990). Derivation of auditory filter shapes from notched-noise data. *Hearing Research*, 47(1-2) :103–138.

[Greenwood, 1990] Greenwood, D. D. (1990). A cochlear frequency-position function for several species–29 years later. *The Journal of the Acoustical Society of America*, 87(6) :2592–2605.

[Guo and Li, 2003] Guo, G. and Li, S. Z. (2003). Content-based audio classification and retrieval by support vector machines. *Neural Networks, IEEE Transactions on*, 14(1) :209–215.

[Guy and Rémy, 2013] Guy, R. and Rémy, P. (2013). Fonctionnement de la cochlée.

[Guyon et al., 2002] Guyon, I., Weston, J., Barnhill, S., and Vapnik, V. (2002). Gene selection for cancer classification using support vector machines. *Machine learning*, 46(1-3) :389–422.

[Hall, 1999] Hall, M. A. (1999). *Correlation-based feature selection for machine learning*. PhD thesis, The University of Waikato.

[Hall, 2000] Hall, M. A. (2000). Feature Selection for Discrete and Numeric Class Machine Learning. In *Machine Learning. Proc. Seventeenth International conference on Machine Learning*, pages 359–366.

[Hastie and Tibshirani, 1998] Hastie, T. and Tibshirani, R. (1998). Classification by pairwise coupling. *The annals of statistics*, 26(2) :451–471.

[Hatch and Stolcke, 2006] Hatch, A. O. and Stolcke, A. (2006). Generalized linear kernels for one-versus-all classification : application to speaker recognition. In *Acoustics, Speech and Signal Processing, 2006. ICASSP 2006 Proceedings. 2006 IEEE International Conference on*, volume 5, pages V–V. IEEE.

[He et al., 1999] He, J., Liu, L., and Palm, G. (1999). A discriminative training algorithm for vq-based speaker identification. *Speech and Audio Processing, IEEE Transactions on*, 7(3) :353–356.

[Hermansky, 1990] Hermansky, H. (1990). Perceptual linear predictive (plp) analysis of speech. *The Journal of the Acoustical Society of America*, 87 :1738.

[Hermansky and Morgan, 1994] Hermansky, H. and Morgan, N. (1994). Rasta processing of speech. *Speech and Audio Processing, IEEE Transactions on*, 2(4) :578–589.

[Hermansky and Sharma, 1998] Hermansky, H. and Sharma, S. (1998). Traps – classifiers of temporal patterns. In *IN PROCEEDINGS OF 5TH INTERNATIONAL CONFERENCE ON SPOKEN LANGUAGE PROCESSING, ICSLP 98*, pages 1003–1006.

[Hermansky and Sharma, 1999] Hermansky, H. and Sharma, S. (1999). Temporal patterns (traps) in asr of noisy speech. In *in Proc. ICASSP*, pages 289–292.

[Hernandez et al., 2007] Hernandez, E. M., Adiloglu, K., Annies, R., Purwins, H., and Obermayer, K. (2007). Perceptual representation for classification of everyday sounds. In *Proc. of the Conference on Interaction with Sound*, volume 2, pages 90–95. Citeseer.

[Ho and Scott, 1997] Ho, K. M. and Scott, P. D. (1997). Zeta : A global method for discretization of continuous variables. In Heckerman, D., Mannila, H., and Pregibon, D., editors, *KDD*, pages 191–194. AAAI Press.

[Ho and Scott, 1998] Ho, K. M. and Scott, P. D. (1998). An efficient global discretization method. In Wu, X., Ramamohanarao, K., and Korb, K. B., editors, *PAKDD*, volume 1394 of *Lecture Notes in Computer Science*, pages 383–384. Springer.

[Holte, 1993] Holte, R. C. (1993). Very simple classification rules perform well on most commonly used datasets. In *Machine Learning*, pages 63–91.

[Hsu et al., 2010] Hsu, C.-W., Chang, C.-C., and Lin, C.-J. (2010). A practical guide to support vector classification.

[Hsu and Lin, 2002] Hsu, C.-W. and Lin, C.-J. (2002). A comparison of methods for multiclass support vector machines. *Neural Networks, IEEE Transactions on*, 13(2) :415–425.

[Huang et al., 2001] Huang, X., Acero, A., and Hon, H. (2001). *Spoken Language Processing : A Guide to Theory, Algorithm, and System Development*. Prentice Hall PTR.

[Istrate et al., 2006a] Istrate, D., Castelli, E., Vacher, M., Besacier, L., and Serignat, J.-F. (2006a). Information extraction from sound for medical telemonitoring. *Information Technology in Biomedicine, IEEE Transactions on*, 10(2) :264–274.

