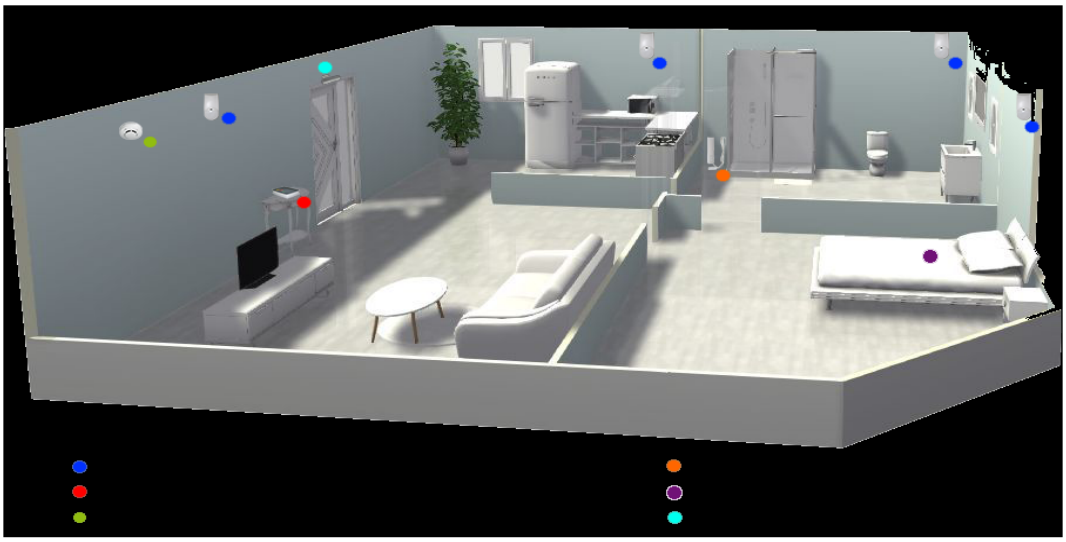
Analyse de l'environnement sonore pour le maintien à domicile et la reconnaissance d'activités de la vie courante, des personnes âgées

Maxime Robin

Par **Maxime ROBIN**

Analyse de l'environnement sonore pour le maintien à domicile et la reconnaissance d'activités de la vie courante, des personnes âgées

Thèse présentée
pour l'obtention du grade
de Docteur de l'UTC



Soutenue le 17 avril 2018
Spécialité : Informatique - Bio-ingénierie : Unité de Recherche Biomécanique et Bio-ingénierie (UMR-7338)
D2421

Thèse présentée pour obtenir le grade de docteur
Université de technologie de Compiègne

Discipline : Informatique – Bio-ingénierie

Analyse de l'environnement sonore pour le maintien à domicile et la reconnaissance d'activités de la vie courante, des personnes âgées

Par : **Maxime Robin**

Membres du jury:

Rapporteur : Jacques Demongeot, Professeur des Universités Émérite, AGEIS, Université Grenoble Alpes

Rapporteur : Edwige Pissaloux, Professeur des Universités, LITIS — Université de Rouen

Examinatrice : Catherine Marque, Professeur des Universités, BMBI UMR 7338 – Université de Technologie de Compiègne

Examinatrice : Anna Ozguler, Docteur épidémiologie, SAMU 92 – Hôpital Raymond Poincaré

Directeur de thèse : Dan Istrate, Enseignant-Chercheur HDR, BMBI UMR 7338 – Université de Technologie de Compiègne

Directeur de thèse : Jérôme Boudy, Professeur à Télécom SudParis, SAMOVAR UMR 5157 – Télécom Sud-Paris

Membre invité : Vincent Kraus, Encadrant entreprise, KRG Corporate

Date de soutenance : 17 avril 2018

Table des matières

Résumé

L'âge moyen de la population française et européenne augmente, cette constatation apporte de nouveaux enjeux techniques et sociétaux, les personnes âgées étant les personnes les plus fragiles et les plus vulnérables, notamment du point de vue des accidents domestiques et en particulier des chutes. C'est pourquoi de nombreux projets d'aide au personnes âgées : techniques, universitaires et commerciaux ont vu le jour ces dernières années.

Ce travail de thèse a été effectuée sous convention CIFRE, conjointement entre l'entreprise KRG Corporate et le laboratoire BMBI (Biomécanique et Bio-ingénierie) de l'UTC (Université de technologie de Compiègne). Elle a pour objet de proposer un capteur de reconnaissance de sons et des activités de la vie courante, dans le but d'étoffer et d'améliorer le système de télé-assistance déjà commercialisé par la société.

Plusieurs méthodes de reconnaissance de parole ou de reconnaissance du locuteur on déjà été éprouvées dans le domaine de la reconnaissance de sons, entre autres les techniques : GMM (Modèle de mélange gaussien – *Gaussian Mixture Model*), SVM-GSL (Machine à vecteurs de support, GMM-super-vecteur à noyau linéaire – *Support vector machine GMM Supervector Linear kernel*) et HMM (Modèle de markov caché – *Hidden Markov Model*). De la même manière, nous nous sommes proposés d'utiliser les i-vecteurs pour la reconnaissance de sons. Les i-vecteurs sont utilisés notamment en reconnaissance de locuteur, et ont révolutionné ce domaine récemment. Puis nous avons élargi notre spectre, et utilisé l'apprentissage profond (*Deep Learning*) qui donne actuellement de très bon résultats en classification tous domaines confondus. Nous les avons tout d'abord utilisés en renfort des i-vecteurs, puis nous les avons utilisés comme système de classification exclusif.

Les méthodes précédemment évoquées ont également été testées en conditions bruités puis réelles. Ces différentes expérimentations nous ont permis d'obtenir des taux de reconnaissance très satisfaisants, les réseaux de neurones en renfort des i-vecteurs et les réseaux de neurones seuls étant les systèmes ayant la meilleure précision, avec une amélioration très significative par rapport aux différents systèmes issus de la reconnaissance de parole et de locuteur.

Mots clefs

Reconnaissance de sons, i-vecteurs, réseau de neurones profonds, paramètres acoustiques, RER (*Remarkable Energy Rate*), classification en milieux bruités.

Abstract

The average age of the French and European population is increasing ; this observation brings new technical and societal challenges. Older people are the most fragile and vulnerable, especially in terms of domestic accidents and specifically falls. This is why many elderly people care projects : technical, academic and commercial have seen the light of day in recent years.

This thesis work was carried out under CIFRE agreement, jointly between the company KRG Corporate and the BMBI laboratory (Biomechanics and Bioengineering) of the UTC (Université of Technologie of Compiègne). Its purpose is to offer a sensor for sound recognition and everyday activities, with the aim of expanding and improving the tele-assistance system already marketed by the company.

Several speech recognition or speaker recognition methods have already been proven in the field of sound recognition, including GMM (Modèle de mélange gaussien – *Gaussian Mixture Model*), SVM-GSL (Machine à vecteurs de support, GMM-super-vecteur à noyau linéaire – *Support vector machine GMM Supervector Linear kernel*) and HMM (Modèle de markov caché – *Hidden Markov Model*). In the same way, we proposed to use i-vectors for sound recognition. I-Vectors are used in particular in speaker recognition, and have revolutionized this field recently. Then we broadened our spectrum, and used Deep Learning, which currently gives very good results in classification across all domains. We first used them to reinforce the i-vectors, then we used them as our exclusive classification system.

The methods mentioned above were also tested under noisy and then real conditions. These different experiments gaves us very satisfactory recognition rates, with neural networks as reinforcement for i-vectors and neural networks alone being the most accurate systems, with a very significant improvement compared to the various speech and speaker recognition systems.

Keywords

Sounds recognition, i-vectors, deep learning, RER (*Remarkable Energy Rate*), acoustic parameters, classification in noisy environments.

Table des figures

Liste des tableaux

Glossaire

ANA *Absolute Normalization and Add*. 55, 66, 70, 87, 101

AVC Accident Vasculaire Cérébral. 21

dB Décibel. 50, 51, 93, 94, 100, 101

EFR *Eigenfactor Radial*. 52, 53

ELU *Exponential Linear Unit*. 82

EM Algorithme d'espérance-maximisation – *Expectation-Maximization*. 13, 41, 46, 47

FFT Transformée de Fourier rapide – *Fast Fourier Transform*. 34, 35

GAFA Google, Apple, Facebook, Amazon. 65

GEM Généralisation de l'algorithme EM (Algorithme d'espérance-maximisation – *Expectation-Maximization*) – *Generalized EM*. 46–48

GMM Modèle de mélange gaussien – *Gaussian Mixture Model*. 5, 7, 9, 14, 28, 29, 33, 41–46, 57

GPGPU *General-purpose processing on graphics processing units*. 65

GPU Processeur graphique – *Graphics Processing Unit*. 88

GRU *Gated recurrent unit*. 76, 77

HMM Modèle de markov caché – *Hidden Markov Model*. 5, 7, 33, 34, 45

HTTPS *HyperText Transfer Protocol Secure*. 92

IA Intelligence Artificielle. 65, 66

IoT Internet des objets – *Internet Of Things*. 15

JFA *Joint Factor Analysis*. 45, 46

LDA Analyse discriminante linéaire – *Linear Discriminant Analysis*. 52

LSTM *Long short-term memory*. 76, 77

MFCC *Mel-Frequency Cepstral Coefficients*. 9, 23, 35–37, 42, 45, 49, 57, 58, 71, 84, 114

MNIST *Mixed National Institute of Standards and Technology*. 27

NFD *Normalized First or Delta*. 55, 100

PLDA *Probabilistic Linear Discriminant Analysis*. 52–54

ReLU *Rectified Linear Units*. 83

RER *Remarkable Energy Rate*. 6, 8, 9, 11, 35, 39, 40, 61, 71, 114

SC *Spectral Centroid*. 35, 38, 39, 57, 71, 114

SNR Rapport signal sur bruit – *Signal to Noise Ratio*. 51, 62, 63, 67, 69, 70, 73, 85, 87, 89, 91–95, 100, 101

SphNorm *Spherical Nuisance Normalisation*. 52–54

SRF *Spectral Rolloff Point*. 9, 35, 37–39, 57, 71, 114

SVM Machine à vecteurs de support– *Support vector machine*. 42, 43

SVM-GSL Machine à vecteurs de support, GMM-super-vecteur à noyau linéaire – *Support vector machine GMM Supervector Linear kernel*. 5, 7, 33, 42–45, 57, 64

TV Matrice de variabilité totale – *Total Variability matrix*. 45–47

UBM Modèle du monde – *Universal Background Model*. 46, 47, 49

WCCN *Within-Class Covariance Normalisation*. 52, 53

ZCR *Zero Crossing Rate*. 9, 34, 38, 39, 57, 71, 114

Chapitre 1

Contexte de la thèse

1.1 Introduction

Ces dernières années nous avons pu assister à l'avènement de l'IoT (Internet des objets – *Internet Of Things*), et donc à une explosion de communications commerciales et de consommation des objets connectés – de 6 millions d'objets connectés en 2016, nous devrions atteindre les 20 millions d'ici 2020 [2] –, un objet connecté étant un objet courant possédant une connexion wifi ou bluetooth et passant par une passerelle – téléphone mobile, routeur internet – et permettant d'agir sur l'objet à distance, ou de stocker les données que ce dernier collecte. Cette explosion est aujourd'hui encore incontrôlée et ne propose que des objets de confort apparent. En effet il n'y a pas de véritable solution de gestion des objets connectés uniformisée, et donc utiliser divers fournisseurs (ou marque) d'objets connectés revient à un fastidieux parcours quotidien pour l'utilisateur final. Ces objets (ampoules, bracelets et montres, prises électriques, pèse-personne, etc.) n'apportent pas non plus de réelle plus-value à l'utilisation, par rapport à leur équivalent non connecté, en effet peu d'offres d'objets connectés proposent des systèmes intelligents permettant de conseiller ou d'aider son utilisateur au quotidien, la fouille et l'utilisation des données étant bien souvent effectués par l'entreprise pour la collecte et revente de données à des tiers. Actuellement la maison intelligente aidant ses occupants semble encore être une utopie futuriste.

En dépit de ce constat pessimiste, il est important de noter que la multiplication de capteurs connectés permet de simplifier l'agrégation et l'utilisation de capteurs pour un projet donné. Cela diminue leur prix et permet aux utilisateurs avancés d'effectuer une synthèse et une mise en relation plus aisée des différents capteurs en présence et utilisables. Désormais il est possible de prendre les capteurs les plus intéressants et d'utiliser leur données sans devoir passer par une phase de conception de capteurs et par conséquent de rendre les différents projets réalisables sur le plan pécuniaire.

L'utilisation de ces nouveaux capteurs a permis l'émergence de la "santé connectée" et de la télémédecine, un domaine où le capteur n'est plus uniquement un outil de confort mais une véritable aide de vie. La télémédecine est définie par le fait d'effectuer un acte médical (télé-consultation, télé-expertise, téléassistance médicale, télésurveillance) à distance. La télé-santé et la santé connectée agissent plus comme une aide de vie axée sur la santé du patient. Ces domaines permettent de contourner succinctement le problème des déserts médicaux, qui pour la plupart touchent des personnes isolées qui en ont le plus besoin. Actuellement la santé connectée agit bien souvent comme les systèmes de santé réels et seulement en réaction à un problème donné. Grâce à la multiplication de capteurs et de données nous pouvons désormais entrevoir la possibilité de nous tourner vers une médecine préventive et qui peut être automatisée grâce à une analyse plus fine ou du moins avec un plus grand nombre de paramètres (contextuels, environnementaux, vitaux, quotidiens, etc.) pris en compte.

1.1.1 Maintien de personnes âgées à domicile, enjeux et dangers

En Europe la population est vieillissante, la population âgée étant définie par le nombre de personnes de plus de 65 ans (Figure 1.1) [44]. Cette population âgée de plus en plus nombreuse, implique de nouveaux enjeux sociétaux mais également sociaux. En effet, les personnes âgées représentent la population la plus touchée par les accidents domestiques, mais aussi la plus isolée et la plus vulnérable. Si l'on met ce fait en corrélation avec le constat que les seniors mettent également plus de temps en moyenne que les autres populations à se remettre d'un accident, nous pouvons mettre en évidence la nécessité de sécuriser les personnes âgées à domicile ou tout du moins de limiter l'impact a posteriori d'un accident à domicile. En effet l'utilisation de capteurs et d'intelligences artificielles, pourrait pallier en partie l'isolement des personnes âgées. L'isolement social ne sera pas pour autant résolu, mais un accès plus simple à la technologie et avec un meilleur suivi médical et paramédical, de cette population, pourrait du moins prévenir ou identifier un isolement grandissant, reflet d'une baisse d'activité de la personne.

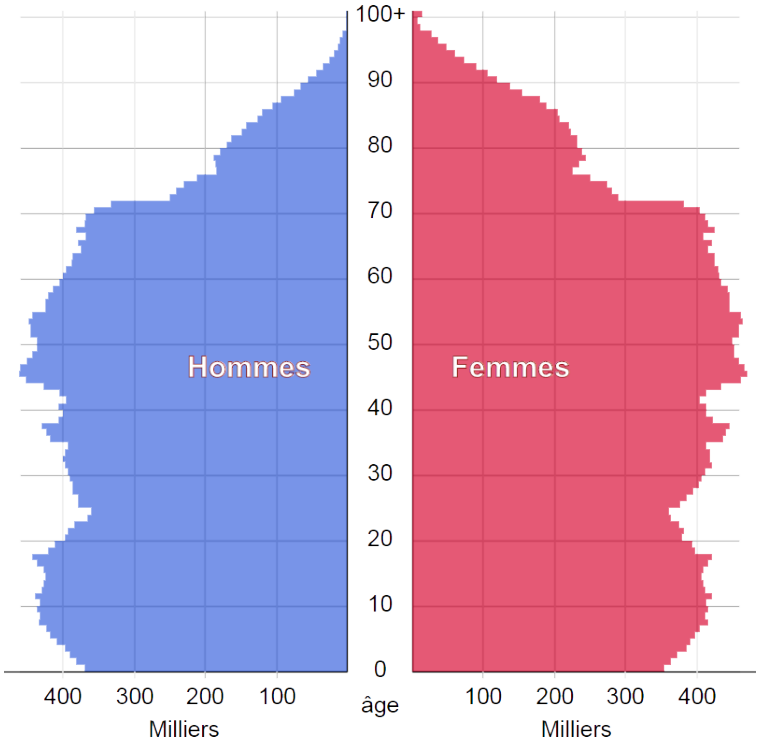


FIGURE 1.1 – Pyramide des ages en France, au 1er Janvier 2018 (Chiffres Insee)

De plus les accidents à domicile concernent surtout les enfants et les personnes âgées, les chutes étant le premier facteur de décès pour les accidents à domicile (Figure 1.2). En effet 88% des décès par chute concernent les seniors [1], mais également 70% des chutes de personnes âgées ont lieu à domicile. Le maintien à domicile des personne âgées en sécurité passe donc par une prévention ou du moins une détection des chutes de ces dernières. Les chutes à problèmes sont liées pour la plupart à leurs séquelles, par exemple si la personne n'arrive pas à se relever ou à demander de l'aide. En plus d'affecter l'assurance morale et l'état psychologique de la personne, cela affecte indirectement son physique, car la personne se bride par peur d'une nouvelle chute, ce qui a tendance à accroître les risques d'une nouvelle chute. Il est donc nécessaire de pouvoir prévoir les chutes, ou le cas échéant de pouvoir apporter un sentiment de protection, et de suivi des personnes âgées vivant seules (en couple ou non) au quotidien. La sécurité apportée par un système ambiant, une maison intelligente, est susceptible de renforcer l'assurance d'une personne âgée ainsi que sa façon d'agir au quotidien sans avoir la peur du danger et de ses répercussions. Ce regain ou ce maintien d'activité pourrait vraisemblablement être un moyen de lutter

également contre l'isolement social de la personne et a fortiori contre les chutes. En effet cette dernière serait plus prompte à sortir, ou du moins à avoir une activité physique plus régulière, et la perte de musculature étant plus faible, de fait le risque de chute serait décru.

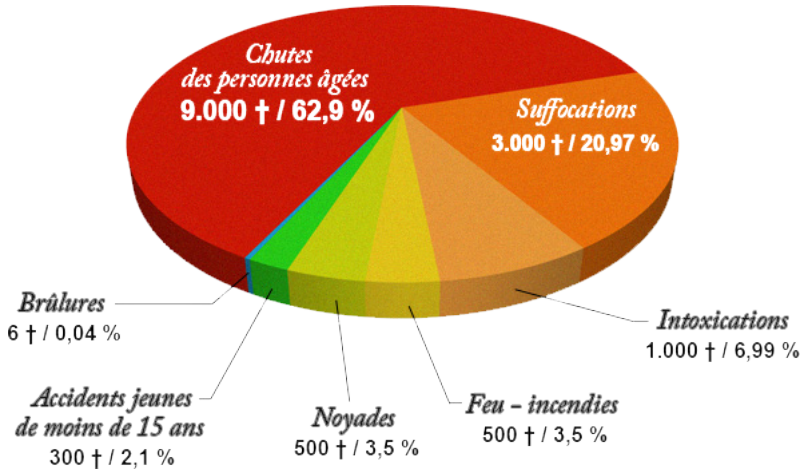


FIGURE 1.2 – Décès suite aux accidents de la vie courante (2010) (Inserm)

1.1.2 Reconnaissance et suivi des activités des personnes âgées par télésurveillance

La télésurveillance/téléassistance pour les personnes âgées a tout d'abord commencé par de simples dispositifs portatifs à boutons d'alerte à porter autour du cou ou du poignet. Ces systèmes permettent à la personne en situation de détresse de pouvoir signifier celle-ci afin d'être mise en relation avec un centre de traitement qui alertera les services nécessaires et appropriés. Nonobstant cette affirmation précédente, qui ne correspond qu'à seulement une minorité des cas ; les centres de traitement, le plus souvent, n'arrivent pas à déterminer la cause de la détresse de la personne âgée (le plus souvent dû à un manque partagé d'efficacité de communication, reconnue comme le problème de levée de doute), et envoient le SAMU par défaut. Cela a pour double effet néfaste de surcharger les services d'urgences et de rendre l'utilisation de la télé-assistance mal vu ou inefficace d'après les familles et les bénéficiaires. Toutefois, une amélioration de ce système notable est l'ajout d'accéléromètres et de capteurs de verticalité qui permettent une détection de chutes automatisée. Cependant cette amélioration ne répond pas au problème des chutes dites molles (lorsque la personne ralentit sa chute en se retenant à un objet ou en glissant le long d'un mur ou apparenté).