[Istrate et al., 2006b] Istrate, D., Vacher, M., and Serignat, J. F. (2006b). Generic implementation of a distress sound extraction system for elder care. In *Engineering in Medicine and Biology Society, 2006. EMBS'06. 28th Annual International Conference of the IEEE*, pages 3309–3312. IEEE.

[Ito et al., 2011] Ito, A., Aiba, A., Ito, M., and Makino, S. (2011). Evaluation of abnormal sound detection using multi-stage gmm in various environments. In *Twelfth Annual Conference of the International Speech Communication Association*.

[Jaakkola and Haussler, 1998] Jaakkola, T. and Haussler, D. (1998). Exploiting generative models in discriminative classifiers. In *In Advances in Neural Information Processing Systems 11*, pages 487–493. MIT Press.

[Jain et al., 2000] Jain, A. K., Duin, R. P. W., and Mao, J. (2000). Statistical pattern recognition : A review. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 22(1) :4–37.

[Jain et al., 2002] Jain, P., Hermansky, H., and Kingsbury, B. (2002). Distributed speech recognition using noise-robust mfcc and traps-estimated manner features. In *INTERSPEECH'02*, pages –1–1.

[Jang et al., 2008] Jang, D., Jin, M., and Yoo, C. (2008). Music genre classification using novel features and a weighted voting method. In *Multimedia and Expo, 2008 IEEE International Conference on*, pages 1377–1380.

[Janvier et al., 2012] Janvier, M., Alameda-Pineda, X., Girin, L., and Horaud, R. P. (2012). Sound-event recognition with a companion humanoid. In *IEEE International Conference on Humanoid Robotics (Humanoids)*.

[Johnston, 1988] Johnston, J. D. (1988). Transform coding of audio signals using perceptual noise criteria. *Selected Areas in Communications, IEEE Journal on*, 6(2) :314–323.

[Jurado and Robledano, 2007] Jurado, C. and Robledano, D. (2007). Auditory filters at low frequencies : Erb and filter shape. Technical report, Aalborg University.

[Ke et al., 2005] Ke, Y., Hoiem, D., and Sukthankar, R. (2005). Computer vision for music identification. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 1, pages 597–604. IEEE.

[Kenny, 2005] Kenny, P. (2005). Joint factor analysis of speaker and session variability : Theory and algorithms. *CRIM, Montreal,(Report) CRIM-06/08-13*.

[Kenny et al., 2007a] Kenny, P., Boulianne, G., Ouellet, P., and Dumouchel, P. (2007a). Joint factor analysis versus eigenchannels in speaker recognition. *Audio, Speech, and Language Processing, IEEE Transactions on*, 15(4) :1435–1447.

[Kenny et al., 2007b] Kenny, P., Boulianne, G., Ouellet, P., and Dumouchel, P. (2007b). Speaker and session variability in gmm-based speaker verification. *Audio, Speech, and Language Processing, IEEE Transactions on*, 15(4) :1448–1460.

[Kenny et al., 2008] Kenny, P., Ouellet, P., Dehak, N., Gupta, V., and Dumouchel, P. (2008). A study of interspeaker variability in speaker verification. *Audio, Speech, and Language Processing, IEEE Transactions on*, 16(5) :980–988.

[Kerber, 1992] Kerber, R. (1992). Chimerge : Discretization of numeric attributes. In Swartout, W. R., editor, *AAAI*, pages 123–128. AAAI Press / The MIT Press.

[Kinnunen, 2003] Kinnunen, T. (2003). Spectral features for automatic text-independent speaker recognition. Technical report, University of Joensuu.

[Kinnunen, 2005] Kinnunen, T. (2005). *Optimizing Spectral Feature Based Text-independent Speaker Recognition*. Dissertations / University of Joensuu, Computer Science. University of Joensuu.

[Kinnunen and Alku, 2009] Kinnunen, T. and Alku, P. (2009). On separating glottal source and vocal tract information in telephony speaker verification. In *Acoustics, Speech and Signal Processing, 2009. ICASSP 2009. IEEE International Conference on*, pages 4545–4548. IEEE.

[Kinnunen et al., 2008] Kinnunen, T., Lee, K.-A., and Li, H. (2008). Dimension reduction of the modulation spectrogram for speaker verification. In *The Speaker and Language Recognition Workshop (Odyssey 2008), Stellenbosch, South Africa*.

[Kinnunen and Li, 2010a] Kinnunen, T. and Li, H. (2010a). An overview of text-independent speaker recognition : From features to supervectors. *Speech communication*, 52(1) :12–40.

[Kinnunen and Li, 2010b] Kinnunen, T. and Li, H. (2010b). An overview of text-independent speaker recognition : From features to supervectors. *Speech Communication*, 52(1) :12 – 40.

[Kotsiantis and Kanellopoulos, 2006] Kotsiantis, S. and Kanellopoulos, D. (2006). Discretization techniques : A recent survey.

[Kumar et al., 2011] Kumar, K., Kim, C., and Stern, R. M. (2011). Delta-spectral cepstral coefficients for robust speech recognition. In *Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on*, pages 4784–4787. IEEE.

[Le-Qing, 2011] Le-Qing, Z. (2011). Insect sound recognition based on mfcc and pnn. In *Multimedia and Signal Processing (CMSP), 2011 International Conference on*, volume 2, pages 42–46. IEEE.

[Lee and Rakotonirainy, 2011] Lee, J. and Rakotonirainy, A. (2011). Acoustic hazard detection for pedestrians with obscured hearing. *Intelligent Transportation Systems, IEEE Transactions on*, 12(4) :1640–1649.

[Leggetter and Woodland, 1995] Leggetter, C. and Woodland, P. (1995). Maximum likelihood linear regression for speaker adaptation of continuous density hidden markov models. *Computer speech and language*, 9(2) :171.

[Leng et al., 2012] Leng, Y. R., Tran, H. D., Kitaoka, N., and Haizhou, L. (2012). Selective gammatone envelope feature for robust sound event recognition. *IEICE TRANSACTIONS on Information and Systems*, 95(5) :1229–1237.

[Leng et al., 2010] Leng, Y. R., Tran, H. D., Kitaoka, N., and Li, H. (2010). Selective gammatone filterbank feature for robust sound event recognition. In *Eleventh Annual Conference of the International Speech Communication Association*.

[Liao et al., 2011] Liao, W.-H., Wen, J.-Y., and Kuo, J.-H. (2011). Streaming audio classification in smart home environments. In *Pattern Recognition (ACPR), 2011 First Asian Conference on*, pages 593–597. IEEE.

[Lin and Abdulla, 2007] Lin, Y. and Abdulla, W. H. (2007). Robust audio watermarking technique based on gammatone filterbank and coded-image. In *Signal Processing and Its Applications, 2007. ISSPA 2007. 9th International Symposium on*, pages 1–4. IEEE.

[Linde et al., 1980] Linde, Y., Buzo, A., and Gray, R. (1980). An algorithm for vector quantizer design. *Communications, IEEE Transactions on*, 28(1) :84–95.

[Liu et al., 2002] Liu, H., Hussain, F., Tan, C. L., and Dash, M. (2002). Discretization : An enabling technique. *Data Min. Knowl. Discov.*, 6(4) :393–423.

[Liu and Setiono, 1995] Liu, H. and Setiono, R. (1995). Chi2 : feature selection and discretization of numeric attributes. In *Tools with Artificial Intelligence, 1995. Proceedings., Seventh International Conference on*, pages 388–391.

[Lyon, 1982] Lyon, R. (1982). A computational model of filtering, detection, and compression in the cochlea. In *Acoustics, Speech, and Signal Processing, IEEE International Conference on ICASSP '82.*, volume 7, pages 1282–1285.

[Lyon, 2010] Lyon, R. F. (2010). Machine hearing : An emerging field [exploratory dsp]. *Signal Processing Magazine, IEEE*, 27(5) :131–139.

[Lyon et al., 2010a] Lyon, R. F., Katsiamis, A. G., and Drakakis, E. M. (2010a). History and future of auditory filter models. In *ISCAS*, pages 3809–3812. IEEE.

[Lyon et al., 2010b] Lyon, R. F., Rehn, M., Bengio, S., Walters, T. C., and Chechik, G. (2010b). Sound retrieval and ranking using sparse auditory representations. *Neural computation*, 22(9) :2390–2416.

[Ma et al., 2006] Ma, L., Milner, B., and Smith, D. (2006). Acoustic environment classification. *ACM Transactions on Speech and Language Processing (TSLP)*, 3(2) :1–22.

[Madjarov and Gjorgjevikj, 2009] Madjarov, G. and Gjorgjevikj, D. (2009). Multi-class classification using support vector machines in decision tree architecture. In *EUROCON 2009, EUROCON '09. IEEE*, pages 288–295.

[Madjarov and Gjorgjevikj, 2012] Madjarov, G. and Gjorgjevikj, D. (2012). Hybrid decision tree architecture utilizing local svms for multi-label classification. In *Hybrid Artificial Intelligent Systems*, volume 7209 of *Lecture Notes in Computer Science*, pages 1–12. Springer Berlin Heidelberg.

[Madzarov and Gjorgjevikj, 2010] Madzarov, G. and Gjorgjevikj, D. (2010). Evaluation of distance measures for multi-class classification in binary svm decision tree. In *Proceedings of the 10th international conference on Artificial intelligence and soft computing : Part I*, ICAISC'10, pages 437–444, Berlin, Heidelberg. Springer-Verlag.