De plus cela oblige la personne à porter le système en permanence.

L'utilisation d'un système de télé-surveillance pour la reconnaissance et le suivi des activités d'une personne âgée, peut permettre de détecter les facteurs de risque de chute, ou suite à une chute de pouvoir aider les différents acteurs à comprendre cette chute pour la traiter avec efficience.

La détection de chutes étant extrêmement complexe avec des capteurs peu invasifs et de faible précision, il est plus aisé de suivre les activités potentielles d'une personne âgée en fonction de la pièce d'action et du temps d'occupation de cette pièce. Ainsi il est possible de remarquer les dérives d'activités par rapport à la moyenne, donc de déterminer une anomalie potentiellement dangereuse pour la personne et de déclencher une alerte préventive ou directe en fonction de la gravité de cette anomalie. Cette détection pourra ensuite alerter la personne ou les personnes pouvant lui venir en aide et ainsi permettre de prévenir la personne et ses proches du danger possible. Afin de rendre cette prévention efficace il est nécessaire de la faire dans une optique d'aide et d'amélioration du comportement de la personne. En effet, si celle-ci est alarmiste elle sera contre productive et incitera la personne à se refermer et effectuer moins d'activité ce qui amplifiera d'autant plus le problème détecté.

1.2 L'entreprise KRG Corporate et la solution commercialisée

L'entreprise KRG Corporate propose une solution de télésurveillance à domicile (Figure 1.3) sous la marque SeniorAdom depuis novembre 2012. Cette solution imaginée avec le concours de professionnels de la santé pour les seniors, se compose de capteurs de mouvements, de capteurs de porte ainsi que d'un capteur de présence au lit, et si la personne le souhaite d'un bouton pressoir à porter.

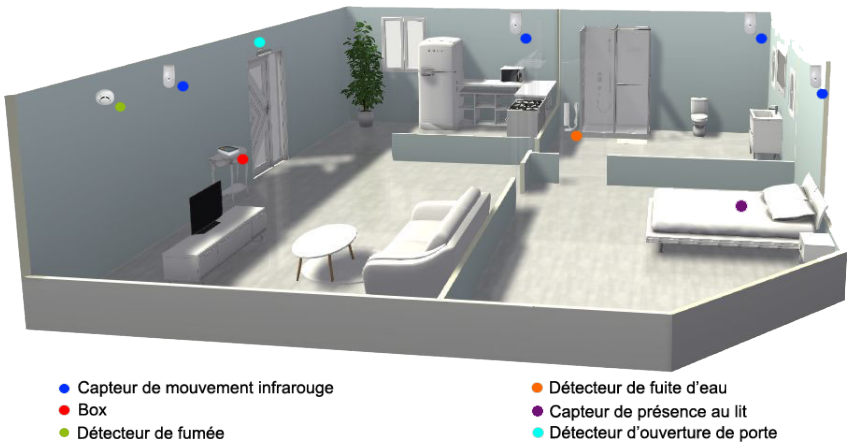


FIGURE 1.3 – Solution de télésurveillance SeniorAdom et les différents capteurs

Ce type de système permet de suivre les activités de la personne au quotidien et il est fondé sur l'hypothèse qu'une personne est installée dans une routine plus ou moins fixe. Le système peut donc dresser les activités d'une personne et effectuer une détection d'anomalie d'activité. Le processus de détection d'anomalies prend du temps et n'est pas en mesure d'alerter les secours dans un temps restreint. Cela permet toutefois d'éviter les complications dues au chutes où la personne est dans l'impossibilité de se relever ; ce type de chutes représentant les chutes qui ont les complications les plus graves et qui entraînent le plus souvent une perte d'autonomie totale ou partielle.

1.3 La chaire eBioMed

La chaire eBioMed (Figure 1.4) s'inscrit dans la démarche précédemment expliquée, qui consiste à utiliser les objets connectés dans un but de santé, dans la lignée du nombre croissant d'applications naissantes de l'internet des objets pour la santé. Bien que la plupart de ces objets ne peuvent répondre à l'appellation de dispositifs médicaux, et ne remplissent pas les normes régissant cette appellation, ils ouvrent de nouveaux horizons pour la santé connectée. Ces horizons sont le cœur du travail de la chaire eBioMed.

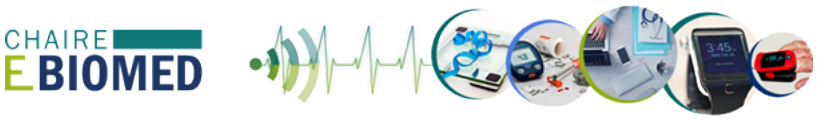


FIGURE 1.4 – Logo et domaines d'activités de la chaire eBioMed

Ces perspectives s'étendent de la télé-expertise délivrée par un spécialiste médical au télé-diagnostic, en passant par la télé-surveillance médicale des personne sur leur lieu de vie. Ces méthodes constituent une réponse potentielle face aux défis sociétaux que constituent le vieillissement de la population, et la hausse constante de personne touchées par des maladies chroniques, le tout en réduisant le coût des systèmes de santé.

Depuis septembre 2014 la chaire eBioMed cherche à développer des outils connectés novateurs, pour la prise en charge du patient à domicile. Les travaux au sein de cette chaire ciblent trois thématiques :

- la détection du risque de naissance prématurée pour les grossesses à risques, en constant accroissement (environ soixante mille par an en France) ;
- le suivi médical à distance de maladies chroniques et des polypathologies qui y sont liées, en particulier du diabètes – trois millions de personnes en France – et les conséquences des AVC (Accident Vasculaire Cérébral) – un toutes les quatre minutes en France– ;
- la télé-surveillance des personnes âgées pour favoriser leur maintien à domicile, mais aussi améliorer et personnaliser leur prise en charge en cas de nécessité ;
- la rééducation fonctionnelle à domicile à travers des jeux sérieux pour pallier le manque de kinésithérapeutes.

1.4 Télécom SudParis - SAMOVAR/ARMEDIA

L'équipe de recherche ARMEDIA appartenant au laboratoire SAMOVAR (UMR - CNRS) travaille, depuis plus de 15 ans, sur le problème de la détection et de le prévention de chutes et a développé différents dispositifs et systèmes : terminal ambulatoire de détection de chutes [5], systèmes de fusion pour réduire le taux de fausses alarmes tout en améliorant la sensibilité de détection. ARMEDIA a également des activités de recherche dans le domaine de l'e-santé comme l'analyse de l'écriture, de la voix, et des comportements, dans le but de détecter des signes de pathologies neurodégénératives (maladies d'Alzheimer et de Parkinson). D'autre activités de recherches sont centrées sur l'imagerie médicale des organes en 3 dimensions (cœur, système digestif) et sur les neurosciences.

1.5 Objectifs et problématiques de la thèse

Afin d'améliorer son système d'alertes et de limiter l'invasivité de son système l'entreprise KRG Corporate a proposé d'utiliser un capteur sonore, un microphone, afin de caractériser et d'identifier les activités de la personne ou, le cas échéant, la détresse potentielle d'une personne. Nous travaillerons donc sur le son, depuis la détection d'un événement acoustique jusqu'à la reconnaissance de ce son et de l'activité pouvant le produire.

Le contexte de cette thèse est donc non seulement contraint par des nécessités techniques mais aussi théoriques. En effet si l'on fait l'analogie avec les systèmes de reconnaissance de parole qui avec des scores de 95% à 97% de bonnes reconnaissances peuvent sembler très bons. Malgré leur performances ils demeurent non utilisables car le nombre d'erreurs, aussi faible soit-il, était trop fréquent pour avoir un système de reconnaissance de parole utilisable au quotidien, comme nous pouvons le faire avec *Ok Google*, *Cortana* ou *Siri*, qui proposent un taux de bonnes reconnaissances de plus de 99.5%. Cette précision est atteignable grâce aux corrections et aux prédictions dues au fait que la parole suit un ensemble de règles grammaticales et structurelles. La reconnaissance de sons, bien qu'étant un champ de recherche proche de la reconnaissance de parole ne peut atteindre de tels résultats en ignorant le contexte applicatif. En effet, il est difficile de reconnaître un son pour un humain sans connaissance du contexte [52]. De plus le manque de structure dans une succession de sons, mais aussi la limitation du vocabulaire afin de définir et décrire un son, sont des problèmes qui peuvent influer fortement sur la précision des systèmes de reconnaissance de sons.

Cette reconnaissance de sons sera, de par sa nature, vouée à être installée chez le client et devra donc pouvoir effectuer une reconnaissance de sons temps réel [1]. Un système temps réel étant défini par sa capacité à répondre à un problème dans un temps défini sans jamais déborder. Notre système étant voué à travailler en continu et sans interruption nous pouvons parler d'un système temps réel lorsque ce dernier est en mesure de ne pas prendre de retard dans sa détection, ou du moins enclin à limiter ce retard et à le résorber dans un temps raisonnable. Ces valeurs n'étant pas quantifiables, nous pouvons dire heuristiquement qu'il est donc question de pouvoir traiter un son dans un temps au maximum deux fois plus long que sa durée, car le nombre d'occurrences de sons potentiellement exploitables et intéressants est très faible, et permet de résorber très rapidement le retard que le système serait capable de prendre. Les sons à traiter ont une durée maximale de 5 à 10 secondes, ce qui nous fait une contrainte de 20 secondes au maximum pour effectuer la classification. Ce temps, bien que long en apparence, reste convenable pour permettre d'alerter les services de secours et leur permettre de se rendre sur place, les temps moyens d'intervention du SAMU étant de 8 à 23 minutes en fonction de la région concernée. Nous

1. Nous parlons bien ici de reconnaissance temps réel au sens informatique du terme, et non de reconnaissance en temps réel.

verrons dans la suite du document que les systèmes proposés sont capables d'effectuer la classification en moins d'une seconde, la contrainte de 20 secondes de traitement par son n'étant qu'une contrainte applicative assurant que le système sera stable et ne perdra pas de données, et sera toujours en mesure de traiter un nouveau son dans ce délai.

Il est donc question de mettre en place un système de reconnaissance de sons le plus précis possible, temps réel, mais aussi de prévoir son implication dans le processus décisionnel du système existant en fonction de la certitude accordée à la réponse apportée par la reconnaissance de sons. Le tout en proposant un système éthiquement compatible et donc non invasif du fait de l'usage d'un traitement de reconnaissance de sons automatisée. Le système devant avoir un faible coût, et avec le minimum d'installation possible il devra être capable de donner de bons résultats dans un environnement hostile, il ne proposera qu'un seul microphone placé dans l'appartement du bénéficiaire dans un endroit choisi par la personne, cet endroit sera par conséquent potentiellement non optimal. Nous devrons donc proposer des solutions pour travailler en environnement bruité et très hétérogène. En effet, d'un appartement à l'autre la variabilité des sons de la vie courante est grande en fonction des habitudes de vie de la personne, mais aussi de l'acoustique d'un appartement donné. De plus le système proposé devra travailler en environnement bruité. Il n'est pas rare d'avoir une télévision ou une radio allumée en permanence, mais aussi des aléas sonores de la vie quotidienne (machine à laver, bruit de la rue, etc.). C'est pourquoi nous devrons également prendre en compte ces deux aspects dans la réalisation du système proposé.

1.6 Questionnements éthiques sur un système de reconnaissance automatique de sons

Lorsque l'on propose d'installer des capteurs chez une personne il est nécessaire d'aborder le problème d'un point de vue éthique. En effet, il serait dangereux d'installer un système, qui serait capable d'enregistrer les sons captés chez une personne et de permettre à un tiers (humain ou machine) de pouvoir les récupérer librement. D'un point de vue sécurité informatique il est possible d'encrypter les fichiers de sons pour les traiter de façon déportée ou du moins de les stocker sous forme paramétrique (les MFCC (*Mel-Frequency Cepstral Coefficients*) et autres paramètres explicités en section 2.3.1) qui présente l'intérêt de ne pas être réversible.

De plus, il est malhonnête de faire rimer sécurité et diminution des libertés ainsi que de la vie privée. Ce système a pour but d'effectuer une supervision bienveillante et doit être installé chez la personne, avec la charge de déterminer une potentielle détresse. Il est donc primordial de ne pas permettre de détourner un tel système : un tiers ne doit pas pouvoir recueillir ou surveiller la personne

bénéficiant de cette solution. C'est pourquoi il est à la charge du concepteur ou du développeur de créer ce système dès le début en suivant des principes éthiques [3] (en anglais cette pratique est appelée : *ethics by design*). Cette pratique a pour but de travailler dès le lancement d'un projet, de réfléchir aux travers possibles ainsi qu'aux usages détournés non souhaités et de les intégrer directement dans la conception et la création du projet. Cette approche proactive de problèmes éthiques d'un projet, est de plus en plus privilégiée par rapport à une approche réactive visant à corriger les problèmes éthiques d'un projet une fois ceux-ci se présentant, souvent en réaction à une mauvaise publicité (*bad buzz*). Mais aussi de limiter tout débordement de l'utilisation de données collectées dans un but mercantile et/ou d'outil d'aliénation de masse, qui peuvent et sont souvent créateurs eux aussi de mauvaise publicité.

Le système que nous souhaitons mettre en place ayant un but d'aide à la personne et de prévention, peut être détourné et pourrait également permettre de surveiller automatiquement les activités des personnes à grande échelle. C'est pourquoi nous nous devons de limiter cette possibilité durant sa conception. Cependant cet horizon, bien que plausible n'est pas et ne doit pas être limitant ou bloquant pour la mise en place d'un tel projet. En effet, de mon point de vue, si un projet peut avoir un effet néfaste sur la société, ne rien faire est l'une des pires solutions. C'est au concepteur d'un tel système de limiter ce dernier afin de proposer une solution éthique. Le concepteur ne doit pas s'appuyer sur de fausses excuses de l'acabit de celle évoquée au début de ce paragraphe : "Ce système est prévu dans un but bienveillant" [39]. Si l'application d'une technologie est possible, il y aura toujours quelqu'un pour l'utiliser. C'est donc au scientifique, au concepteur, souhaitant limiter les effets néfastes de cette dernière, de résoudre les problèmes qui se présentent le plus éthiquement possible tout en limitant au maximum les retombées néfastes [38].

Pour suivre au mieux les concepts sus-cités le système devra agir comme une boîte noire en déterminant et en ayant pour seule sortie le niveau de danger potentiel de la personne. Cette boîte noire influencera le système auquel elle sera rattachée, en fonction des sons captés et des activités courantes supposées. Le fait d'utiliser un serveur décentralisé de traitement pour la reconnaissance de sons n'est pas limitant, cependant ce dernier ne doit pas recueillir les sons mais seulement leurs formes paramétriques. Ces paramètres ne doivent pas contenir d'information permettant d'identifier le système émetteur. Le serveur de traitement de la reconnaissance de sons doit donc être circonscrit à cette seule tâche.

Cette solution de fonctionnement par boîte noire opaque, pose le problème de responsabilité. Nous nous attarderons principalement sur la responsabilité légale. En effet il devient impossible de retracer le fonctionnement et la prise de décision d'un tel système. Actuellement, d'un point de vue légal les entreprises se déchargent de toute responsabilité et l'acquéreur en devient le seul responsable. Cependant les intelligences artificielles deviennent de plus en plus performantes et intelligentes, tout en devenant dans le même temps de plus en plus géné-

ralistes. Il sera donc, prochainement, nécessaire d'appréhender une intelligence artificielle comme être à part entière responsable d'un point de vue légal. Pour le moment il existe plusieurs système multi-acteurs ressemblant en législation, si l'on renomme le développeur en "parent", l'utilisateur en "employeur" ou en "professeur", et le système intelligent en "enfant" ; nous pouvons entrevoir le problème de la responsabilité partagée en fonction des différents acteurs. Chacun ayant un rôle dans l'apprentissage du système ou "enfant" et possiblement dans l'aboutissement à une décision, une action plutôt qu'une autre. En schématisant le problème légal, nous pourrions donc poser les questions suivantes. À partir de quel moment un système intelligent devient-il majeur ? En admettant que ce système possède un libre arbitre, dans quel cas une décision spontanée relève-t-elle de ce dernier ?

En effet de plus en plus de scientifiques et philosophes s'accordent à dire que "l'intelligence humaine n'est pas une spécificité humaine" en se basant sur le caractère indémontrable de cette dernière [19]. Par exemple, pour le cas de la conscience on la distingue en deux catégories : la conscience phénoménale et la conscience d'accès [7]. La conscience phénoménale repose sur l'interprétation subjective d'un événement subi et incommunicable, par exemple une douleur. Dans ce cas il est possible d'imaginer ce que l'on aurait pu ressentir à ce moment mais il est également impossible de prouver le fait de posséder ou non une conscience phénoménale. C'est pourquoi on ne peut prouver qu'une machine ou intelligence artificielle est dénuée de conscience phénoménale. La conscience d'accès quand à elle est le fait de pouvoir rapporter une perception en la décrivant. Actuellement cette dernière n'est pas mise en place dans les systèmes dits intelligents. Une fois que cette conscience sera mise en place, le paragraphe précédent sera moins sujet à une interprétation pouvant s'apparenter à de la science-fiction. C'est pourquoi il est nécessaire dès maintenant de définir et d'appréhender les systèmes intelligents comme entité à part entière mais aussi de les créer avec des principes éthiques dès le commencement et d'intégrer cette démarche tout au long de leur développement.

1.7 Organisation du document

Dans le but de répondre aux problématiques soulevées durant ce chapitre, nous proposons l'agencement suivant pour ce manuscrit de thèse.

Dans le chapitre 2 nous décrirons toutes les techniques et outils actuels pour la reconnaissance d'événements sonores, après avoir défini ces derniers. Nous proposerons aussi en section 2.3.1 un nouvel outil utilisable pour définir un son.

Le chapitre 3 présentera l'utilisation d'une technique de reconnaissance de locuteur adaptée à la reconnaissance de sons. Nous proposerons donc un premier système utilisable utilisant cette technique. Le système proposé sera égale-

ment testé selon différents types et niveaux de bruits, afin de tester sa résilience en environnement bruité. Nous proposerons également une méthode de fusion d'évaluateurs afin d'améliorer la précision obtenue par rapport à l'utilisation d'une seule méthode d'évaluation.

Le chapitre 4 commencera par une présentation succincte des réseaux de neurones. Ces derniers étant proposés pour remplacer les différentes méthodes d'évaluations parcourues au chapitre précédant. De manière analogue, nous testerons cette nouvelle proposition avec les mêmes types et niveaux de bruits afin de pouvoir comparer ce nouveau système avec le précédent.

Le chapitre 5 complémentera la présentation des différents réseaux de neurones, notamment de l'apprentissage profond (*Deep Learning*). L'idée étant de proposer un nouveau système n'utilisant que cette technologie et de le comparer avec les deux propositions de systèmes précédemment explicités, en testant ce nouveau système dans les mêmes conditions que précédemment.

Le chapitre 6 testera le système proposé au chapitre 5 en conditions écologiques. Il commencera par la présentation complète du système, ainsi que des moyens de mise en œuvre, tout en explicitant les méthodes proposées pour rendre le système éthique, malgré les contraintes nécessaires à la réalisation de ce test.

Le chapitre 7 explicitera les différentes conclusions afférentes aux méthodes et problématiques soulevées dans ce document. Puis il abordera les différentes perspectives d'utilisations et d'améliorations envisageables.

Chapitre 2

État de l'art de la reconnaissance de sons

2.1 Introduction

La reconnaissance de sons de l'environnement est un domaine de recherche depuis 1990 environ avec les travaux de Bregman [11], mais reste un des domaines de la perception par ordinateur les moins explorés. Si l'on compare la reconnaissance de sons par rapports aux autres domaines de la perception par ordinateur il est aisé de se rendre compte du fossé existant entre cet axe et les autres axes de recherche de ce domaine. Le domaine le plus représenté en machine learning est la reconnaissance d'images avec la reconnaissance d'écriture et de chiffres pour commencer, ce dernier étant devenu le test et le tutoriel standard des algorithmes d'apprentissage avec la base de données MNIST (*Mixed National Institute of Standards and Technology*), et obtenant des résultats extrêmement précis avec des apprentissages très primitifs (≈92% de bonnes reconnaissances avec le premier tutoriel de Tensorflow [1] ; Tensorflow étant l'environnement de Deep Learning proposé en OpenSource par Google). Le meilleur résultat sur cette base de données étant de 99,79% de bonnes reconnaissances [61]. Dans le même domaine de la reconnaissance d'image par ordinateur, on peut noter les tests Cifar-10 et Cifar-100 qui consistent à faire de la reconnaissance d'image sur des photos d'animaux ou de moyens de locomotion. Le test Cifar-10 est composé de 10 classes contenant chacune 6000 images, 6 classes sont des images d'animaux : oiseaux, chats, chiens, biches, chevaux, grenouilles et quatre de véhicules : avions, voitures, bateaux et camions ; à ce jour le meilleur résultat obtenu est de 96.53% [28]. Le test Cifar-100 propose 100 classes de 600 images chacune et le meilleur score obtenu est de 75.72% [17].

Si l'on se recentre sur la reconnaissance par capteurs sonores, on pense en premier lieu à la reconnaissance de parole et aux systèmes proposés par Google

1. https ://www.tensorflow.org/get_started/mnist/beginners

avec *Ok Google !* et Apple avec *Siri* qui sont les deux systèmes de reconnaissance de paroles les plus utilisés à ce jour. On parle dans ce domaine de deux types de reconnaissance de parole, soit ceux en vocabulaire étendu où il est question de retranscrire la phrase prononcée (comme *Ok Google !* et *Siri*), et ceux en vocabulaire restreint qui correspondent à des systèmes de commande où une liste de mots clefs doit être détectée mais pas le sens sémantique de la phrase ; on les utilise notamment dans les standards téléphoniques automatisés ou dans les assistants vocaux "main libres" pour les conducteurs, avec des numéros pour faire un choix ou une instruction telle que *oui, non, annuler, etc.* pour valider ou invalider une action. Les systèmes de reconnaissance de parole sont désormais capables d'atteindre des taux de reconnaissance similaire a celui des humains (sauf en environnement bruité). Mais depuis l'été 2017, Microsoft a également annoncé un taux de reconnaissance supérieur aux humains avec un taux d'erreur de 5.1% contre 5.9% pour un humain [62].

Ces deux domaines identifiés précédemment recouvrent la plus grande partie de la recherche en perception par ordinateur, l'avantage de ces deux domaines est qu'il est aisé de décomposer l'information globale en parties recomposables architecturalement entre elles, pour recomposer l'information à reconnaître. Dans le cas d'une image il est possible de trouver des ressemblances structurelles entre deux images appartenant à une même classes, ainsi que des différences pour distinguer deux classes similaires. Par exemple dans le test CIFAR-10 on peut dire qu'une voiture et un camion possèdent des roues, cependant la distance entre le sol et le haut du véhicule est supérieure pour les camions que pour les voitures, cette distance étant mesurable par exemple à partir de la taille d'une roue. Dans le cas de la parole, il est possible de découper les phrases en mots et les mots en phonèmes, qui représentent l'information portée par le son capté. Ces décompositions possibles permettent également de simplifier et rendre possible la compréhension et la communications des personnes et chercheurs entre eux, ainsi que d'avoir une façon de décrire à la machine le spécimen à reconnaître.

Dans le cas de la reconnaissance de locuteur il est plus difficile de décrire efficacement la personne à reconnaître. Mais il existe quelques descripteurs permettant de réduire la complexité de la représentation finale. Pour les citer de façon non exhaustive : homme ou femme, en fonction de la tranche d'âge : enfant, adolescent, adulte ou senior, le rythme de parole. Ce qui permet d'effectuer un pré-traitement pour aider le système de reconnaissance. Dans la reconnaissance de locuteur on distingue deux méthodes distinctes de reconnaissance, dépendante du texte ou non. Dans le premier cas l'utilisateur se doit de dire une phrase prédéfinie et pré-enregistrée, ce qui permet de guider d'autant plus le système ; cette méthode est la plus utilisée dans les systèmes d'identification biométrique dans la sécurité. L'autre méthode permet de distinguer principalement plusieurs personnes dans un flux audio donné, et donc par combinaison avec la reconnaissance de parole de retranscrire une conversation multi-locuteurs. La précision des systèmes de reconnaissance de locuteur est acceptable depuis 1990 [50] (>95% de bonnes reconnaissances parmi 49 classes) s'appuyant sur les GMM

(Modèle de mélange gaussien – *Gaussian Mixture Model*) et de 98.88% avec les travaux de N. Dehak avec les i-vecteurs (développé dans le chapitre 3) [21] qui sont "assimilables" à une évolution des GMM.

La reconnaissance de sons quant à elle est un sujet de recherche très peu exploré, malgré le fait qu'il soit aisé de concevoir que l'environnement sonore apporte un flux d'informations très important. En effet les bruits divers sont prépondérants dans notre environnement quotidien qu'ils soient des sons d'alertes (klaxon, sonneries, alarmes) ou divers, un son ponctuel qui diffère de l'environnement sonore moyen. Ces sons attirent notre attention et nous les analysons bien souvent de par leur provenance géographique et de par leur type (de son produit). En effet un son inattendu, qu'il soit fort ou non, attirera indubitablement notre attention, puis nous essayerons de deviner ce qui a produit le son et l'urgence d'action nécessaire en fonction de ses caractéristiques : sa proximité, sa direction, s'il est impulsionnel, s'il a une certaine persistance temporelle ou durée. Cependant en reconnaissance de sons, une partie des informations est souvent manquante : la localisation du son est bien souvent impossible. En effet les microphones directionnels et les systèmes de localisation sonores sont souvent chers, fragiles et sensibles à leur qualité d'installation, ce qui les rend ardus à utiliser dans un système destiné au grand public, dans un environnement inconnu. Il est donc question d'effectuer la reconnaissance de sons avec des moyens restreints par rapport à un être humain, sachant que la reconnaissance exacte de sons est également difficile pour un être humain et laisse apparaître un forte incertitude de par la variabilité possible dans l'ensemble des sons environnants [52].

Les travaux sur la reconnaissance de sons sont également disparates car très ciblés, ce qui explique les difficultés et le peu de visibilité sur la reconnaissance de sons, par exemple on trouve des travaux sur la reconnaissance de l'environnement [14], le reconnaissance de sons distincts par niveau de bruit [24], la reconnaissance d'altercations dans un environnement urbain [4], la reconnaissance de sons dans une situation donnée [55][12]. Cette dissémination des différents travaux s'explique par le fait que tout ce qui n'est ni de la parole, ni de la musique, est regroupé sous l'appellation de son, mais également car l'information portée par un son est consubstantiel à l'environnement dans lequel ce son est perçu. Nous nous intéresserons donc uniquement aux sons pouvant être perçus dans un habitat et pouvant signifier une activité identifiable ou un risque, une détresse pour son occupant.

2.2 Les sons et leur environnement

Actuellement, les systèmes de reconnaissance d'événements acoustiques sont très dépendants de leur système d'application, le nombre de sons à reconnaître étant limité au domaine d'application, les autres sons sont alors considérés comme du bruit. Cependant il serait utopiste, actuellement, de vouloir créer

un système de reconnaissance d'événements acoustiques capable de reconnaître tous les sons. En outre il est chimérique de vouloir établir une liste exhaustive des sons. Nonobstant cette difficulté, un son de l'environnement reste caractérisable ; VanDerveer [59] proposa les critères suivants pour définir un son de l'environnement :

— un événement le produit ;
— il est le reflet d'un ou d'une série d'événements causaux ;
— son traitement est plus compliqué qu'un son pur généré en laboratoire ;
— il ne relève pas de la reconnaissance de parole (plus généralement de communication selon sa définition, la parole n'étant pas le seul type de son nous permettant de communiquer, par exemple les interjection ne font pas partie de la parole, mais sont pour autant un moyen de communication).

Dans ce travail nous considérerons tout ce qui n'est pas de la parole comme son, en effet les cris, les pleurs, etc. sont considérés par cette définition comme faisant partie d'un système de communication humain. Cependant, nous considérerons d'un coté la parole et de l'autre les sons. Les modes de communication humains n'étant pas de la parole feront donc par définition partie des sons. La parole sera elle aussi traitée par notre système sans pour autant chercher à retranscrire le contenu, mais pour déléguer cette tâche à un système de reconnaissance de parole si la nécessité de retranscription est pertinente.

La figure 2.1 est une proposition de regroupements des sons potentiellement retrouvables dans un appartement. Les listes d'exemples proposés dans la figure ne sont pas exhaustives. Nous nous intéresserons principalement dans nos travaux au regroupement des sons naturels pouvant signifier un danger ou une activité normale pour une personne âgée à son domicile.

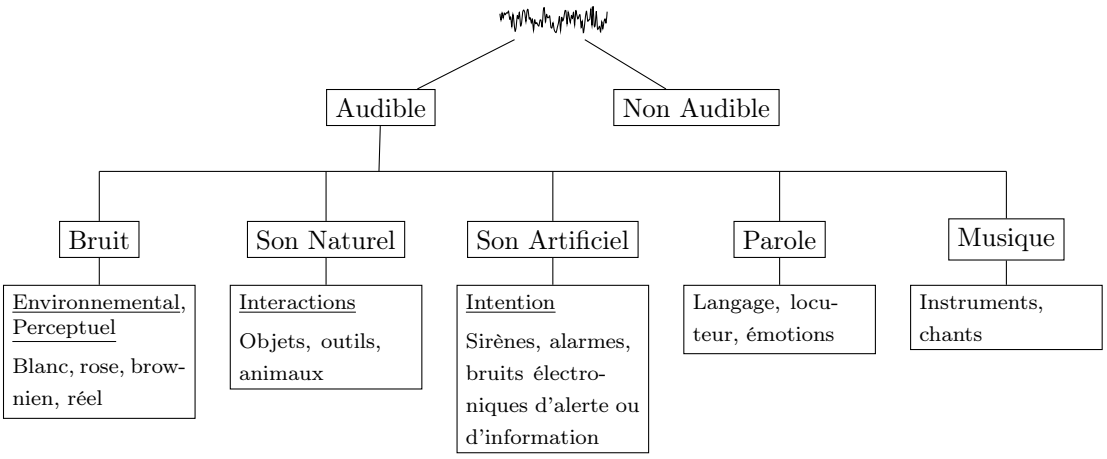


FIGURE 2.1 – Sons pouvant être perçus dans un appartement

De même nous pouvons regrouper les sons qui nous intéressent par caractéristiques. La figure 2.2 présente les différents regroupement par caractéristiques que nous proposons au sein des sons pouvant être perçus dans un appartement.

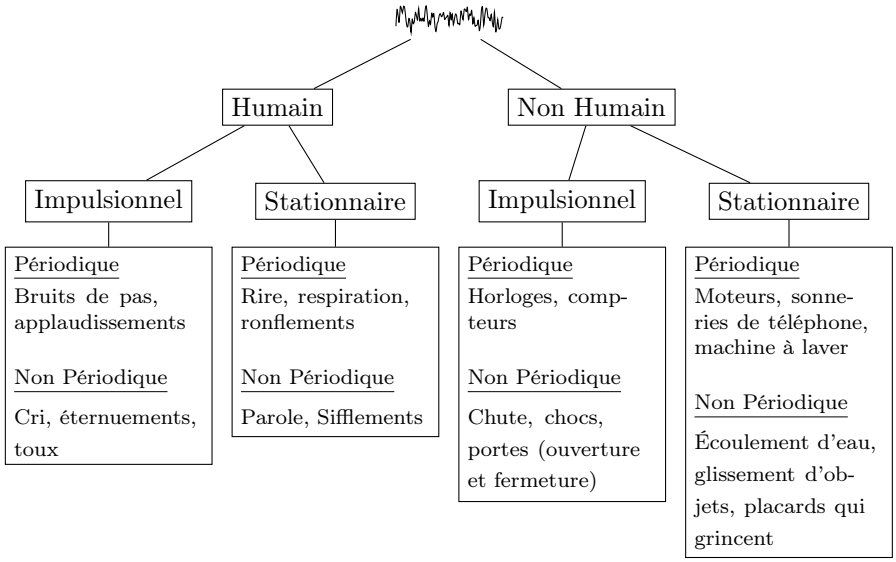


FIGURE 2.2 – Sons potentiellement percevables dans un appartement regroupés par caractéristiques

2.2.1 Détection d'événements en environnement bruité

Le système proposé étant prévu pour être utilisé en continu il est nécessaire de reconnaître un événement acoustique potentiellement intéressant, et ainsi d'essayer de ne traiter que cette partie du signal. En effet, il serait contre-productif d'analyser le son en continu et d'effectuer une reconnaissance en continu. Aussi, se baser sur la segmentation de sons qui résultent d'une variation de l'activité acoustique moyenne de l'habitat semble plus pertinent. Par analogie on peut dire que c'est aussi notre méthode de fonctionnement. Nous ne passons pas notre temps à nous demander quel son vient d'être produit, tous les sons perçus font partie inconsciemment de l'environnement sonore actuel et sont ainsi occultés. Cependant, même si nous sommes au milieu d'une conversation et que quelqu'un prononce notre nom ou qu'un déclencheur [2] quelconque survient, nous allons le repérer même si ce son se fond dans l'environnement sonore. Il en va de même avec un son qui s'ajouterait brusquement. Par exemple une sirène de pompier, un crissement de pneu, un objet qui tombe ou un choc, attireraient indubitablement notre attention afin d'analyser ce son. C'est pourquoi il est judicieux d'avoir un système de détection d'événements acoustiques efficace afin de n'analyser que les échantillons porteurs potentiels d'informations. On pose donc la même hypothèse qu'en reconnaissance de parole : l'échantillon prélevé correspond à un signal à reconnaître, par extension une source sonore. Pour la reconnaissance de parole on cherche à reconnaître une phrase, ou du moins un groupe de phrase d'une même personne. En reconnaissance de locuteur et en reconnaissance de sons on cherche à reconnaître une seule source à la fois.

Afin de segmenter les sons présentant un intérêt, nous utiliserons comme système de référence l'algorithme de détection proposé par D. Istrate [34]. Cet algorithme basé sur une transformée en ondelettes, puis une évaluation de l'énergie de chaque coefficient, permet de détecter le début d'une variation de signal dans l'environnement sonore. Au moment de cette détection un seuil d'arrêt est fixé et un second seuil temporel est fixé arbitrairement afin de déterminer la fin de la variation. Nous avons fusionné les comportements de ces deux seuils d'arrêts en mettant en place un seuil adaptatif, qui est plus faible en début de détection puis qui augmente conjointement au temps écoulé depuis le début de la détection. Cette méthode permet de proposer une découpe plus pertinente de la variation détectée. La figure 2.3 présente les deux algorithmes que nous venons de présenter.

2. i.e. un son n'étant pas attendu, dénotant par rapport à l'environnement, par exemple un bris de verre dans une réception

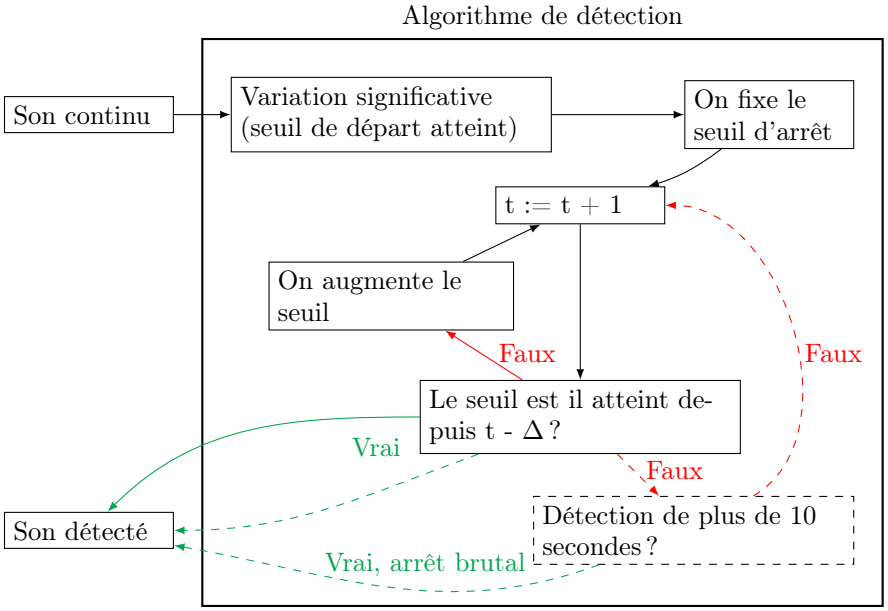


FIGURE 2.3 – Algorithmes de détection présentés (en pointillés l'algorithme de D. ISTRATE)

2.3 Classification d'événements

Une fois l'isolement d'une variation de l'environnement sonore effectuée, il est nécessaire de reconnaître ce son afin de pouvoir en déterminer la source et l'activité de la personne. La classification d'événements fait partie du domaine de l'apprentissage automatisé (*Machine Learning*). Ce domaine est découpé en deux grandes catégories : les approches à apprentissage supervisé et celles à apprentissage non supervisé. La première approche cherche à reconnaître l'échantillon proposé à partir d'une base de connaissance ou de modèles – notre cas – qui est inférée durant son apprentissage. La seconde approche cherche à trouver de manière aveugle des liens et donc des regroupement dans un nuage de données.

Parmi les méthodes de classification propres à la reconnaissance de sons, adaptées de celle de locuteur ou de celle de parole, on peut noter les méthodes statistiques les plus performantes et les plus couramment utilisées suivantes : GMM, SVM-GSL (Machine à vecteurs de support, GMM-super-vecteur à noyau linéaire – *Support vector machine GMM Supervector Linear kernel*) et HMM (Modèle de markov caché – *Hidden Markov Model*). Chacune de ces méthodes de classification peut être décrite selon le schéma générique 2.4. Nous ne nous attarderons pas sur les HMM car ces derniers ne sont pas adaptés, a priori, aux

sons. Cette technique nécessite une structure temporelle définie afin de pouvoir définir des probabilités de succession entre les différents états d'un système. Pour la parole on parle de phonèmes (sonorités, assimilable grossièrement aux syllabes), et l'on peut donc définir une structure avec des probabilité futures et passées pour un phonème détecté donné. Les sons ne possédant pas de structure temporelle définie, il est donc difficile d'utiliser les HMM, nous ne les décrirons donc pas.

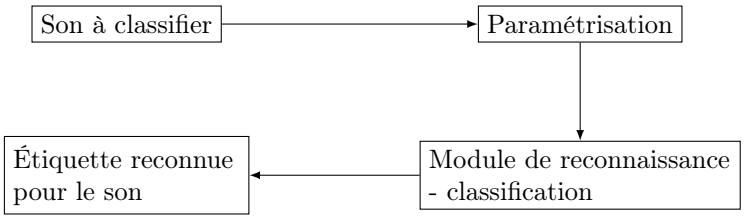


FIGURE 2.4 – Processus standard de reconnaissance de sons

2.3.1 Paramètres acoustiques utilisés

Afin d'utiliser ces méthodes statistiques il est nécessaire d'effectuer un prétraitement du son, se présentant comme un vecteur unidimensionnel possédant un grand nombre d'échantillons par seconde – dans notre cas la fréquence d'échantillonnage est de 16kHz – ; un échantillon étant une valeur entière. Nous utiliserons une méthode de fenêtrage connue sous le nom de fenêtre de HAMMING (Équation 2.1) [30], qui permet d'atténuer les lobes secondaires afin d'améliorer la FFT (Transformée de Fourier rapide – *Fast Fourier Transform*) que nous appliquerons subséquemment. De plus ce type de fenêtrage est recommandé pour les signaux sinusoïdaux avec des courbes rapprochées (les harmoniques), ce qui est le plus courant dans l'analyse de sons.

$$\begin{gathered} w(n) = \alpha - \beta \cos\left(\frac{2\pi n}{N-1}\right) \\ \text{avec, } \alpha = 0.54,\ \beta = 1 - \alpha = 0.46 \end{gathered} \tag{2.1}$$

Une fois le fenêtrage effectué nous pouvons extraire les paramètres acoustiques, dont on distingue 2 catégories de paramètres : temporels et fréquentiels. Parmi les paramètres temporels nous utiliserons et détaillerons le ZCR (*Zero*

Crossing Rate) et le RER (*Remarkable Energy Rate*), qui est un paramètre que nous avons créé pour permettre de différentier plus efficacement certaines classes de sons. Parmi les paramètres fréquentiels nous utiliserons et détaillerons les MFCC (*Mel-Frequency Cepstral Coefficients*), le SRF (*Spectral Rolloff Point*) et le SC (*Spectral Centroid*). Certains de ces paramètres sont inspirés des propriétés biologiques de la paroles et l'ouïe. Par exemple l'échelle de Mel est basée sur la réponse en fréquence de l'oreille humaine, et le cepstre (le spectre du spectre) est basé sur les propriétés du conduit vocal.

MFCC

Suite à ce fenêtrage nous pouvons extraire les MFCC. Ces paramètres seront les principaux paramètres que nous utiliserons dans la suite de ce travail. Ces coefficients sont calculés suite aux différentes étapes présentées dans la figure 2.5.