[Markel et al., 1977] Markel, J., Oshika, B., and Gray Jr, A. (1977). Long-term feature averaging for speaker recognition. *Acoustics, Speech and Signal Processing, IEEE Transactions on*, 25(4) :330–337.

[Matos et al., 2006] Matos, S., Birring, S. S., Pavord, I. D., and Evans, H. (2006). Detection of cough signals in continuous audio recordings using hidden markov models. *Biomedical Engineering, IEEE Transactions on*, 53(6) :1078–1083.

[McDermott, 2009] McDermott, J. H. (2009). The cocktail party problem. *Current Biology*, 19(22) :R1024–R1027.

[Michalski and Stepp, 1983] Michalski, R. S. and Stepp, R. E. (1983). Learning from observation : Conceptual clustering. In Michalski, R. S., Carbonell, J. G., and Mitchell, T. M., editors, *Machine Learning : An Artificial Intelligence Approach*, chapter 11, pages 331–364. Tioga.

[Mierswa and Morik, 2005] Mierswa, I. and Morik, K. (2005). Automatic feature extraction for classifying audio data. *Machine Learning Journal*, 58 :127–149.

[Minh and Lee, 2004] Minh, V. D. and Lee, S. (2004). Pca-based human auditory filter bank for speech recognition. In *Signal Processing and Communications, 2004. SPCOM '04. 2004 International Conference on*, pages 393–397.

[Moore, 1986] Moore, B. C. (1986). Parallels between frequency selectivity measured psychophysically and in cochlear mechanics. *Scand Audiol Suppl*, 25 :139–52.

[Moore and Glasberg, 1983] Moore, B. C. and Glasberg, B. R. (1983). Suggested formulae for calculating auditory-filter bandwidths and excitation patterns. *The Journal of the Acoustical Society of America*, 74 :750.

[Moore et al., 1997] Moore, B. C., Glasberg, B. R., and Baer, T. (1997). A model for the prediction of thresholds, loudness, and partial loudness. *Journal of the Audio Engineering Society*, 45(4) :224–240.

[Moore et al., 1995] Moore, B. C., Glasberg, B. R., van der Heijden, M., Houtsma, A. J., and Kohlrausch, A. (1995). Comparison of auditory filter shapes obtained with notched-noise and noise-tone maskers. *The Journal of the Acoustical Society of America*, 97(2) :1175–82.

[Moreno et al., 2003] Moreno, P. J., Ho, P., and Vasconcelos, N. (2003). A kullback-leibler divergence based kernel for svm classification in multimedia applications. *Advances in neural information processing systems*, 16 :1385–1393.

[Munkong and Juang, 2008] Munkong, R. and Juang, B.-H. (2008). Auditory perception and cognition. *Signal Processing Magazine, IEEE*, 25(3) :98–117.

[Noma, 2002] Noma, H. S. K.-i. (2002). Dynamic time-alignment kernel in support vector machine. In *Advances in Neural Information Processing Systems 14 : Proceedings of the 2002 Conference*, volume 2, page 921. MIT Press.

[Ntalampiras et al., 2009] Ntalampiras, S., Potamitis, I., and Fakotakis, N. (2009). On acoustic surveillance of hazardous situations. In *Acoustics, Speech and Signal Processing, 2009. ICASSP 2009. IEEE International Conference on*, pages 165–168. IEEE.

[Patterson et al., 1987] Patterson, R., Nimmo-Smith, I., Holdsworth, J., and Rice, P. (1987). An efficient auditory filterbank based on the gammatone function. In *a meeting of the IOC Speech Group on Auditory Modelling at RSRE*, volume 2.

[Patterson, 1976] Patterson, R. D. (1976). Auditory filter shapes derived with noise stimuli. *The Journal of the Acoustical Society of America*, 59(3) :640–654.

[Patterson et al., 1995] Patterson, R. D., Allerhand, M. H., and Giguère, C. (1995). Time-domain modeling of peripheral auditory processing : A modular architecture and a software platform. *The Journal of the Acoustical Society of America*, 98(4) :1890–1894.

[Patterson and Henning, 1977] Patterson, R. D. and Henning, G. B. (1977). Stimulus variability and auditory filter shape. *The Journal of the Acoustical Society of America*, 62(3) :649–664.

[Peeters, 2004] Peeters, G. (2004). A large set of audio features for sound description (similarity and classification) in the CUIDADO project. Tech. rep., IRCAM.

[Peters and Moore, 1992] Peters, R. and Moore, B. C. (1992). Auditory filter shapes at low center frequencies in young and elderly hearing-impaired subjects. *The Journal of the Acoustical Society of America*, 91(1) :256–66.