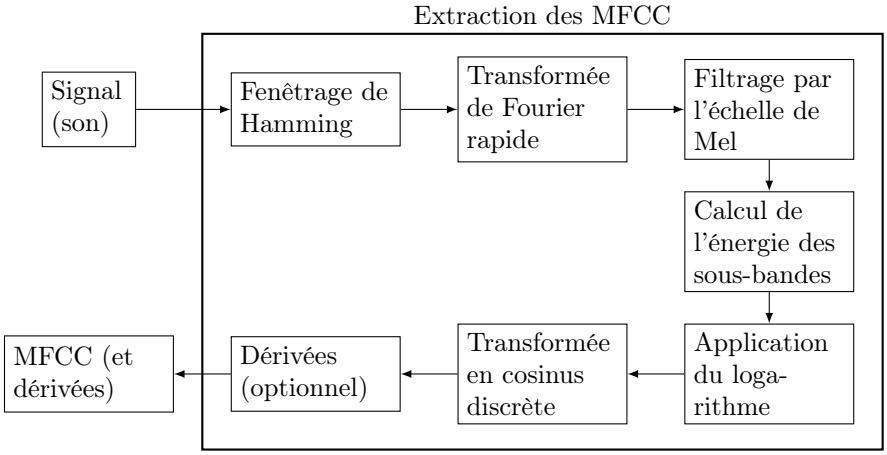


FIGURE 2.5 – Schéma des différentes étapes d'extraction des MFCC

Afin de calculer les coefficients MFCC résultant des différentes étapes présentées, nous commençons par calculer la FFT pour chaque fenêtre suivant l'équation 2.2. Avec N étant le nombre d'échantillons par fenêtre d'indice τ.

$$c_{\tau,k}^{(1)} = \left| \frac{1}{N} \sum_{j=0}^{N-1} s_j \exp\left[-i2\pi \frac{jk}{N}\right] \right| \quad \text{avec, } k = 0, 1, \ldots, (N/2) - 1 \tag{2.2}$$

Il est ensuite question de calculer le spectre d'amplitude selon l'échelle de Mel en utilisant un banc de filtres passe bande triangulaires (Figure 2.6). Cette étape consiste à appliquer l'équation 2.3 avec $d_{j,k}$ étant le filtre triangulaire j appliqué à k.

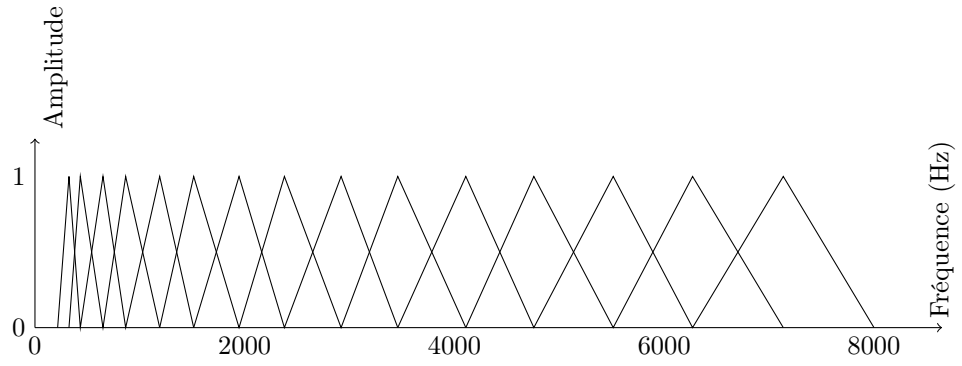


FIGURE 2.6 – Exemple de banc de filtres MFCC avec 15 filtres passe-bande triangulaires

$$c_{\tau,j}^{(2)} = \sum_{k=0}^{\frac{N}{2}-1} d_{j,k} c_{\tau,k}^{(1)} \quad \text{avec, } j = 0, 1, \ldots, N_d \tag{2.3}$$

Nous appliquons ensuite la fonction logarithme suivant l'échelle de Mel (l'équation 2.4 permet de passer de Hertz en mels) au signal de densité spectrale pour imiter la perception humaine par rapport au niveau du son (Équation 2.5) [43].

$$m = 2595 \cdot \log\left(1 + \frac{f}{700}\right) \tag{2.4}$$

$$c_{\tau,j}^{(3)} = \log\left(c_{\tau,j}^{(2)}\right) \tag{2.5}$$

Nous pouvons ensuite appliquer la transformée en cosinus discrète afin d'obtenir les coefficients cepstraux (le cepstre pouvant être assimilé au spectre du spectre). Le cepstre du signal étant obtenu par l'équation 2.6 – avec s le signal temporel d'origine et F la transformée de Fourier, ce dernier nous permettant donc de déterminer les coefficients cepstraux (MFCC) suivant l'équation 2.7. N_{mc} étant le nombre de coefficients cepstraux voulus.

$$c_{\tau,j} = F^{-1}\{log|F(s_n)|\} \tag{2.6}$$

$$\begin{aligned} c^{(4)}_{\tau,j} &= \sum_{j=1}^{N_d} c^{(3)}_{\tau,j} \cos\left[\frac{k(2j-1)\pi}{2N_d}\right] \\ &\text{avec, } j = 0, 1, \ldots, N_{mc} < N_d \end{aligned} \tag{2.7}$$

SRF

Le SRF est un descripteur fréquentiel du signal, il reflète la forme du signal en utilisant les énergies les plus élevées de ce dernier (Figure 2.7). Il représente les parties du spectre où la puissance est supérieure à 95% de la puissance totale, la valeur communément utilisée du SRF étant 95%, peut être modifiable en fonction des sons à traiter afin de mieux gérer la discrimination de ces derniers, cette valeur étant commune pour différencier la voix des autres sons dans le cas de la discrimination sons - parole - musique. Ce paramètre est calculable selon l'équation 2.8, avec $\gamma = 0.95$ et $S_p(k)$ étant la puissance du signal pour une bande de fréquence donnée k.

$$\sum_{k<SRF} S_p(k) = \gamma \sum_{k} S_p(k) \tag{2.8}$$

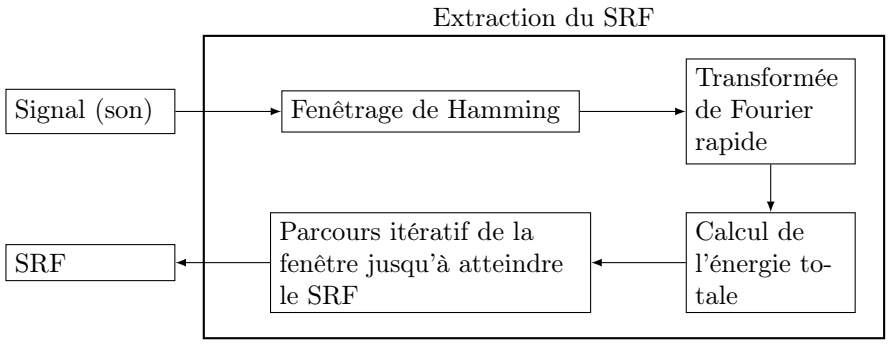


FIGURE 2.7 – Schéma des différentes étapes d'extraction du SRF pour une fenêtre donnée

SC

Le SC est assimilable à un cas particulier du SRF qui correspond au centre de gravité des fréquences pour un signal donné. Il est le plus souvent calculé comme la moyenne pondérée des fréquences du signal pour une fenêtre donnée t (Équation 2.9). Communément on utilise la formule 2.8 avec $\gamma = 0.5$.

$$SC_t \triangleq \frac{\sum_{i=1}^{N_t} f_i S_t(f_i)}{\sum_{i=1}^{N_t} S_t(f_i)} \tag{2.9}$$

ZCR

Le ZCR est un descripteur temporel, il permet de représenter la fréquence principale du signal et il est caractérisé par le nombre de fois où le signal change de signe (Équation 2.10) pour une fenêtre donnée t (Figure 2.8). Usuellement un écart est toléré autour de 0, c'est pourquoi on applique un delta à la fonction de signe afin d'éviter d'avoir la fréquence principale du bruit lorsque le bruit est faible.

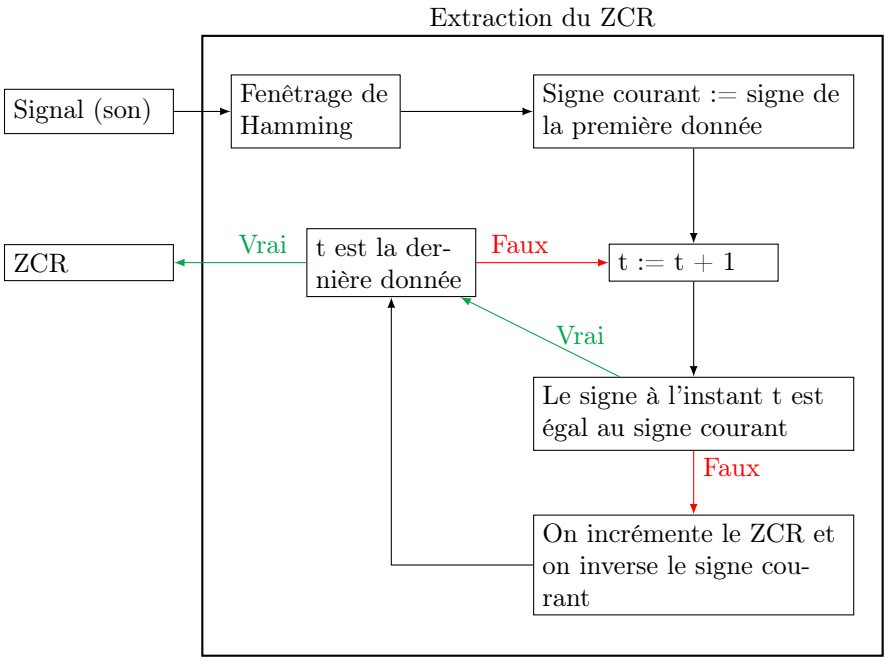


FIGURE 2.8 – Schéma des différentes étapes d'extraction du ZCR

$$ZCR_t \triangleq \frac{1}{2L}\sum_{\tau=1}^{L} |sgn\left(s_t(\tau)\right) - sgn\left(s_t(\tau-1)\right)| \qquad (2.10)$$
$$\text{avec, } sgn\left(s_t(\tau)\right) = \begin{cases} 1, & \text{si } s_t(\tau) \geqslant 0 \\ -1, & \text{si } s_t(\tau) < 0 \end{cases}$$

Création d'un nouveau paramètre acoustique – RER

Afin de différencier certains types de son entre eux il est nécessaire d'utiliser des paramètres acoustiques supplémentaires, par rapport à ceux utilisés en reconnaissance de locuteur ou de parole. A l'exemple des paramètres ZCR, SC et SRF utilisés lors des travaux de D. ISTRATE [34] et de M. SEHILI [53], nous avons introduit un nouveau paramètre : le RER présenté à l'équation 2.11 [51]. Ce paramètre est dérivé de l'enveloppe du signal ; il représente l'enveloppe du si-

gnal lorsque l'énergie, de ce dernier, est la plus importante. Il appartient à donc la famille des descripteurs temporels. Le RER utilise un seuil noté θ, calculé à partir du plus faible maxima et d'une valeur souhaitée ζ que nous avons fixée à 50% dans nos travaux, il peut être modifié en fonction du type d'enveloppe que l'on souhaite avoir. Ce paramètre permet de discriminer les sons à forte variation relativement à son énergie moyenne, de ceux plus constant autour de leur énergie moyenne. L'algorithme d'extraction est représenté en figure 2.9.

$$
\begin{aligned}
RER &= \sum_{\tau=0}^{L} rer(\tau) \\
\text{avec, } rer(\tau) &= \begin{cases} 1, & \text{si } s_t(\tau) \geq \theta \\ 0, & \text{si } s_t(\tau) < \theta \end{cases} \\
\text{et, } \theta &= min(|max(signal)|, |min(signal)|) * \zeta \\
&\text{avec, } s_t \text{ l'échantillon à } \tau
\end{aligned}
\tag{2.11}
$$

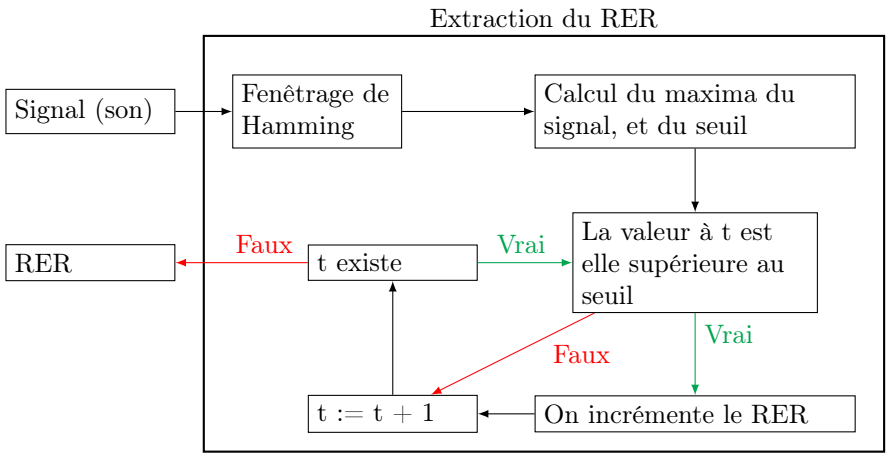


FIGURE 2.9 – Schéma des différentes étapes d'extraction du RER

2.3.2 GMM

Une densité de probabilité gaussienne est une distribution statistique représentée par deux variables, sa moyenne μ et sa variance ou écart-type σ (Équation

2.12), Cela signifie que les données sont donc réparties de façon égale à droite et à gauche de la courbe par rapport à la moyenne.

$$y = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \tag{2.12}$$

La distribution de Gauss est donc la plus couramment utilisée pour représenter des données mono-dimensionnelles ne présentant qu'un seul sommet dans leur répartition statistique. Lorsqu'il y a plusieurs sommets il est nécessaire de les représenter avec plusieurs gaussiennes, l'agrégation peut être faite grâce à ce que l'on appelle un GMM, qui est donc une nouvelle distribution de probabilité résultant de multiples distributions de probabilités, on peut en voir un exemple sur la figure 2.10 où les distributions gaussiennes sont en couleur et le GMM résultant est en noir.

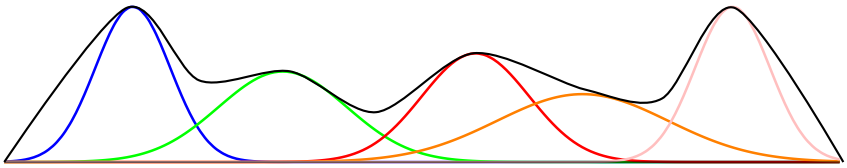


FIGURE 2.10 – Exemple de GMM mono-dimensionnel

Un modèle de mélange gaussien est donc un modèle statistique représentant une distribution statistique, dans notre cas des sons, appelé densité mélange [25][57][48]. Lors de l'apprentissage, il est question de déterminer à l'aide du critère de maximum vraisemblance – le plus souvent effectué par l'algorithme itératif EM (Algorithme d'espérance-maximisation – *Expectation-Maximization*) – la variance, la moyenne et le poids de chaque gaussienne afin de générer le GMM. La relation 2.13 représente la mise en équation d'un GMM, avec x un vecteur de coefficients de dimension D, w_i représente le poids de chaque coefficient i, μ_i la moyenne, Σ_i la matrice de covariance, et $N(x|\mu_i, \Sigma_i)$ la composante de densité gaussienne correspondante (Équation 2.14), où x correspond aux observations des paramètres cepstraux.

$$p(x|\lambda) = \sum_{i=1}^{K} w_i N(x|\mu_i, \Sigma_i) \tag{2.13}$$

$$N(x|\mu_i, \Sigma_i) = \frac{1}{(2\pi)^{D/2}|\Sigma_k|^{1/2}} \; exp\left(-\frac{1}{2}(x-\mu_k)^T\Sigma_k^{-1}(x-\mu_k)\right) \qquad (2.14)$$

Cette phase correspond donc à la phase d'apprentissage d'un modèle GMM. La phase de classification cherche à déduire pour un échantillon donné la classe la plus probable à partir du calcul de la probabilité de vraisemblance d'une fenêtre de signal donné [23][8], puis d'utiliser une moyenne géométrique pour obtenir la vraisemblance sur tout le signal. En pratique, les GMM ne sont pas appliquées sur le signal temporel mais sur une transformée en paramètres acoustiques du signal, suite à une découpe de ce dernier par fenêtrage, le plus souvent par un fenêtrage de Hamming par tranches de 16ms, cette taille permet de garantir la stationnarité du signal, avec un recouvrement de 50%. Les paramètres acoustiques les plus utilisés sont les MFCC et leurs dérivées premières et secondes. En reconnaissance de sons on utilise également quelques paramètres additionnels [34] décrits en section 2.3.1.

2.3.3 SVM-GSL

Les SVM-GSL sont un dérivé des SVM (Machine à vecteurs de support– *Support vector machine*). Les SVM-GSL utilisent en effet une méthode statistique, les GMM couplée à une méthode discriminante, les SVM.

SVM

Les SVM [18] ont donc un principe de classification différent des GMM. Durant la phase d'apprentissage on cherche à trouver l'hyperplan maximisant la marge entre deux classes (Figure 2.11). C'est-à-dire trouver l'hyperplan qui minimise les erreurs de classification dans un nuage de points entre deux classes, par maximisation des marges de part et d'autre de l'hyperplan séparateur. En classification multi-classes, il faudra donc adapter la méthode SVM qui ne permet que de classer une donnée dans l'une des deux classes. Deux méthodes sont le plus communément utilisées, *one-versus-all* et *one-versus-one*.

Dans le cas du *one-versus-all* on crée N classifieurs SVM, avec N étant le nombre de classes, où chaque classifieur est composé de deux classes, une étant une classe n et l'autre étant le regroupement de toutes les autres. On choisit donc dans ce cas la classe ayant la meilleure note de confiance (marge).

Dans le cas du *one-versus-one* on crée $\frac{N(N-1)}{2}$ classifieurs, dans ce cas on procède au vote majoritaire, c'est-à-dire que la classe ayant remporté le plus de duels est celle qui sera la classe choisie.

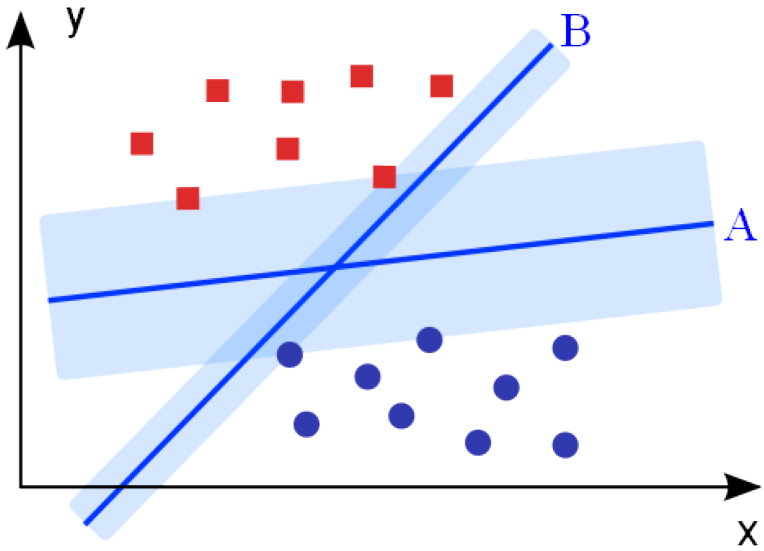


FIGURE 2.11 – Exemple d'hyperplans séparant deux classes, ici le plan A est celui qui maximise (par rapport à B) la marge

SVM-GSL

Le noyau SVM-GSL est un système SVM adapté aux sons au sens large. Chaque point représente un son dans le domaine de la reconnaissance de sons. En effet chaque son sera extrait sous la forme d'un GMM, puis le système SVM-GSL [53] cherchera l'hyperplan entre les deux classes évaluées en cours. Lors de la phase de classification le système déterminera une probabilité d'appartenance à chaque classe en testant toutes les classes deux à deux. L'utilisation d'un système SVM demande des vecteurs de dimension très élevée et entraîne donc un fort coût machine durant le processus car il est nécessaire pour chaque son donné de créer le GMM associé et donc d'effectuer un apprentissage GMM par son.

2.4 Conclusion

Il existe différents systèmes de reconnaissance de sons par exemple les travaux de S. CHU sur la reconnaissance de l'environnement [14] avec un taux de bonnes reconnaissances de 82% sur 14 classes de sons, ainsi que les travaux de A. DUFAUX sur la reconnaissance de sons en environnement bruité [24] qui utilisent un système statistique avec un taux de bonnes reconnaissances de 98%

sur 6 classes de sons.

A l'heure actuelle deux systèmes ont été proposés [34][53] pour la reconnaissance de sons chez les personnes âgées. Un premier utilisant des GMM et un second utilisant les SVM-GSL. Ces deux systèmes présentent une précision respectivement de 71.1% et de 75.4%. Cependant le système SVM-GSL qui propose la meilleure précision nécessite une forte puissance de calcul et de mémoire dû à l'apprentissage du GMM pour chaque son à classer.

L'entreprise KRG Corporate, qui souhaite intégrer à sa solution la reconnaissance de sons, ne peut donc utiliser cette seconde méthode, car l'augmentation de la mémoire ainsi que la puissance embarquée pour utiliser cette méthode ferait exploser le coût final du système. Nous devrons donc pour la classification de sons proposer un système avec un coût de calcul limité et précis répondant à des contraintes de temps réel. Ce système doit également être robuste au bruit, pour éviter une trop grande perte de précision et donc d'utilisabilité en pratique. Le système doit également être capable de s'adapter au système proposé par la société afin d'apporter des informations supplémentaires sur les activités de la personne grâce à ce module de reconnaissance de sons. Ce module agira donc comme un capteur complémentaire au système afin de déterminer au mieux le comportement actuel d'une personne âgée.

Chapitre 3

Classification de sons de la vie courante à base d'i-vecteurs

3.1 Introduction aux i-vecteurs

Les i-vecteurs on été créés pour la reconnaissance du locuteur. Ils ont été proposés en complément des méthodes abordées dans le chapitre précédent. Ces méthodes de classification tels que les GMM, les SVM-GSL et les HMM, ont comme point commun d'être des méthodes de classification de grande dimension. Les i-vecteurs ont d'abord été proposés par N. Dehak [21] pour avoir un intermédiaire entre la représentation par MFCC de faible dimension et les super-vecteurs de grande dimension obtenus à l'aide des GMM. La création de ces vecteurs de taille intermédiaire a découlé des travaux de P. Kenny sur le JFA (*Joint Factor Analysis*) [36]. En effet, le JFA modélise la variabilité du locuteur et de la source séparément, le JFA est calculé selon l'équation 3.1, avec m la composante indépendante du locuteur, V la matrice "*eigenvoice*[1]", y les facteurs dépendants du locuteur, U la matrice *eigenchannel*, x la matrice de dépendance du canal, D la matrice résiduelle (diagonale) et z les facteurs spécifiques résiduels du locuteur, s étant le super-vecteur représentant le locuteur "idéal".

$$s = m + Vy + Ux + Dz \tag{3.1}$$

Les i-vecteurs quant à eux sont créés à partir d'une seule matrice TV (Matrice de variabilité totale – *Total Variability matrix*) de faible dimension. Cette

1. Méthode inspirée des *eigenface*, où il est question de reconnaître un visage. Cette méthode permet de définir un visage selon une somme polynomiale de visages moyens.

matrice TV regroupe la variabilité du locuteur et de la source. Les i-vecteurs font partie des méthodes de reconnaissance de locuteur probabilistes, tout comme le JFA dont il découle, et à l'instar des GMM dont le JFA est lui même issu. Il est à noter qu'il existe d'autres approches pour la reconnaissance du locuteur.

- Les méthodes de classification par ressemblance (mesure de similarité)
  - Les plus proches voisins [32]
  - Quantification vectorielle [54]
- Les réseaux de neurones
  - Réseaux de neurones à décalage temporel [6]
  - Réseaux de neurones à arbres de décision [26]

Les méthodes de classification par ressemblance ont tendance à être peu fiables en conditions réelles. Les i-vecteurs quant à eux proposent de bons résultats aussi en conditions réelles. Ces derniers proposent donc d'extraire le super-vecteur GMM sous une forme plus restreinte. Un i-vecteur est donc représenté selon l'équation 3.2.

$$M = m + Tw \tag{3.2}$$

Où M représente le super-vecteur GMM, m la matrice UBM (Modèle du monde – *Universal Background Model*), T la matrice TV et w l'i-vecteur correspondant.

Les matrices UBM et TV sont extraites selon l'algorithme GEM (Généralisation de l'algorithme EM – *Generalized EM*) (Algorithme 1) [22]. Ce dernier est une généralisation de l'algorithme EM. Il existe donc plusieurs implantations de l'algorithme EM. Nous utiliserons l'algorithme GEM pour la suite conformément à l'état de l'art. Cet algorithme permet d'approcher les paramètres au sens du maximum vraisemblance d'un modèle probabiliste. Il se décompose en deux parties, l'évaluation de l'espérance (E) et la maximisation (M) des paramètres du modèle à déterminer. Durant l'étape E on calcule la vraisemblance à partir d'une matrice θ actuelle. L'étape M estime ensuite le maximum de vraisemblance des paramètres trouvés à l'étape E précédente. Ceci par le biais de l'expression $Q(\theta', \theta)$ qui est le critère de Kullback-Leibler à maximiser.

L'UBM est donc un GMM devant représenter les paramètres d'un modèle indépendant du locuteur. Ce modèle est aussi dit générique et est souvent appelé

Initialisation au hasard de θ;
$c = 0$;
while *L'algorithme n'a pas convergé* **do**
  Choisir $\theta^{(c+1)}$ tel que $Q(\theta^{(c+1)}, \theta^{(c)}) > Q(\theta^{(c)}, \theta^{(c)})$;
  $c = c + 1$;
end

Algorithme 1 : Algorithme GEM

modèle "du monde". Ces paramètres sont déterminés suivant l'algorithme EM [49] sur un grand nombre d'enregistrements de parole de plusieurs locuteurs. La matrice TV est obtenue en extrayant les paramètres statistiques issus des équations de Baum-Welch [37] en utilisant l'UBM ; pour un son donné de L fenêtres $\{y_1, y_2, \cdots, y_L\}$ et l'UBM Ω composé de C composantes définies dans un espace de dimensions F. Les statistiques de Baum-Welch estiment l'i-vecteur pour un son donné suivant les équations 3.3.

$$\begin{aligned} N_c &= \sum_{t=1}^{L} P(c|y_t, \omega) \\ F_c &= \sum_{t=1}^{L} P(c|y_t, \omega) y_t \end{aligned} \tag{3.3}$$

avec, $c = 1, \cdots, C$ correspondant à l'index du modèle gaussien

La figure 3.1 représente les processus d'apprentissage et de décodage, d'extraction des sons pour l'apprentissage et pour l'évaluation, d'un système d'i-vecteurs.

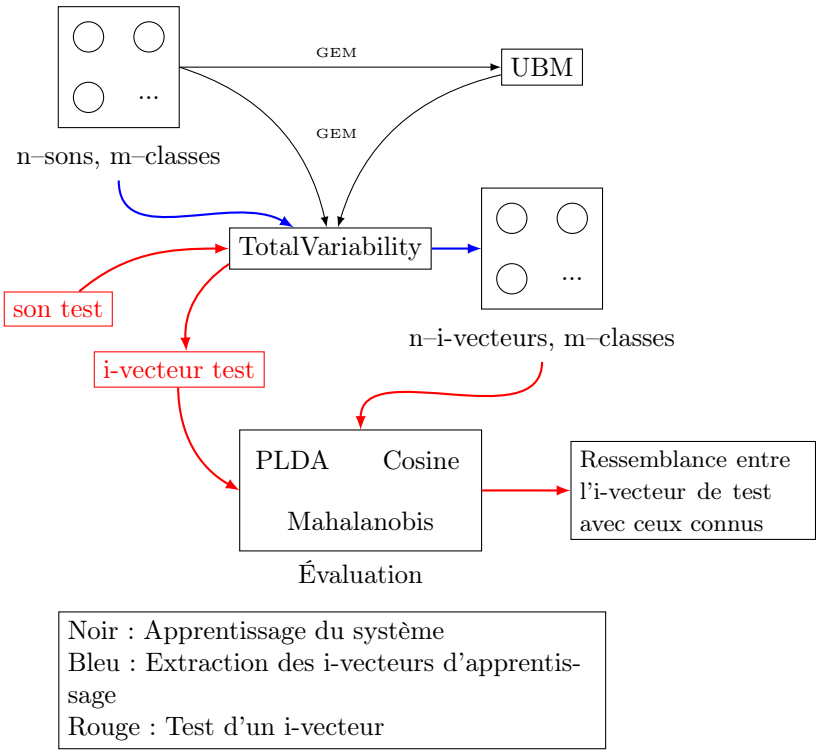


FIGURE 3.1 – Processus d'extraction et d'évaluation des i-vecteurs

3.2 Application des i-vecteurs à la classification de sons

Les i-vecteurs depuis leur création sont très utilisés dans le domaine de la parole et de la reconnaissance du locuteur. Ils ont permis non seulement de dépasser l'état de l'art, mais aussi d'améliorer les performances et les temps de réponse de ces systèmes. Ils permettent de réduire et de résumer un enregistrement audio présentant entre 16000 et 44000 données par seconde – dans la plupart des cas – en un vecteur dont la taille est de l'ordre de la centaine de données. Les i-vecteurs permettent donc une classification de coût de calcul faible, efficace et précise, ce qui en fait un atout majeur pour les implanter dans un système embarqué.

3.2.1 L'utilisation des i-vecteurs pour la reconnaissance de sons

L'idée principale qui a guidé la reconnaissance de sons vers l'utilisation des i-vecteurs est la forte ressemblance avec la reconnaissance du locuteur. En effet à la différence de la reconnaissance de parole qui se base sur un modèle probabiliste de syntaxe et de grammaire, ces deux domaines de recherche ne peuvent pas se baser sur d'autres informations et corrections. Dans notre cas la reconnaissance de la source d'un son revient à reconnaître le son, et par extension, un groupe de sons identifiés. L'information portée par le son est à peu près équivalente à reconnaître la source du son. Un autre avantage théorique abondant en faveur des i-vecteurs est le modèle UBM compris dans le calcul de ces derniers, qui permet une adaptation plus efficace pour les utilisateurs du système. Cette adaptation théorique s'appuie sur le fait que le nombre de sons différents ainsi que la variabilité des sons identifiables dans un logement sont limités et semblables d'un appartement à l'autre.

Dans le domaine de la reconnaissance de sons il est courant d'utiliser des paramètres supplémentaires afin de mieux différentier les sons à reconnaître. Nous avons cependant commencé par évaluer les i-vecteurs pour la reconnaissance de sons dans des conditions équivalentes à celles utilisées pour la reconnaissance de locuteur. Nous avons tout de même adapté les paramètres utilisés en nous inspirant des pratiques de la reconnaissance de sons. Les paramètres utilisés sont 19 coefficients [2] MFCC, ainsi et leurs dérivées première et seconde du même ordre soit 57 coefficients au total.

2. Nombre de coefficients MFCC usuellement utilisés dans l'état de l'art de la reconnaissance de sons.

3.2.2 Le corpus de sons utilisé

Afin de pouvoir comparer nos travaux aux travaux précédents de M. Sehili [53], il a été décidé d'utiliser la même base de sons (Tableau 3.1) que ce dernier durant ses travaux. Nous avons également étoffé cette base en ajoutant du bruit de mesure aux sons. Pour commencer nous utiliserons du bruit blanc, qui bien qu'absent en conditions réelles, présente un intérêt théorique et représente également de très mauvaises conditions pour effectuer un processus de reconnaissance. Nous ajouterons également des bruits d'un appartement dans diverses situations que nous avons identifiées comme les plus pertinentes. Nous aurons donc des bruits d'appartement sans activité particulière, des bruits de ce même appartement avec les fenêtres ouvertes, ce qui représentera un cas un peu plus défavorable car il ajoutera des bruits externes et variant beaucoup d'un appartement à un autre, et pour finir des sons avec une machine à laver le linge en cours de fonctionnement, qui est l'un des bruits modifiant le plus l'environnement sonore d'un appartement. Chacun de ces types de bruits crée en lui même 4 nouvelles bases en fonction du rapport signal sur bruit : 00dB (Décibel), 10dB, 20dB et 40dB.

Classe de sons	**Échantillons**	**Durée totale (s)**
Breathing – Respiration	50	106.44
Cough – Toux	62	181.69
Dishes – Vaisselle	98	303.77
Door Clapping – Claquement de porte	114	62.70
Door Opening – Ouverture de porte	21	138.94
Electrical Shaver – Rasoir électrique	62	420.33
Female Cry – Pleurs féminins	36	268.19
Female Scream – Cris féminins	70	216.83
Glass Breaking – Bris de verre	101	99.52
Hair Dryer – Sèche cheveux	40	224.86
Hand Clapping – Applaudissements	54	218.65
Keys – Clefs	36	166.34
Laugh – Rires	49	272.65
Male Scream – Cris masculins	87	202.11
Paper – Papier	63	330.66
Sneeze – Éternuements	32	51.67
Water – Eau	54	484.72
Yawn – Bâillement	20	95.87

Tableau 3.1 – Distribution des classes de sons utilisée

Ces bruits ont été ajoutés de manière additive, pour créer ces différents sons

bruités. Le calcul du SNR (Rapport signal sur bruit – *Signal to Noise Ratio*) a été effectué en 4 étapes. Nous commençons par calculer l'énergie moyenne du signal (Équation 3.4).

$$(E_{\text{signal utile}})_{dB} = 10 \cdot \log\left(\frac{1}{N}\sum_{i=0}^{N-1} s_i^2\right) \tag{3.4}$$

Puis nous déterminons le niveau de l'énergie moyenne du son bruitant pour obtenir le SNR désiré (Équation 3.5).

$$E_{\text{bruit nécessaire}} = 10^{\frac{(E_{\text{signal utile}})_{dB} - SNR}{10}} \tag{3.5}$$

Ainsi que le calcul de l'énergie moyenne par échantillon du bruit (Équation 3.6).

$$E_{\text{bruit}} = \frac{1}{N}\sum_{i=0}^{N-1} b_i^2 \tag{3.6}$$

Et donc le calcul du coefficient de multiplication de chaque échantillon de bruit, pour obtenir le SNR désiré se fait suivant l'équation 3.7.

$$\text{Coefficient} = \sqrt{\frac{E_{\text{bruit nécessaire}}}{E_{\text{bruit}}}} \tag{3.7}$$

3.3 Méthode d'évaluation et fusion de résultats d'i-vecteurs

Il existe plusieurs méthodes d'évaluation d'i-vecteurs communément utilisées. Ces méthodes comportent une phase de normalisation des i-vecteurs suivie

d'une phase d'évaluation. Cette phase de normalisation permet de lisser les artefacts et les valeurs extrêmes à l'intérieur de notre base d'i-vecteurs et donc d'améliorer significativement les résultats d'un système d'i-vecteurs [21].

Méthodes de normalisation

Les méthodes de normalisation varient en fonction de la méthode d'évaluation utilisée, certaines normalisations étant plus ou moins adaptées à la méthode d'évaluation utilisée. On peut citer parmi les méthodes de normalisation WCCN (*Within-Class Covariance Normalisation*) [31], PLDA (*Probabilistic Linear Discriminant Analysis*) [47], EFR (*Eigenfactor Radial*) [10] et SphNorm (*Spherical Nuisance Normalisation*) [9].

Normalisation WCCN

Le but de la méthode WCCN est de créer l'échelle de l'espace des i-vecteurs inversement proportionnelle à la matrice de covariance interne à une classe ; cette matrice est calculée pour une classe donnée selon l'équation 3.8. Avec A étant la matrice de projection LDA (Analyse discriminante linéaire – *Linear Discriminant Analysis*). $\overline{w_s}$ (Équation 3.9) la moyenne du LDA projetée pour une classe donnée s.

$$W \triangleq \frac{1}{S}\sum_{s=1}^{S}\frac{1}{n_s}\sum_{i=1}^{n_s}(A^t w_i^s - \overline{w_s})(A^t w_i^s - \overline{w_s})^t \tag{3.8}$$

$$\overline{ws} \triangleq \frac{1}{n_s}\sum_{i=1}^{n_s} A^t w_i^s \tag{3.9}$$

Normalisation PLDA

Le but de la normalisation PLDA est de ramener les i-vecteurs dans un espace déterminé par une gaussienne multidimensionnelle et donc de lisser les variations brutales dans l'espace correspondant à une classe donnée. Cette normalisation permet entre autres de limiter l'impact du bruit dans le i-vecteur w' normalisé correspondant à w, selon l'équation 3.10.

$$w' = \frac{\Sigma^{-\frac{1}{2}}(w - \mu_i)}{||\Sigma^{-\frac{1}{2}}(w - \mu_i)||} \tag{3.10}$$

Normalisation EFR

La méthode EFR cherche à conditionner les i-vecteurs et à réduire la variabilité entre les sources en redéfinissant un i-vecteur donné w en un i-vecteur w' en appliquant l'équation 3.11, où V et $\overline{\omega}$ sont respectivement la matrice de covariance et le vecteur moyen estimé pour une classe donnée.

$$w' = \frac{V^{-\frac{1}{2}}(x - \overline{w})}{\sqrt{(w - \overline{\omega})V^{-1}(w - \overline{\omega})}} \tag{3.11}$$

Normalisation SphNorm

Lorsqu'on applique une normalisation de la longueur sur les i-vecteurs ces derniers peuvent être représentés dans un espace sphérique de volume défini, ce qui rend difficile le calcul du WCCN. C'est pourquoi cette nouvelle normalisation a été introduite en utilisant des matrices sphériques de covariance W_i, un i-vecteur w en utilisant la normalisation SphNorm w' selon l'équation 3.12 appliquée i le nombre d'itérations choisies.

$$w' = \frac{W_i^{-\frac{1}{2}}(w - \mu_i)}{||W_i^{-\frac{1}{2}}(w - \mu_i)||} \tag{3.12}$$

Méthodes d'évaluation

Les méthodes d'évaluation utilisées pour les i-vecteurs sont : la distance cosinus proposée par N. Dehak [20] et le PLDA proposée par S.J.D. Prince [47].

Distance cosinus

La distance cosinus cherche à déterminer la distance entre deux vecteurs (w_1, w_2) en mesurant le cosinus de l'angle entre eux. Pour les i-vecteurs cela signifie mesurer la distance entre l'angle d'une classe donnée et l'i-vecteur de test. Cette distance peut être exprimée suivant l'équation 3.13.

$$score(w_1, w_2) = \frac{w_1^t w_2}{\sqrt{w_1^t w_1}\sqrt{w_2^t w_2}} \tag{3.13}$$

Distance de Mahalanobis

Il est également possible d'utiliser la distance de Mahalanobis pour mesurer la distance (ou dissimilarité) entre deux vecteurs donnés. Cette distance est calculée selon l'équation 3.14, avec Σ une matrice de covariance, permettant de pondérer davantage les composantes de faibles variances.

$$D_M = \sqrt{(w_1 - w_2)^T \Sigma^{-1} (w_1 - w_2)} \tag{3.14}$$

Évaluation de score par PLDA

Lors de l'évaluation PLDA il est question de créer chaque gaussienne correspondant aux i-vecteurs représentant les classes, puis nous calculons le rapport de vraisemblance (log-likelihood ratio llr) entre l'i-vecteur à tester et les gaussiennes, selon l'équation 3.15, avec H_1 l'hypothèse où w_1 et w_2 sont deux i-vecteurs de la même classe de sons, et H_0 de deux classes de sons différentes.

$$llr = ln \frac{p(w_1, w|H_1)}{p(w_1, w|H_0) \cdot p(w_2, w|H_0)} \tag{3.15}$$

Parmi les différentes méthodes proposées, nous avons choisi d'utiliser les deux méthodes de normalisation SphNorm et PLDA avec la méthode d'évaluation PLDA. Ces méthodes ont été choisies de manière heuristique d'après leur constance et leur précision globale. En effet la variabilité de notre système étant plus grande que dans la reconnaissance du locuteur, une approche probabiliste permet d'appréhender plus efficacement l'environnement sonore proposé. La normalisation SphNorm permettant de réduire l'impact du bruit ambiant est aussi très efficace dans notre cas.