[Pickles, 2008] Pickles, J. (2008). *Introduction to the Physiology of Hearing 3e*. Emerald Group Publishing Limited.

[Picone, 1993] Picone, J. (1993). Signal modeling techniques in speech recognition. *Proceedings of the IEEE*, 81(9) :1215–1247.

[Platt, 1999] Platt, J. C. (1999). Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. In *Advances in Large Margin Classifiers*, pages 61–74. MIT Press.

[Platt et al., 1999] Platt, J. C., Cristianini, N., and Shawe-Taylor, J. (1999). Large margin dags for multiclass classification. In *NIPS*, volume 12, pages 547–553.

[Quinlan, 1986] Quinlan, J. R. (1986). Induction of decision trees. *Mach. Learn.*, 1(1) :81–106.

[Quinlan, 1987] Quinlan, J. R. (1987). Simplifying decision trees. *Int. J. Man-Mach. Stud.*, 27(3) :221–234.

[Quinlan, 1993] Quinlan, J. R. (1993). *C4.5 : programs for machine learning*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.

[Rabiner, 1989] Rabiner, L. (1989). A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2) :257–286.

[Rademacher and Mertins, 2006] Rademacher, J. and Mertins, A. (2006). Auditory filterbank based frequency-warping invariant features for automatic speech recognition. In *Proc. ITG-Fachtagung Sprachkommunikation*, Kiel.

[Radocy and Boyle, 2003] Radocy, R. and Boyle, J. (2003). *Psychological foundations of musical behavior*. Charles C. Thomas.

[Reynolds, 2009] Reynolds, D. A. (2009). Gaussian mixture models. In *Encyclopedia of Biometrics*, pages 659–663. Springer.

[Reynolds et al., 2000] Reynolds, D. A., Quatieri, T. F., and Dunn, R. B. (2000). Speaker verification using adapted gaussian mixture models. *Digital signal processing*, 10(1) :19–41.

[Reynolds and Rose, 1995] Reynolds, D. A. and Rose, R. C. (1995). Robust text-independent speaker identification using gaussian mixture speaker models. *IEEE Transactions on Speech and Audio Processing*, pages 72–80.

[Romani et al., 1975] Romani, G. L., Williamson, S. J., and Kaufman, L. (1975). Tonotopic organization of the human auditory cortex. *Psychiatry*, 132 :650.

[Rougui et al., 2009] Rougui, J., Istrate, D., and Souidene, W. (2009). Audio sound event identification for distress situations and context awareness. In *Engineering in Medicine and Biology Society, 2009. EMBC 2009. Annual International Conference of the IEEE*, pages 3501–3504, Minneapolis, USA.

[Sahidullah and Saha, 2012] Sahidullah, M. and Saha, G. (2012). Design, analysis and experimental evaluation of block based transformation in {MFCC} computation for speaker recognition. *Speech Communication*, 54(4) :543 – 565.

[Schafer, 1969a] Schafer, R. (1969a). *The New Soundscape*. Universal Edition, Vienna.

[Schafer, 1969b] Schafer, R. (1969b). *The new soundscape : a handbook for the modern music teacher*. BMI Canada.

[Schapire and Singer, 1999] Schapire, R. E. and Singer, Y. (1999). Improved boosting algorithms using confidence-rated predictions. *Machine learning*, 37(3) :297–336.

[Scheirer and Slaney, 1997] Scheirer, E. and Slaney, M. (1997). Construction and evaluation of a robust multifeature speech/music discriminator. In *Acoustics, Speech, and Signal Processing, 1997. ICASSP-97., 1997 IEEE International Conference on*, volume 2, pages 1331–1334 vol.2.

[Schluter et al., 2007] Schluter, R., Bezrukov, L., Wagner, H., and Ney, H. (2007). Gammatone features and feature combination for large vocabulary speech recognition. In *Acoustics, Speech and Signal Processing, 2007. ICASSP 2007. IEEE International Conference on*, volume 4, pages IV–649–IV–652.

[Schmidt and Gish, 1996] Schmidt, M. and Gish, H. (1996). Speaker identification via support vector classifiers. In *Acoustics, Speech, and Signal Processing, 1996. ICASSP-96. Conference Proceedings., 1996 IEEE International Conference on*, volume 1, pages 105–108 vol. 1.

[Schroeder, 1977] Schroeder, M. R. (1977). Recognition of complex acoustic signals. *Life Science Research Report*, 55 :323–328.

[Schuller et al., 2011] Schuller, B., Batliner, A., Steidl, S., and Seppi, D. (2011). Recognising realistic emotions and affect in speech : State of the art and lessons learnt from the first challenge. *Speech Communication*, 53(9) :1062–1087.

[Schölkopf et al., 1995] Schölkopf, B., Burges, C., and Vapnik, V. (1995). Extracting support data for a given task. In *KDD*, volume 95, pages 252–257.

[Schölkopf and Smola, 2002] Schölkopf, B. and Smola, A. J. (2002). *Learning with Kernels*. MIT Press.

[Sehili et al., 2010] Sehili, M. A., Istrate, D., and Boudy, J. (2010). Primary investigations of sound recognition for a domotic application using support vector. *Annals of the University of Craiova, Series : Automation, Computers, Electronics and Mechatronics*, 7(34)(2) :61–65.

[Sehili et al., 2012a] Sehili, M. A., Istrate, D., Dorizzi, B., and Boudy, J. (2012a). Daily sound recognition using a combination of gmm and svm for home automation. In *Signal Processing Conference (EUSIPCO), 2012 Proceedings of the 20th European*, pages 1673–1677.

[Sehili et al., 2012b] Sehili, M. A., Lecouteux, B., Vacher, M., Portet, F., Istrate, D., Dorizzi, B., and Boudy, J. (2012b). Sound environment analysis in smart home. In *Ambient Intelligence*, volume 7683, pages 208–223. Springer Berlin Heidelberg.

[Senoussaoui et al., 2010] Senoussaoui, M., Kenny, P., Dehak, N., and Dumouchel, P. (2010). An i-vector extractor suitable for speaker recognition with both microphone and telephone speech. In *Proc. Odyssey Speaker and Language Recogntion Workshop*, pages 28–33.

[Shannon, 1948] Shannon, C. E. (1948). A mathematical theory of communication. *The Bell System Technical Journal*, 27 :379–423, 623–656.

[Shannon and Weaver, 1949] Shannon, C. E. and Weaver, W. (1949). *The Mathematical Theory of Communication*. Number vol. 1 in Illini books. University of Illinois Press.

[Shao et al., 2009] Shao, Y., Jin, Z., Wang, D., and Srinivasan, S. (2009). An auditory-based feature for robust speech recognition. In *Proceedings of the 2009 IEEE International Conference on Acoustics, Speech and Signal Processing*, ICASSP '09, pages 4625–4628, Washington, DC, USA. IEEE Computer Society.

[Shriberg et al., 2005] Shriberg, E., Ferrer, L., Kajarekar, S., Venkataraman, A., and Stolcke, A. (2005). Modeling prosodic feature sequences for speaker recognition. *Speech Communication*, 46(3) :455–472.

[Sivasankaran and Prabhu, 2013] Sivasankaran, S. and Prabhu, K. (2013). Robust features for environmental sound classification. In *Electronics, Computing and Communication Technologies (CONECCT), 2013 IEEE International Conference on*, pages 1–6. IEEE.

[Slaney, 1993] Slaney, M. (1993). An efficient implementation of the patterson-holdsworth auditory filter bank. *Apple Computer, Perception Group, Tech. Rep*.

[Smith and Abel, 1999] Smith, J. O. and Abel, J. S. (1999). Bark and erb bilinear transforms. *IEEE Transactions on Speech and Audio Processing*, 7 :697–708.

[Smith and Gales, 2002] Smith, N. and Gales, M. (2002). Speech recognition using svms. In *Advances in Neural Information Processing Systems 14*, pages 1197–1204. MIT Press.

[Smith III and Abel, 1999] Smith III, J. O. and Abel, J. S. (1999). Bark and erb bilinear transforms. *Speech and Audio Processing, IEEE Transactions on*, 7(6) :697–708.

[Solomonoff et al., 2005] Solomonoff, A., Campbell, W. M., and Boardman, I. (2005). Advances in channel compensation for svm speaker recognition. In *Proc. ICASSP*, volume 1, pages 629–632.

[Somervuo et al., 2006] Somervuo, P., Harma, A., and Fagerlund, S. (2006). Parametric representations of bird sounds for automatic species recognition. *Audio, Speech, and Language Processing, IEEE Transactions on*, 14(6) :2252–2263.

[Sommers and Humes, 1993] Sommers, M. S. and Humes, L. E. (1993). Auditory filter shapes in normal-hearing, noise-masked normal, and elderly listeners. *The Journal of the Acoustical Society of America*, 93 :2903.

[Soong and Rosenberg, 1988] Soong, F. K. and Rosenberg, A. E. (1988). On the use of instantaneous and transitional spectral information in speaker recognition. *Acoustics, Speech and Signal Processing, IEEE Transactions on*, 36(6) :871–879.