Ces deux méthodes d'évaluation des i-vecteurs proposent des résultats similaires. Nous avons donc choisi d'utiliser ces deux méthodes afin de fiabiliser les résultats obtenus par fusion de celles-ci. Cependant il est nécessaire pour un i-vecteur donné – par extension : un son – de pouvoir déterminer quelle est la bonne méthode si ces dernières divergent dans leur prédiction. Nous avons donc

proposé deux algorithmes afin de fusionner les deux modes d'évaluation choisis : NFD (*Normalized First or Delta*) présenté dans l'algorithme 2 et ANA (*Absolute Normalization and Add*) présenté dans l'algorithme 3, avec s_x le score, x étant le rang du score et m_y la méthode utilisée. Et nous avons pris comme référence le meilleur cas où nous arrivons à sélectionner systématiquement la bonne méthode d'évaluation. Les résultats de ces méthodes de fusion sont regroupés dans le tableau 3.2. Dans la suite des résultats nous utiliserons la méthode de fusion ANA car ses résultats sont les plus proches du meilleur cas défini précédemment.

if $s_{1_{m_1}} \neq s_{1_{m_2}}$ **then**
  $\Delta_1 = norm(s_{1_{m_1}}) - norm(s_{2_{m_1}})$;
  $\Delta_2 = norm(s_{1_{m_2}}) - norm(s_{2_{m_2}})$;
  if $\Delta_2 > \Delta_1$ **then**
    return $s_{2_{m_2}}$;
  end
end
return $s_{1_{m_1}}$;

Algorithme 2 : Algorithme NFD

foreach *score* s_x **do**
  $s_{x_{m_1}} = \frac{s_{x_{m_1}}}{|max(s_{m_1})|}$;
  $s_{x_{m_2}} = \frac{s_{x_{m_2}}}{|max(s_{m_2})|}$;
  $S_x = s_{x_{m_1}} + s_{x_{m_2}}$;
end
return $max(S_x)$;

Algorithme 3 : Algorithme ANA

Méthode de fusion	**Score**
Meilleur cas	79.45%
NFD	79.22%
ANA	79.41%

Tableau 3.2 – Comparaison des différentes méthodes de fusion

Afin d'évaluer la précision du système global nous avons sélectionné la méthode *ten-folds*, qui consiste à découper aléatoirement la base de données en 10

parts égales puis à retirer une part de l'apprentissage et de s'en servir comme test, puis à réitérer cette manœuvre pour chacune des parts. Afin de valider nos résultats nous avons effectué cette méthode deux fois afin de valider le fait que nous ne nous trouvions pas dans un cas idéal. Le choix de la méthode *ten-folds* permet de minimiser l'interdépendance entre échantillons et ainsi de se rapprocher au maximum de la précision effective du système évalué [42].

3.4 Résultats de la classification de sons par i-vecteurs

Le système proposé pour la reconnaissance de sons par i-vecteurs est représenté par la figure 3.2.

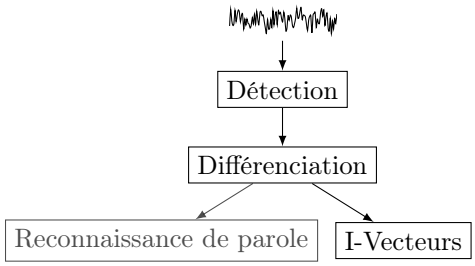


FIGURE 3.2 – Système proposé

La détection est basée sur l'algorithme proposé par D. ISTRATE [34]. Cet algorithme basé sur une transformée en ondelettes, détermine un seuil à partir duquel le système commence à identifier un échantillon potentiellement remarquable. Lorsque ce seuil de détection est atteint un seuil d'arrêt est déterminé. Une fois le signal passant en dessous de ce seuil d'arrêt durant une durée déterminée le son peut être analysé. Nous avons légèrement modifié cet algorithme, ce dernier possédant un deuxième seuil d'arrêt en fonction du temps de détection autorisé, ce second seuil pouvait créer des échantillons incomplets. Cette limitation temporaire a été supprimée et le seuil d'arrêt est désormais fonction du temps et du seuil de détection au départ. Le seuil d'arrêt est donc plus faible au niveau de la détection et augmente au fur et à mesure de la détection, ce qui permet une fin d'échantillonnage plus douce et permet au système de s'adapter plus efficacement à de fortes variations sonores dues à un changement d'activité d'un utilisateur du système. Le module de différentiation a pour but de séparer les parties de parole des autres sons, qui peuvent être données au système de reconnaissance de sons. Il est possible d'utiliser un système d'i-vecteurs à deux

classes (sons et parole), ou une discrimination par les GMM, les deux systèmes ayant tous deux de très bonnes performances.

3.4.1 Évaluation sur des sons non bruités

Nous avons tout d'abord commencé par valider l'utilisation d'i-vecteurs pour la reconnaissance de sons en utilisant les paramètres de la reconnaissance de locuteur adaptés à la reconnaissance de sons. Le signal est découpé en trames de 16ms, auxquelles on applique une fenêtre de Hamming. Pour chaque fenêtre on calcule 19 coefficients MFCC ainsi que leurs dérivées premières et secondes. Les résultats de ce paramétrage est présenté dans le tableau 3.3. Ces résultats sont moins performants que l'état de l'art $\approx -9\%$ par rapport au système SVM-GSL [53]. Néanmoins, la classification par i-vecteurs est beaucoup plus rapide que la classification par SVM-GSL.

Classe de sons	**PLDA Norm**	**Sph Norm**
Breathing	68.00	72.00
Cough	25.81	25.81
Dishes	83.67	79.59
Door Clapping	76.32	82.46
Door Opening	85.71	85.71
Electrical Shaver	98.39	96.77
Female Cry	75.00	72.22
Female Scream	58.57	57.14
Glass Breaking	61.39	69.31
Hair Dryer	97.50	100.00
Hand Clapping	85.19	83.33
Keys	97.22	97.22
Laugh	16.33	8.16
Male Scream	55.17	57.47
Paper	39.68	38.10
Sneeze	28.13	28.13
Water	92.59	90.74
Yawn	25.00	30.00
Total	**66.06**	**66.73**

Tableau 3.3 – Résultats des i-vecteurs pour : 19 MFCC, delta, delta-delta

Nous nous sommes donc proposés d'utiliser des paramètres supplémentaires utilisés également en reconnaissance de sons, tels-que le ZCR, le SC et le SRF,

ainsi que leurs dérivées premières et secondes. Comme nous pouvons le voir dans le tableau 3.4, l'utilisation – en plus des MFCC – des trois paramètres suscités améliore significativement les résultats. Nous pouvons également remarquer que les classes de sons relatives aux sons produits par la voix [3], présentent en moyenne de plus mauvais résultats que les autres classes de sons. Nous pouvons remarquer également si nous regardons la matrice de confusion (Tableau 3.5) que les principales confusions ont lieu dans ce groupe de sons.

Classe de sons	**PLDA Norm**	**Sph Norm**
Breathing	70.00	74.00
Cough	40.32	38.71
Dishes	90.82	84.69
DoorClapping	89.47	89.47
DoorOpening	95.24	95.24
ElectricalShaver	96.77	90.32
FemaleCry	86.11	75.00
FemaleScream	58.57	54.29
GlassBreaking	46.53	57.43
HairDryer	92.50	87.50
HandClapping	92.59	92.59
Keys	88.89	91.67
Laugh	24.49	22.45
MaleScream	51.72	48.28
Paper	66.67	57.14
Sneeze	28.13	21.88
Water	98.15	100.00
Yawn	60.00	65.00
Total	**70.73**	**69.21**

Tableau 3.4 – Résultats en utilisant les paramètres propres à la reconnaissance de sons

3. Il est à noter que la parole est toujours exclue et que l'on ne parle que de sons émis par le conduit vocal, la cavité buccale ou la cavité nasale, comme des cris, pleurs, etc.

Numéro de la classe	*Breathing* (1)	*Cough* (2)	*Dishes* (3)	*Door Clapping* (4)	*Door Opening* (5)	*Electrical Shaver* (6)	*Female Cry* (7)	*Female Scream* (8)	*Glass Breaking* (9)	*Hair Dryer* (10)	*Hand Clapping* (11)	*Keys* (12)	*Laugh* (13)	*Male Scream* (14)	*Paper* (15)	*Sneeze* (16)	*Water* (17)	*Yawn* (18)
(1)	35	3	0	2	0	1	0	0	1	0	0	4	2	0	2	0	0	0
(2)	4	25	4	4	0	0	4	0	5	0	0	0	3	0	0	7	0	6
(3)	0	2	89	0	0	0	0	0	0	0	5	0	0	0	2	0	0	0
(4)	3	0	0	102	0	0	0	0	6	0	0	2	0	0	0	0	1	0
(5)	0	0	0	0	20	0	0	0	0	0	0	0	0	0	1	0	0	0
(6)	0	0	0	0	0	60	0	0	0	1	0	0	1	0	0	0	0	0
(7)	0	0	1	0	0	0	31	0	0	0	0	0	0	3	0	1	0	0
(8)	1	1	0	0	0	0	0	41	1	2	0	0	2	19	1	2	0	0
(9)	8	6	1	6	0	0	0	6	47	0	0	2	2	3	0	14	0	6
(10)	0	0	0	0	0	2	0	0	0	37	0	0	0	0	0	0	1	0
(11)	1	0	2	0	0	0	0	0	0	0	50	0	0	0	1	0	0	0
(12)	0	0	1	1	0	0	0	0	1	0	0	32	1	0	0	0	0	0
(13)	1	5	0	1	0	0	12	1	1	0	0	0	12	6	0	3	0	7
(14)	0	0	0	0	0	1	5	17	3	0	0	0	8	45	0	4	0	4
(15)	5	2	3	0	5	0	0	1	2	1	0	0	0	0	42	0	2	0
(16)	2	7	0	0	0	0	3	0	3	0	0	0	3	1	0	9	0	4
(17)	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	53	0
(18)	0	0	0	1	0	0	0	0	0	0	0	0	6	0	0	1	0	12

Tableau 3.5 – Matrice de confusion (bonnes reconnaissances en gris, principales confusions en rouge)

Nous avons proposé de résoudre ce problème en proposant un système hiérarchique pour les i-vecteurs (Figure 3.3)[51]. Le principe étant de réduire le nombre de confusions entre les sons vocalisés humains (noté *Human* dans la première couche) et les autres sons. Puis dans un second temps de réduire au maximum les confusions entre sons *Humains* eux-mêmes, potentiellement par l'ajout de nouveaux paramètres acoustiques uniquement pour ces derniers. Les résultats de classification, après avoir regroupé les sons humains (Tableau 3.6), montrent une augmentation significative de bonnes reconnaissances. En le combinant avec la seconde couche (Tableau 3.7) on obtient les résultats globaux de classification, et donc un taux de bonnes reconnaissances de 78.45%.

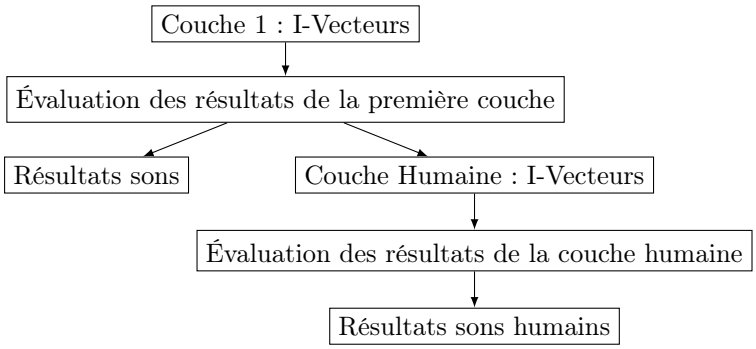


FIGURE 3.3 – Schéma du système de i-vecteurs hiérarchiques

Cluster Name	**PLDA Norm**	**Sph Norm**
Dishes	96.94	90.82
Door Clapping	92.11	92.98
Door Opening	95.24	90.48
Electrical Shaver	93.55	95.16
Glass Breaking	82.18	74.26
Hair Dryer	97.50	97.50
Hand Clapping	90.74	87.04
Human	81.28	82.27
Keys	94.44	94.44
Paper	74.60	68.25
Water	98.15	100.00
Total	**87.04**	**85.70**

Tableau 3.6 – I-Vecteurs hiérarchiques : Première couche

Cluster Name	PLDA Norm	Sph Norm
Breathing	92.11	97.74
Cough	63.64	65.91
Female Cry	88.24	88.24
Female Scream	83.61	81.97
Laugh	61.36	61.36
Male Scream	68.24	70.59
Sneeze	55.56	66.67
Yawn	84.21	68.42
Total	**73.86**	**74.72**

Tableau 3.7 – I-Vecteurs hiérarchiques : Seconde couche

Nous voyons donc que la performance globale du système se dégrade principalement dans la seconde couche. Afin d'améliorer les résultats de la seconde couche de reconnaissance, nous avons introduit le paramètre acoustique RER décrit en section 2.3.1, ce paramètre temporel ayant pour but de discriminer les enveloppes du signal très distinctes, cette modification nous à permis d'augmenter notre taux de bonnes reconnaissance à 79.41%, ce qui représente une hausse de 0.96% de bonnes reconnaissances.

Cluster Name	PLDA Norm	Sph Norm
Breathing	97.37	97.37
Cough	72.73	68.18
Female Cry	88.24	88.24
Female Scream	83.61	90.16
Laugh	63.64	52.27
MaleScream	68.24	69.41
Sneeze	81.48	66.67
Yawn	63.16	63.16
Total	**76.71**	**75.00**

Tableau 3.8 – I-Vecteurs hiérarchiques : Seconde couche avec RER

3.4.2 Évaluation sur des sons bruités - bruit blanc

Afin de valider l'utilisation des i-vecteurs en situation réelle nous devons également les tester avec différents niveaux de bruit. Nous avons commencé par tester avec le bruit blanc, bien que ne présentant qu'un intérêt théorique, il place le système dans des conditions défavorables par rapport au bruit réel. La performance de notre système par rapport aux différents niveaux de bruit est représentée dans la figure 3.4. Pour l'expérimentation nous avons entraîné notre système d'i-vecteurs hiérarchiques sur les sons non bruités, puis nous effectuons les tests par rapport aux sons bruités. Nous pouvons observer que lorsque le SNR est très haut il y a une faible d'incidence du bruit sur les résultats, puis lorsque le SNR diminue la précision du système diminue proportionnellement au SNR.

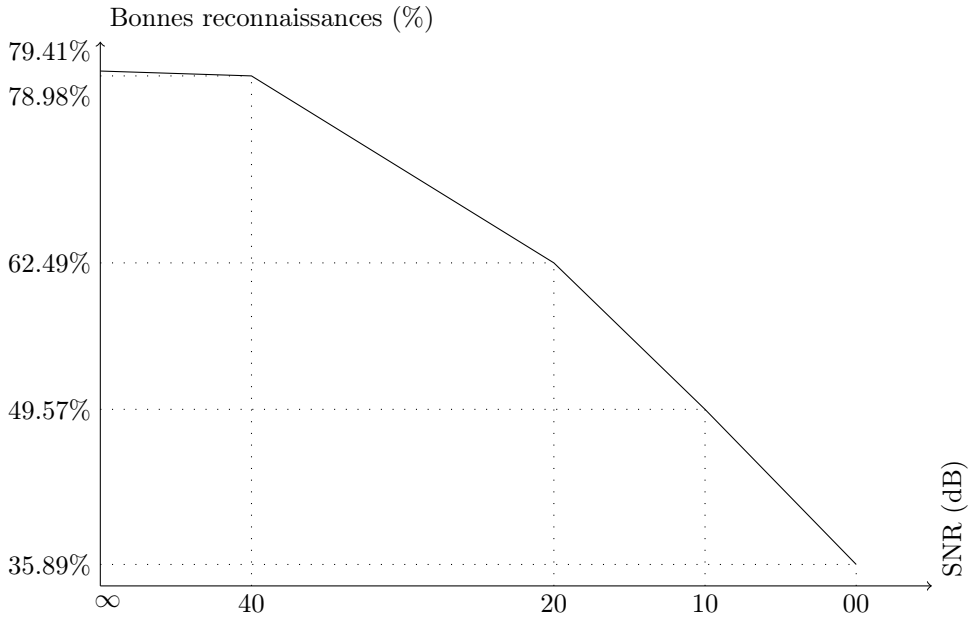


FIGURE 3.4 – Taux de bonnes reconnaissances selon le niveau de bruit

3.4.3 Évaluation sur des sons bruités - bruit réel

Nous avons vu que le taux de bonnes reconnaissance évoluait proportionnellement au SNR en section 3.4.2. Nous remarquons cette même évolution sur les

bruits réels (appartement, fenêtre ouverte, lave-linge) présentés en figure 3.5, notre hypothèse est également vérifiée : le bruit réel est moins impactant sur le taux de bonnes reconnaissances. Les résultats sont regroupés dans le tableau 3.9 et présentent les différents impacts sur les résultats en fonction du type de bruit d'appartement.

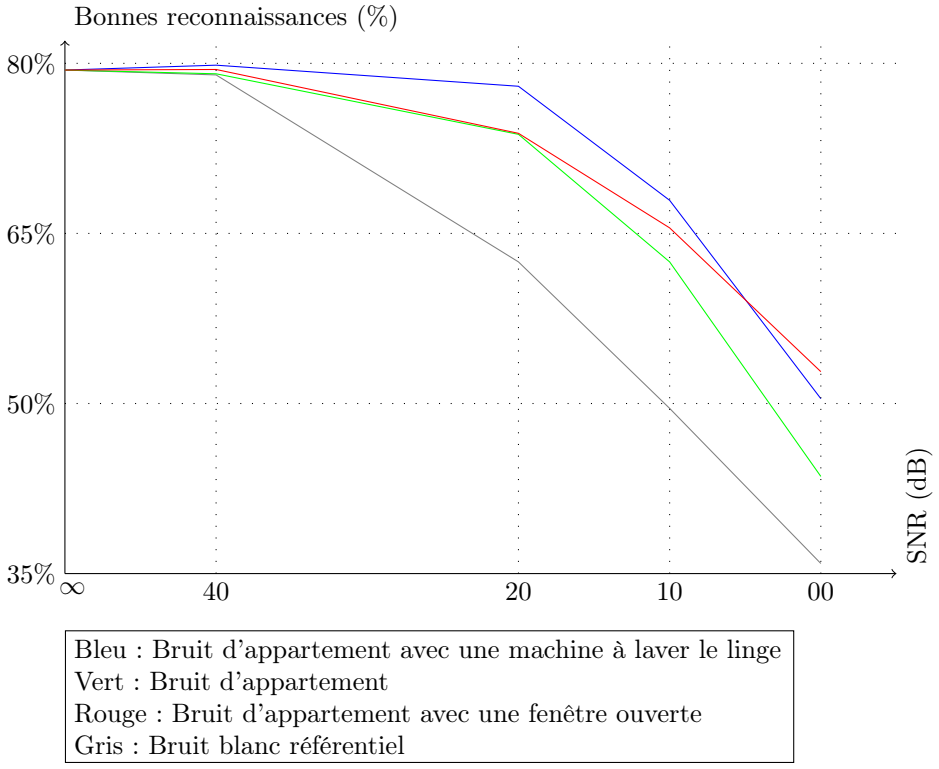


FIGURE 3.5 – Taux de bonnes reconnaissances selon les niveaux de bruit réel

Bruit	**40db**	**20db**	**10db**	**00db**
Appartement	79.08%	73.74%	62.49%	43.14%
Machine à laver	79.84%	77.98%	67.92%	50.43%
Fenêtre ouverte	79.46%	73.83%	64.30%	51.53%

Tableau 3.9 – Résultats selon les différents niveaux de bruit réels

3.5 Conclusion

L'utilisation des i-vecteurs pour la reconnaissance de sons donne de très bons résultats et constitue donc une bonne solution. En effet le système voit son taux de bonnes reconnaissances augmenter de 3.99%, par rapport à la méthode SVM-GSL [53], et avec une méthode d'évaluation plus précise et proche des résultats réels du système. En effet, la méthode d'évaluation utilisée précédemment pour l'évaluation du système SVM-GSL n'évaluait pas l'entièreté de la base de données, et ne possédait pas non plus de multiples évaluations du système. Le second avantage de l'utilisation des i-vecteurs pour la reconnaissance de sons est le temps de réponse par couche qui est de l'ordre de $200ms$ temps réel [4] sur un *Raspberry pi 3* (pour un système SVM-GSL le temps de calcul étant de l'ordre de la seconde sur un ordinateur contemporain), ce qui est un atout majeur pour embarquer une solution de reconnaissance de sons.

4. Voir section 1.5

Chapitre 4

Classification de sons de la vie courante par couplage i-vecteurs/réseaux de neurones

4.1 Introduction à l'utilisation de réseaux de neurones

Les réseaux de neurones ont connu un fort engouement, ces dernières années, de par leur résultats et leur précision meilleure que l'état de l'art dans de nombreux domaines. Ce regain d'attention est dû à l'avènement de l'apprentissage profond, qui lui-même a été rendu possible par l'augmentation de puissance des machines personnelles mais surtout grâce à l'utilisation plus aisée par le grand public des technologies GPGPU (*General-purpose processing on graphics processing units*). Les technologies GPGPU ont pour principe d'exécuter un programme sur une carte graphique ; l'avantage de ce procédé est de pouvoir effectuer un grand nombre de calculs en parallèle à une fréquence moindre. Les réseaux de neurones profonds se prêtent très bien à ces technologies car ils nécessitent de faire beaucoup de fois le même calcul de façon répétitive et sans nécessité d'antériorité.