[Soong et al., 1985] Soong, F. K., Rosenberg, A. E., Rabiner, L., and Juang, B. (1985). A vector quantization approach to speaker recognition. In *Acoustics, Speech, and Signal Processing, IEEE International Conference on ICASSP'85.*, volume 10, pages 387–390. IEEE.

[Spencer, 2000] Spencer, C. (2000). Local and provincial initiatives in rental housing safety.

[Srinivasan and Wang, 2008] Srinivasan, S. and Wang, D. (2008). A model for multitalker speech perception. *The Journal of the Acoustical Society of America*, 124(5) :3213–24.

[Stolcke et al., 2007] Stolcke, A., Kajarekar, S. S., Ferrer, L., and Shrinberg, E. (2007). Speaker recognition with session variability normalization based on mllr adaptation transforms. *Audio, Speech, and Language Processing, IEEE Transactions on*, 15(7) :1987–1998.

[Talavage et al., 2004] Talavage, T. M., Sereno, M. I., Melcher, J. R., Ledden, P. J., Rosen, B. R., and Dale, A. M. (2004). Tonotopic organization in human auditory cortex revealed by progressions of frequency sensitivity. *Journal of neurophysiology*, 91(3) :1282–1296.

[Temko et al., 2005] Temko, A., Monte, E., and Nadeu, C. (2005). Comparison of sequence discriminant support vector machines for acoustic event classification. In *In Proceedings of the International Conference on Acoustics, Speech, and Signal Processing*.

[Temko and Nadeu, 2005] Temko, A. and Nadeu, C. (2005). Classification of meeting-room acoustic events with support vector. In *Machines and Confusion-based Clustering", Proc. ICASSP'05*, pages 505–508.

[Titterington et al., 1985] Titterington, D., Smith, A., and Makov, U. (1985). *Statistical analysis of finite mixture distributions*. Wiley series in probability and mathematical statistics : Applied probability and statistics. Wiley.

[Treichler, 2009] Treichler, J. (2009). Signal processing : A view of the future, part 2 [exploratory dsp]. *Signal Processing Magazine, IEEE*, 26(3) :83–86.

[Vacher et al., 2008] Vacher, M., Fleury, A., Serignat, J.-F., Noury, N., and Glasson, H. (2008). Preliminary evaluation of speech/sound recognition for telemedicine application in a real environment. In *Proceedings of the Interspeech 2008 conference*, pages 496–499.

[Vacher et al., 2004] Vacher, M., Istrate, D., Besacier, L., Serignat, J., and Castelli, E. (2004). Sound detection and classification for medical telesurvey. In *Proceedings of the International Conference on Biomedical Engineering*, pages 395–399.

[Vacher et al., 2011] Vacher, M., Istrate, D., Portet, F., Joubert, T., Chevalier, T., Smidtas, S., Meillon, B., Lecouteux, B., Sehili, M. A., Chahuara, P., and Meniard, S. (2011). The SWEET-HOME Project : Audio Technology in Smart Homes to improve Well-being and Reliance. In *33rd Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC 2011)*, pages 5291–5294, Boston, USA.

[Valenzise et al., 2007] Valenzise, G., Gerosa, L., Tagliasacchi, M., Antonacci, E., and Sarti, A. (2007). Scream and gunshot detection and localization for audio-surveillance systems. In *Advanced Video and Signal Based Surveillance, 2007. AVSS 2007. IEEE Conference on*, pages 21–26.

[Valero and Alías, 2012] Valero, X. and Alías, F. (2012). Gammatone wavelet features for sound classification in surveillance applications. In *Signal Processing Conference (EUSIPCO), 2012 Proceedings of the 20th European*, pages 1658–1662. IEEE.

[VanDerveer, 1979] VanDerveer, N. (1979). *Ecological Acoustics : Human Perception of Environmental Sounds*. PhD thesis, Cornell University.

[Vapnik, 1998] Vapnik, V. (1998). *Statistical learning theory*. Wiley.

[Vapnik, 2010] Vapnik, V. (2010). *The Nature of Statistical Learning Theory*. Information Science and Statistics. Springer.

[Viola and Jones, 2001] Viola, P. and Jones, M. J. (2001). Rapid object detection using a boosted cascade of simple features. In *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, volume 1, pages I–511. IEEE.

[Viola and Jones, 2004] Viola, P. and Jones, M. J. (2004). Robust real-time face detection. *International journal of computer vision*, 57(2) :137–154.

[Vogt and Sridharan, 2008] Vogt, R. and Sridharan, S. (2008). Explicit modelling of session variability for speaker verification. *Computer Speech & Language*, 22(1) :17–38.

[Vogt et al., 2008] Vogt, R. J., Kajarekar, S., and Sridharan, S. (2008). Discriminant nap for svm speaker recognition. In *Odyssey 2008 : The Speaker and Language Recognition Workshop*, Stellenbosch, South Africa. IEEE.

[Wan and Campbell, 2000] Wan, V. and Campbell, W. M. (2000). Support vector machines for speaker verification and identification. In *Neural Networks for Signal Processing X, 2000. Proceedings of the 2000 IEEE Signal Processing Society Workshop*, volume 2, pages 775–784. IEEE.

[Wan and Carmichael, 2005] Wan, V. and Carmichael, J. (2005). Polynomial dynamic time warping kernel support vector machines for dysarthric speech recognition with sparse training data. In *Proc. Annu. Conf. Int. Speech Commun. Assoc.(Interspeech 2005)*.

[Wang et al., 2008] Wang, J.-C., Lee, H.-P., Wang, J.-F., and Lin, C.-B. (2008). Robust environmental sound recognition for home automation. *Automation Science and Engineering, IEEE Transactions on*, 5(1) :25–31.

[Wang and Liu, 1998] Wang, K. and Liu, B. (1998). Concurrent discretization of multiple attributes. In *In Proceedings of the Pacific Rim International Conference on Artificial Intelligence*, pages 250–259. Springer-Verlag.

[Waterman, 1986] Waterman, D. (1986). *A Guide Expert Systems*. The Teknowledge Series in Knowledge Engineering. Addison-Wesley.

[Weiser, 1991] Weiser, M. (1991). The computer for the 21st century. *Scientific american*, 265(3) :94–104.

[West and Cox, 2004] West, K. and Cox, S. J. (2004). Features and classifiers for the automatic classification of musical audio signals. In *Proceedings of the 5th International Conference on Music Information Retrieval (ISMIR 2004)*.

[Weston and Watkins, 1999] Weston, J. and Watkins, C. (1999). Support vector machines for multi-class pattern recognition. In *ESANN*, volume 99, pages 61–72.

[Xiang et al., 2010] Xiang, J.-J., McKinney, M. F., Fitz, K., and Zhang, T. (2010). Evaluation of sound classification algorithms for hearing aid applications. In *Acoustics Speech and Signal Processing (ICASSP), 2010 IEEE International Conference on*, pages 185–188. IEEE.

[Yamakawa et al., 2010] Yamakawa, N., Kitahara, T., Takahashi, T., Komatani, K., Ogata, T., and Okuno, H. G. (2010). Effects of modelling within-and between-frame temporal variations in power spectra on non-verbal sound recognition. In *Proc. 2010 International Conference on Spoken Language Processing, Makuhari*, pages 2342–2345. Citeseer.

[Yuan and Peng, 2012] Yuan, X. and Peng, S. (2012). A research on secure smart home based on the internet of things. In *Information Science and Technology (ICIST), 2012 International Conference on*, pages 737–740. IEEE.

[Zemlin, 1998] Zemlin, W. (1998). *Speech and hearing science : anatomy and physiology*. Allyn and Bacon.

[Zhang and Abdulla, 2005] Zhang, Y. and Abdulla, W. H. (2005). Gammatone auditory filterbank and independent component analysis for speaker identification systems. Technical report, Tech. Rep., The University of Auckland.

[Zhao et al., 2012] Zhao, X., Shao, Y., and Wang, D. (2012). Casa-based robust speaker identification. *Audio, Speech, and Language Processing, IEEE Transactions on*, 20(5) :1608–1616.

[Zheng et al., 2001] Zheng, F., Zhang, G., and Song, Z. (2001). Comparison of different implementations of mfcc. *Journal of Computer Science and Technology*, 16(6) :582–589.

[Zhu et al., 2006] Zhu, X., Barras, C., Lamel, L., and Gauvain, J.-L. (2006). Speaker diarization : From broadcast news to lectures. In *Machine Learning for Multimodal Interaction*, pages 396–406. Springer.

[Zhu et al., 2005] Zhu, X., Barras, C., Meignier, S., and Gauvain, J.-L. (2005). Combining speaker identification and bic for speaker diarization. In *INTERSPEECH*, volume 5, pages 2441–2444.

[Zhuang et al., 2010] Zhuang, X., Zhou, X., Hasegawa-Johnson, M. A., and Huang, T. S. (2010). Real-world acoustic event detection. *Pattern Recognition Letters*, 31(12) :1543–1551.

[Zienowicz et al., 2008] Zienowicz, K., Shihab, A., and Hunter, G. J. A. (2008). Detecting, classifying and predicting salient events using acoustic signals and markov models. In *Intelligent Environments, 2008 IET 4th International Conference on*, pages 1–8.

[Zwicker, 1961] Zwicker, E. (1961). Subdivision of the audible frequency range into critical bands (frequenzgruppen). *The Journal of the Acoustical Society of America*, 33 :248.