Les acteurs majeurs de ce regain d'attention pour les réseaux de neurones et l'apprentissage profond sont les GAFA (Google, Apple, Facebook, Amazon). En effet ces entreprises utilisent depuis 2010 de plus en plus de réseaux de neurones pour répondre à leurs problématiques usuelles, que ce soit en reconnaissance de parole, d'images ou de sons. On peut notamment souligner la performance de Google avec AlphaGo qui a montré qu'une IA (Intelligence Artificielle) était

désormais capable de battre un joueur de Go professionnel [16]. Ce tour de force a été également réalisé grâce à un apprentissage par renforcement où la même IA était dupliquée et jouait contre elle même, ce qui lui a permis d'apprendre à partir d'un grand nombre de parties réelles mais également simulées. Désormais de nombreux travaux autour de la communication entre IA sont réalisées, que ce soit au sujet de tâches collaboratives mais aussi pour le cryptage de données en utilisant des réseaux de neurones profonds.

Aparté sur la singularité technologique

Un certain nombre de futurologues et transhumanistes attendent la singularité technologique pour l'année 2030. La singularité technologique (ou singularité) définit l'instant à partir duquel une IA serait à même de s'auto-améliorer et de s'auto-implanter. Cela entraînerait une évolution exponentielle et incontrôlable de la technologie et des intelligences artificielles. À ce jour les réseaux de neurones sont encore bien loin de cette hypothèse, on peut notamment citer les échecs – qui ont fait le régal des médias traditionnels et traditionalistes – de Microsoft et de son intelligence *Tay* qui a pu être dirigée vers un comportement peu souhaitable. Actuellement nous en sommes encore au stade où nous tentons d'utiliser les IA pour aider le quotidien des personnes, dans un but de reconnaissance et de décision bien défini. Les recherches d'autonomisation des IA commencent à prendre forme mais rencontrent encore de nombreux problèmes de mise en place, ou d'interface avec les humains [64].

4.2 Couplage entre apprentissage profond et i-vecteurs

Nous avons précédemment introduit pour la fusion des différentes méthodes d'évaluation la méthode ANA en section 3.3. Cette méthode nous permettait entre autres d'avoir des résultats proches du maximum atteignable si nous choisissions la bonne méthode d'évaluation de façon systématique, mais aussi de récupérer des nouvelles détections qui auraient eu des scores proches sans jamais être premier, tout en étant très bien classés suivant les deux méthodes d'évaluation des scores. Ce constat nous indique qu'une méthode de sélection ou d'évaluation plus élaborée pourrait améliorer grandement les résultats du système d'i-vecteurs. Nous avons donc exploré la piste des réseaux de neurones dans cette optique, ; en effet, les réseaux de neurones sont réputés pour trouver des corrélations dans de grands jeux de données. Dans le cas d'un apprentissage non supervisé ils permettent de montrer ces corrélations, et dans le cas d'apprentissage supervisé – notre cas – d'utiliser ces corrélations pour faire de la classification de données.

L'apprentissage non supervisé permet dans un jeu de données non triées ou non classées de trouver des similitudes entre les données présentes et de les

rassembler par ressemblance apparente. Ce type d'apprentissage est utile pour fouiller des données afin d'en tirer des tendances ou des comportements et peut être utilisé – par exemple – pour faire de la publicité ciblée en créant donc un profil utilisateur et en utilisant les données des utilisateurs similaires et leur taux d'intérêt pour une publicité, afin de les proposer également aux utilisateurs similaires. Il peut aussi être utilisé comme correcteur orthographique, ou plus largement comme correcteur de données corrompues ou erronées.

L'apprentissage supervisé quant à lui permet, à partir d'une base d'apprentissage étiquetée et préparée à cet effet, de reconnaître après apprentissage, un motif et donc d'effectuer une tâche de classification, ou de prédiction dans le cas de séquence de données.

Dans notre cas nous utiliserons une méthode d'apprentissage supervisée, car notre base de données est étiquetée. Le réseau de neurones pourra donc apprendre à partir des i-vecteurs d'apprentissage et ensuite classer les i-vecteurs de tests, conformément à la méthode *ten-folds* explicitée préalablement. Analogiquement à notre démarche préalable, nous commencerons par tester le système sans bruit, puis nous le testerons avec du bruit blanc, et finalement avec du bruit réel et ce avec différents niveaux de SNR.

Le réseau de neurones utilisé pour l'évaluation des i-vecteurs est un réseau de neurones de type *feed-forward*, c'est-à-dire le premier et le plus simple des réseaux de neurones. Sa représentation est présentée en figure 4.1. L'information traverse toutes les couches dans une seule direction, sans re-bouclage ou cycle entre les couches et perceptrons. Le réseau étant constitué d'une couche d'entrée (en jaune) de 50 neurones correspondant à la taille de nos i-vecteurs, une couche intermédiaire (en vert) de 25 neurones [1] et la couche de sortie (en orange) de 18 neurones correspondant au nombre de classes de la base de données. Le schéma 4.2 représente le processus effectué dans une couche du système, du son à sa classification.

1. On utilise communément une couche cachée avec un nombre de neurones deux fois inférieur à la taille de la couche d'entrée.

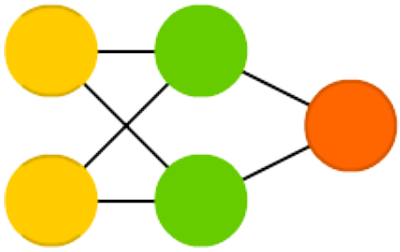


FIGURE 4.1 – Réseau neuronal *feed-forward*

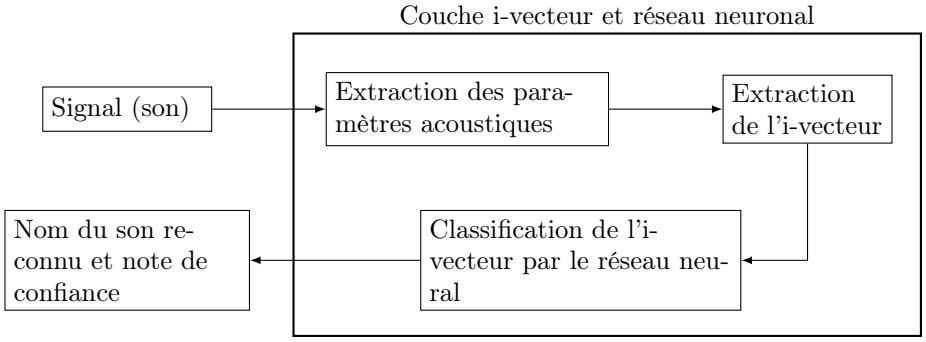


FIGURE 4.2 – Classification d'un son par couplage i-vecteurs et réseau neuronal pour l'évaluation

Nous pouvons observer en figure 4.3 la précision du système. On observe que les résultats sont supérieurs aux méthodes d'évaluation préalablement utilisées et ce avec une amélioration significative de 16.49% de bonnes reconnaissances (soit 95.90%) sans bruit. La résistance au bruit des méthodes d'évaluation précédentes permet de résorber cet écart lorsque du bruit blanc s'ajoute, ceci limite l'écart en présence de bruit blanc à une amélioration moyenne de 5.06%. Les résultats sont regroupés dans le tableau 4.1.

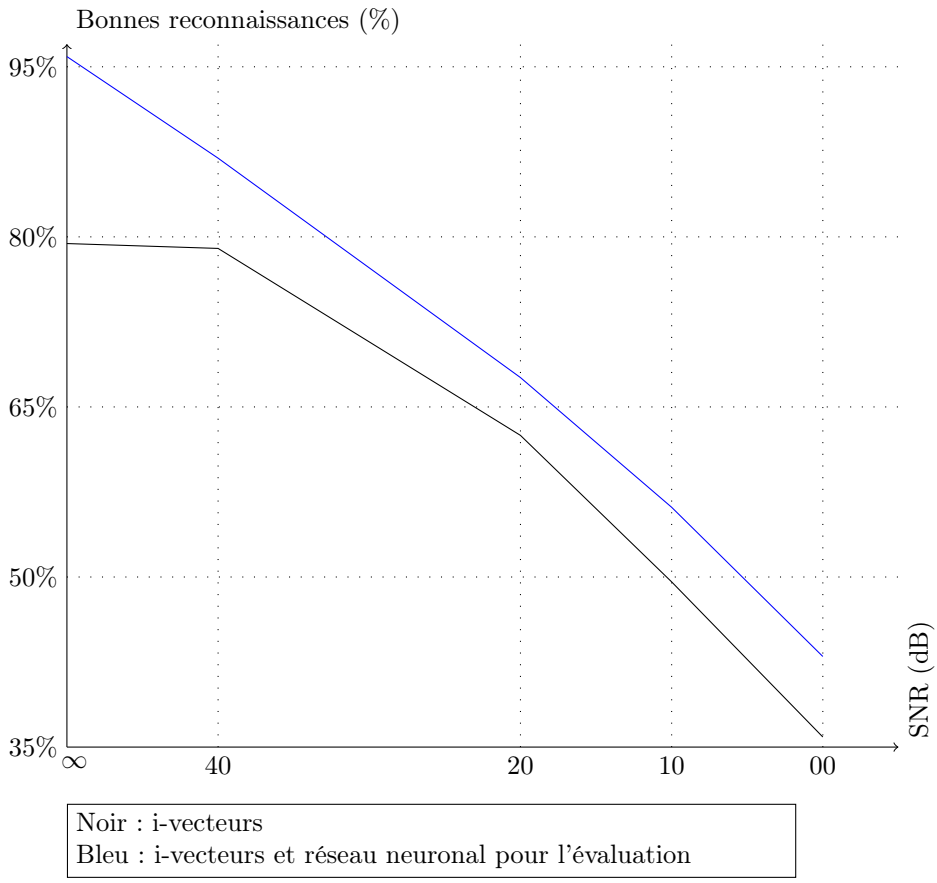


FIGURE 4.3 – Résultats de l'évaluation par réseau de neurones avec du bruit blanc

Bruit	**40db**	**20db**	**10db**	**00db**
Blanc	86.61%	67.59%	56.15%	42.99%

Tableau 4.1 – Résultats du système i-vecteurs et réseau neuronal selon les différents niveaux de bruit blanc

Le système proposé semble donc aussi robuste au bruit avec une évaluation

des i-vecteurs par réseaux de neurones. Nous pouvons donc tester ce dernier sur les bruits réels, les résultats sont présentés en figure 4.4. Conformément à nos hypothèses, le système à une meilleure réponse aux bruits réels qu'au bruit blanc. Nous pouvons également noter que ce nouveau système obtient de meilleurs résultats moyens que les méthodes d'évaluation d'i-vecteurs proposées en section 3.3. Les résultats sont regroupés dans le tableau 4.2.

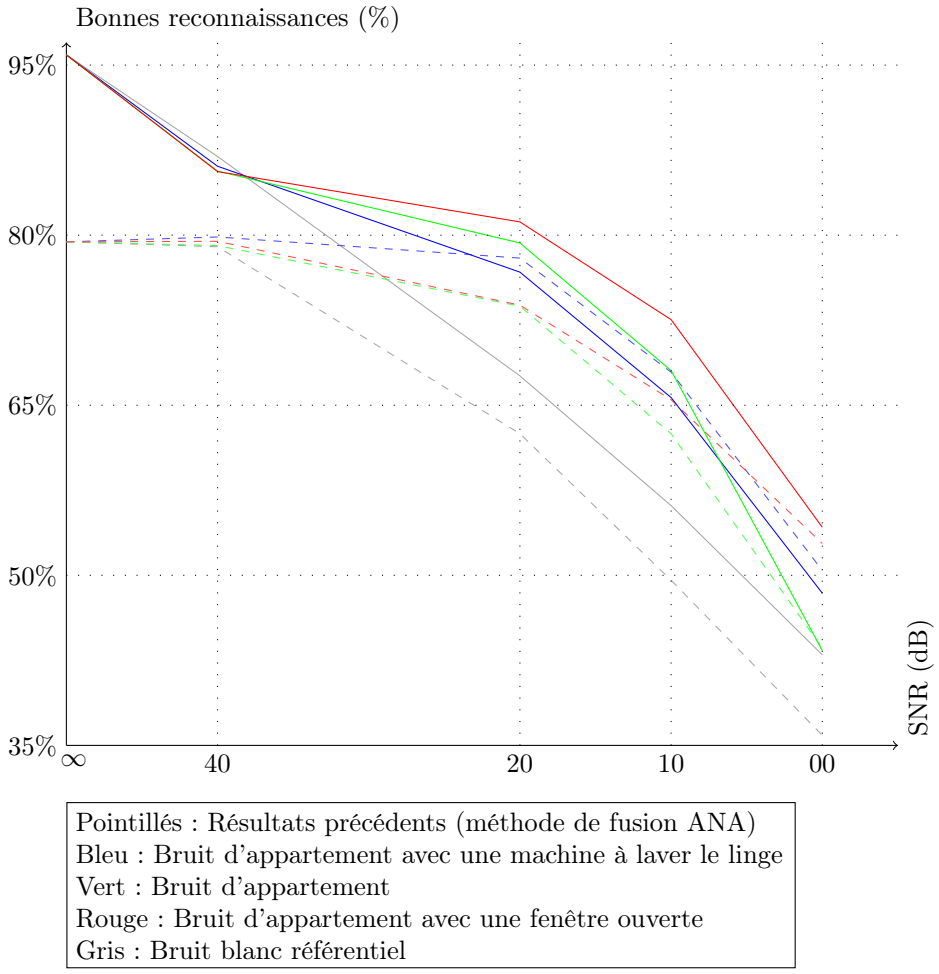


FIGURE 4.4 – Résultats de l'évaluation par réseau de neurones par rapport aux bruits réels.

Bruit	**40db**	**20db**	**10db**	**00db**
Appartement	85.65%	79.31%	68.06%	43.37%
Machine à laver	85.61%	81.17%	72.55%	54.24%
Fenêtre ouverte	86.08%	76.74%	65.68%	48.43%

Tableau 4.2 – Résultats du système i-vecteurs et réseau neuronal selon les différents niveaux de bruit réel

L'évaluation des i-vecteurs par réseau de neurones nous permet donc d'améliorer la précision du système ainsi que sa résistance au bruit. Nous pouvons néanmoins noter que la phase d'apprentissage du système d'évaluation est plus demandeuse en ressources processeur, mais l'évaluation des i-vecteurs reste tout autant efficace que précédemment. L'évaluation des i-vecteurs par réseau de neurones reste donc utilisable dans un système embarqué.

4.3 Utilisation pour la différenciation

La différenciation consiste à séparer trois typologies de sons très différents, la plupart du temps : les sons, la musique et la parole. Du fait des grandes différences, ces systèmes de séparation sont très efficaces souvent moins de 5% d'erreur [58].

Jusqu'à maintenant nous avons évoqué la détection de sons dans l'environnement sonore général (Section 2.2.1), ainsi que la classification de sons en utilisant un système d'i-vecteurs et de réseaux de neurones. Cependant conformément à la figure 3.2 de la section 3.4, il est nécessaire pour notre système d'avoir un module permettant la différenciation des signaux entre parole et sons. Ce module peut être identifié comme un cas particulier de la classification. En effet les classes que nous cherchons à différencier lors de la phase de classification sont ici agglomérés et confrontés à une nouvelle base contenant uniquement de la parole. Pour cette phase nous limiterons les paramètres utilisés à 19 MFCC, le SRF et le SC, ainsi que leur dérivées premières et secondes. Nous ne nous servirons pas du RER et du ZCR car ces paramètres sont de faible influence, d'après nos observations, dans le cas de la différenciation parole - son.

La composition de notre base de données pour la différenciation est présentée dans le tableau 4.3.

Classe de sons	Échantillons	Durée totale (mn)
Sounds – Sons	1049	64.5
Speech – Parole	2000	165.3

Tableau 4.3 – Distribution des classes de sons pour la différenciation

Nous avons ensuite utilisé le même système que celui présenté précédemment pour les i-vecteurs avec la même méthodologie d'évaluation par réseau de neurones, les résultats sont présenté dans le tableau 4.4. Le système proposé présente donc un taux de bonnes reconnaissances de 99.21%, ce qui représente des résultats au niveau ou supérieurs à la précision de l'état de l'art, et nous permettra d'avoir une confiance pratiquement absolue dans le module de différenciation du système.

Classes de sons	Sounds	Speech
Sounds	2072 (98.76%)	26 (01.24%)
Speech	25 (00.62%)	3975 (99.38%)

Tableau 4.4 – Matrice de confusion présentant les résultats du système adapté à la tâche de différenciation

4.4 Conclusion

Nous avons pu montrer que l'utilisation d'un réseau neuronal, pour l'évaluation des i-vecteurs, dans le domaine de la reconnaissance de sons, améliorerait grandement les résultats. Cependant, cette amélioration reste mitigée, en effet, lorsque l'on introduit du bruit dans la base de test, l'écart chute fortement, comme nous pouvons le voir dans la figure 4.5. L'amélioration de 16,49% sur la base de test non bruitée, en entraînement et en validation, est donc à nuancer avec une amélioration moyenne de 5,06%. L'évaluation des i-vecteurs par réseau de neurones est donc plus efficace, dans notre domaine. Nous l'avons appliqué également à la différenciation : sons - paroles, et nous observons également de très bon résultats. Cette méthode nous permet donc de traiter un son après détection en 600ms temps réel [2]. Ce temps de réponse est convenable et cohérent avec la contrainte utilisée en reconnaissance de parole, où la contrainte de temps

2. Voir partie 1.5 pour la signification de temps réel

de réponse maximal communément admise est de 1 seconde.

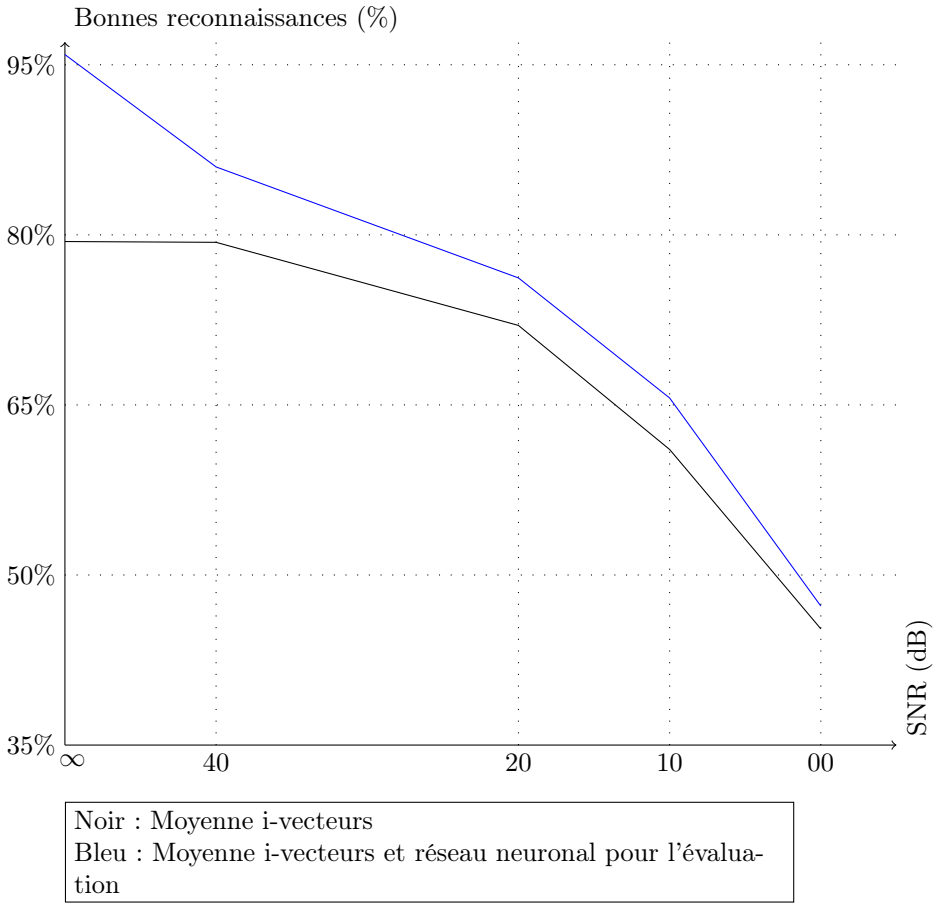


FIGURE 4.5 – Comparaison moyenne entre i-vecteurs et i-vecteurs à évaluation par réseaux de neurones, tous bruits confondus

Chapitre 5

Classification de sons de la vie courante par réseaux de neurones

5.1 Introduction aux différents réseaux de neurones profonds

Nous avons vu dans le chapitre 4 que l'utilisation de réseaux de neurones pour l'évaluation des scores, augmentait significativement la précision de la reconnaissance de sons. Cependant, le système proposé précédemment présente tout de même quelques désavantages : en effet nous utilisons deux couches, et chacune d'elle se déroule en 3 phases : l'extraction des paramètres acoustiques, l'extraction des i-vecteurs à partir des paramètres acoustiques, l'évaluation et la classification de l'i-vecteur ; le système complet peut être schématisé comme sur la figure 5.1. Un seul cycle prend au maximum 200 millisecondes, ce qui nous donne dans notre cas un temps total de 400 millisecondes pour la classification d'un bruit humain. De plus l'utilisation et l'amélioration de ce système implique l'étude des matrices de confusion, dans le but de déterminer les metagroupements à effectuer, afin de créer une couche dans le but d'améliorer la précision du système.

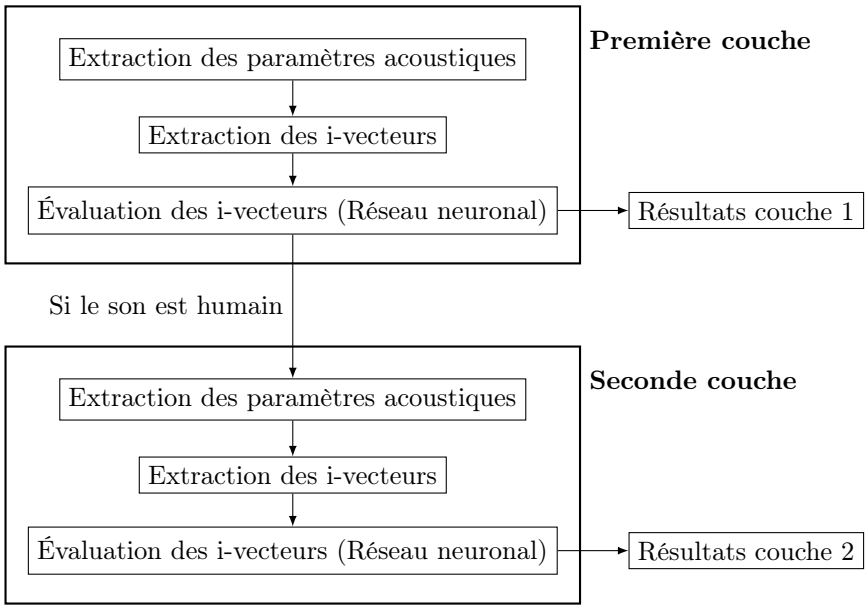


FIGURE 5.1 – Phases composant le système hiérarchique i-vecteurs et réseau de neurones pour l'évaluation

Nous proposons donc afin de pallier les désavantages sus-cités, d'utiliser un seul réseau de neurones pour la classification. Ce système sera donc composé d'une étape d'extraction des paramètres acoustiques puis de la classification du son. Il existe plusieurs catégories de réseau de neurones, les réseaux de neurones *feed-forward* et les réseaux de neurones récurrents dit à mémoire. Les réseaux de neurones récurrents présentent l'avantage de prendre en compte le résultat précédemment donné, et ainsi permettent de mieux gérer les séquences de données – les données présentant une temporalité – par rapport aux réseaux de neurones conventionnels. Ces réseaux de neurones récurrents peuvent posséder divers neurones, les neurones récurrents conventionnels, les LSTM (*Long short-term memory*) [33] [27] et les GRU (*Gated recurrent unit*) [13]. Les neurones (réseaux de neurones) LSTM et GRU sont souvent plus précis que les réseaux récurrents conventionnels cependant ils sont beaucoup plus lents que ces derniers [35]. On peut noter que les réseaux de neurones GRU donnent de résultats équivalents aux LSTM tout en étant plus rapides [15].

Dans un réseau de neurones récurrent ou non on peut utiliser des couches de neurones spéciales : de convolution, d'union (*pooling*), on parle alors de réseau de neurones convolutif (récurrent ou non). L'utilisation de ces couches permet entre autre de pouvoir retrouver certains motifs au sein des données, par exemple

retrouver un visage dans une photo ou une personne, sans avoir à faire le prétraitement pour isoler les parties à reconnaître. Le fait de déléguer une partie des pré-traitements aux réseaux de neurones est une des caractéristiques des réseaux de neurones profonds.

L'utilisation de réseaux de neurones dans les domaines proches de la reconnaissance de sons ont déjà fait leurs preuves, notamment en génération automatique de parole ou de musique [45]. Il semble donc indiqué d'explorer également cette piste pour la reconnaissance de sons.

Les réseaux de neurones, en profondeur

Bien que nous utilisions au quotidien des réseaux de neurones, la compréhension de leur fonctionnement est simple et complexe à la fois. Du moins c'est le sentiment qui se dégage lorsque l'on souhaite s'y intéresser. Nous allons ici tenter de proposer une approche permettant de mieux comprendre ces derniers. Nous allons donc commencer par représenter les deux types de réseaux de neurones expliqués précédemment.

La figure 5.2 représente un réseau neuronal récurrent, les neurones d'entrée sont représentés en jaune et les neurones de sortie en orange. Les neurones bleus représentent les neurones LSTM ou GRU, on observe sur ces derniers une boucle, en effet une des sorties de ces neurones est également une entrée de ce même neurone.

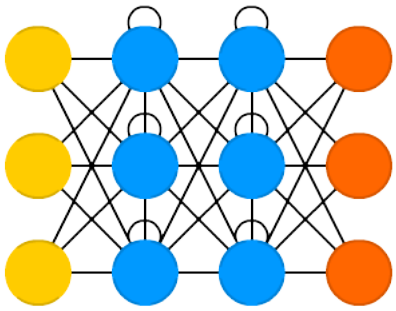


FIGURE 5.2 – Exemple de réseau neuronal récurrent

La figure 5.3 présente un réseau de neurones convolutif, les neurones jaunes et oranges, comme précédemment représentent respectivement les entrées et sorties du réseau neuronal. Les neurones représentés en rose sont des neurones de

convolution ou d'union (*pooling*), et les neurones en vert sont des neurones appartenant à des couches cachées entièrement connectés (*Dense*).

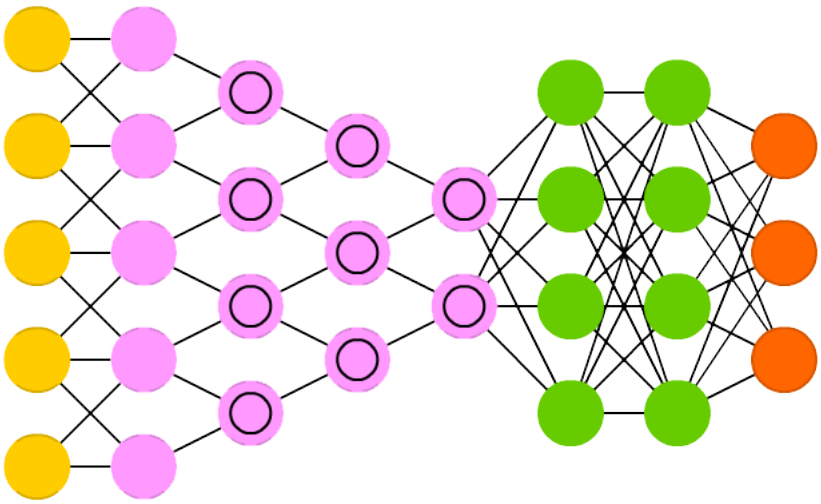


FIGURE 5.3 – Exemple de réseau neuronal convolutif

Comme nous pouvons le remarquer dans les figures 5.2 et 5.3, les neurones sont organisés par niveaux, ce que l'on appelle des couches. En effet il serait fastidieux de créer un réseau neuronal neurone par neurone, c'est pourquoi on utilise des couches possédant un nombre défini de neurones d'un type, puis d'organiser ces couches afin de parvenir aux résultats escomptés. Par exemple un réseau neuronal récurrent convolutif correspondra à la figure 5.3 mais avec des couches récurrentes à la suite des couches convolutives.

On peut donc dire qu'il existe 5 types de couches de neurones : entrée, sortie, récurrentes, convolutives et d'union. Les couches de sortie peuvent utiliser diverses méthodes d'activation afin de déterminer la classe de sortie ou l'appartenance en fonction de l'application souhaitée. Les fonctions d'activation ont pour but de décider si la sortie du neurone sera ou non activée. Les couches cachées sont des couches qui présentent des neurones qui utilisent une méthode d'activation choisie au préalable et l'appliquent à leur entrée. Les couches récurrentes sont composées de neurones semblables au neurones de couches cachées à l'exception que l'une de leur sortie sera également entrée pour la/les données suivantes, c'est pourquoi l'on parle parfois de réseaux de neurones à mémoire.

Les couches convolutives utilisent des neurones convolutifs, ces derniers ont pour fonction d'appliquer un traitement sur une portion de la donnée en entrée 5.4. Une couche convolutive effectue donc le même traitement sur toutes les données de manière légèrement décalée pour chaque neurone.

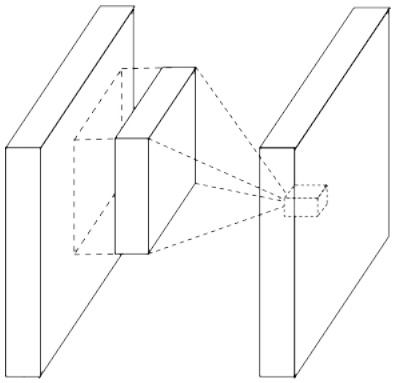


FIGURE 5.4 – Exemple de traitement d'un neurone convolutif, pour des données en 2 dimensions

Les couches d'union quant à elles permettent de réduire la taille des données par mise en commun à partir d'une fonction précise. Un exemple de mise en commun par fonction max est présenté dans la figure 5.5, la taille de mise en commun, de 2 par 2 ici, est réglable en fonction du résultat escompté.

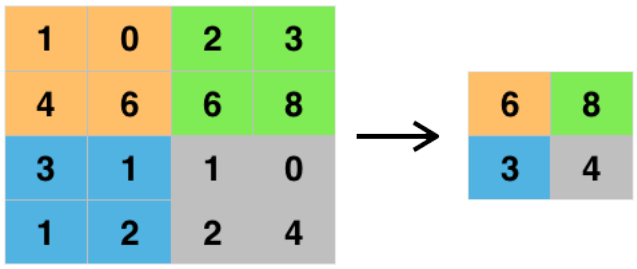


FIGURE 5.5 – Exemple de traitement d'une couche d'union par maximum (*Max-Pooling*)

5.2 Réseau de neurones pour la classification des sons de la vie courante

Les réseaux de neurones récurrents sont souvent utilisés pour travailler avec des données temporelles. Cependant ces derniers sont principalement utilisés dans des processus de création [41] [60] [63], bien qu'ils soient également parfois utilisés en classification [29]. Nous utiliserons ici une approche, n'utilisant pas de couches de neurones récurrentes. Cette approche pourrait être assimilée à de la reconnaissance d'images et donc de motifs sonores au sein des sons, le temps devenant un paramètre, une dimension dans les données, pour le système et non plus une succession de donnés. Cette approche pourrait nous permettre donc de reconnaître les sons par patrons. Nous proposons donc d'utiliser un réseau de neurones convolutif. Son architecture est présentée dans la figure 5.6.

Les réseaux de neurones peuvent être utilisés avec des données de taille variable, néanmoins il est plus commun d'utiliser une taille de données d'entrée fixe, cela permet bien souvent une meilleure précision du réseau neuronal. Pour ce faire, pour un son donné en apprentissage ou en test, nous allons sélectionner aléatoirement une seconde. Dans le cas où le son dure moins d'une seconde nous comblerons artificiellement la fin avec la valeur nulle. Les sons de notre base de données étant en moyenne d'une seconde, avec certains sons de moins d'une seconde et d'autre ayant une durée pouvant aller jusqu'à 30 secondes. Afin de valider l'utilisation de valeurs nulles pour allonger artificiellement les sons, nous avons également comblé les sons à partir de fichiers contenant 1 heure d'enregistrement sans événement, en les concaténant avant et/ou après le son durant moins d'une seconde, cette manœuvre n'a qu'une très faible influence sur la précision du réseau de neurones.

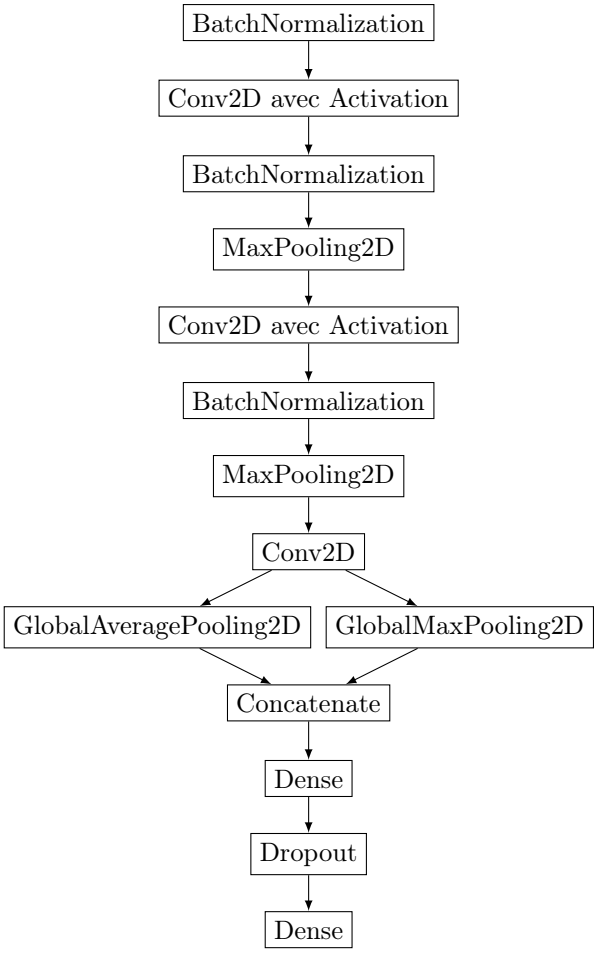


FIGURE 5.6 – Réseau de neurones pour la reconnaissance de sons

Les différentes couches du réseau de neurones

La figure 5.6 fait apparaître différents types de couches que nous allons décrire tant au niveau de leur fonctionnement ainsi que de leur influence sur la classification.

Les couches *BatchNormalization* normalisent les activations de la couche précédente pour un groupement de données (*batch*), en optimisant l'activation moyenne proche de 0, et l'activation de l'écart type proche de 1.

Les couches *Conv2D* sont les couches convolutives selon deux dimensions, elles permettent de lisser chaque point par rapport à ses proches voisins, cela permet de lisser les artefacts et donc de faciliter la détection des motifs.

Les couches *MaxPooling2D* réduisent la taille de la donnée en cours, en transformant chaque groupement en une seule donnée (ici, la plus grande). Cette étape permet la détection des motifs.

La couche *Dropout* est une couche utile durant l'apprentissage du système et permet d'éviter le sur-apprentissage (*overfitting*). Le sur-apprentissage est le fait qu'un réseau de neurones apprenne trop fortement les données qu'il retrouve pendant son apprentissage, ce qui le rend moins efficace lorsqu'il trouve une donnée qu'il n'a jamais vue. Ici cette couche va aléatoirement mettre une partie des informations reçues à la valeur 0. Dans notre cas la moitié des entrées sera mise à la valeur 0, une valeur plus faible ne permettait pas d'éviter le sur-apprentissage, et une valeur plus forte aboutissant à une moins bonne précision.

Fonctions d'activation et d'optimisation choisies

Pour les activations des couches *Conv2D* nous avons utilisé la formule ELU (*Exponential Linear Unit*) [17] définie selon l'équation 5.1, avec x la valeur d'entrée de la fonction d'activation, et α l'hyper-paramètre ELU. Ce paramètre contrôle la valeur à laquelle l'ELU sature pour des entrées négatives. Cette méthode présente une vitesse d'apprentissage supérieure mais également une meilleure précision de classification [17].

$$
\begin{aligned}
f(x) &= \begin{cases} x, & \text{si } x > 0 \\ \alpha(exp(x) - 1), & \text{si } x \leq 0 \end{cases} \\
f'(x) &= \begin{cases} 1, & \text{si } x > 0 \\ f(x) + \alpha, & \text{si } x \leq 0 \end{cases}
\end{aligned}
\tag{5.1}
$$

Pour la couche *Dense* suivant la concaténation, nous utilisons la méthode ReLU (*Rectified Linear Units*) [40] définie selon l'équation 5.2. Cette méthode a pour effet de modifier toutes ses valeurs négatives en 0.

$$f(x) = max(0, x) \tag{5.2}$$

Pour la dernière couche nous avons utilisé la fonction softmax définie selon l'équation 5.3. Cette méthode réduit l'ensemble de ses sorties, en probabilités d'appartenance à une classe ; pour chacune des valeurs calculées, après application de la formule nous obtenons un vecteur ($\sigma(z)$) de taille K – correspondant au nombre de classes à reconnaître – dont la somme est égale à 1.

$$\sigma(z)_j = \frac{exp(z_j)}{\sum_{k=1}^{K} exp(z_k)} \qquad \forall j \in \{1, \cdots, K\} \tag{5.3}$$

Nous avons choisi comme optimiseur *rmsprop* [56], défini selon l'équation 5.4, afin d'effectuer la rétropropagation du gradient (*Backpropagation*), méthode qui calcule l'erreur commise par chaque neurone, dans le but d'appliquer à chaque neurone une modification de son poids (synaptique) plus ou moins importante en fonction de son implication dans l'erreur commise. Avec $f'(\theta_t)$ la dérivée de l'erreur à l'itération t, α le facteur d'apprentissage et γ le facteur d'oubli. Cette méthode utilise l'amplitude des gradients récents pour normaliser les gradients. Elle garde une moyenne mobile sur les gradients carrés (RMS), par laquelle le gradient courant est divisé.

$$\begin{aligned} r_t &= (1-\gamma) f'(\theta_t)^2 + \gamma r_{t-1} \\ v_{t+1} &= \frac{\alpha}{\sqrt{r_t}} f'(\theta_t) \\ \theta_{t+1} &= \theta_t - v_{t+1} \end{aligned} \tag{5.4}$$

5.2.1 Évaluation sur des sons non bruités

Dans un premier temps nous nous sommes proposés de tester le réseau de neurones précédemment exposé avec des sons non bruités. Cette démarche nous

permet de valider la pertinence de l'utilisation d'un réseau neuronal ainsi que la précision que celui peut avoir. Pour ce faire à partir d'un fichier audio nous en extrayons 1 seconde. Sur cette seconde nous extrayons les MFCC sur 63 fenêtres de 256 points chacune, puis nous utiliserons 19 coefficients MFCC, ainsi que leurs dérivées premières et secondes. Ceci constituera l'entrée de notre réseau de neurones. La base de donnée est découpée en deux parties : 60% pour l'entraînement et 40% pour la validation. Cette découpe est due à la faible taille de notre base de données. Usuellement en apprentissage profond la base d'entraînement est plus restreinte que la base de validation. Nous avons effectué cette découpe et leur validation plusieurs fois de façon aléatoire, pour valider le fait que nous résultats ne sont pas le fruit d'un optimal local.

Cette méthodologie nous a permis d'obtenir une précision de 94.27%, ce qui est 1.63% moins efficace que la méthode proposée en section 4. Malgré ce résultat moins efficace, on peut noter une nette simplification du système et de sa mise en place. De plus le temps de calcul pour ce système à base de réseau neuronal est de 100ms pour un son seul, et de 10ms par son pour une grande quantité de sons (plus de 150 000 sons à traiter en même temps). Ce qui résulte en un temps de traitement entre 4 fois et 40 fois, plus rapide que le système proposé précédent. L'amélioration des performances notable pour un très grand nombre de sons s'explique par la parallélisation du chargement des fichiers et de leur groupement pour le transfert dans la mémoire du GPU en vue de leur traitement par le réseau neuronal.

5.2.2 Évaluation sur des sons bruités - bruit blanc

Nous avons ajouté du bruit blanc – comme précédemment – aux sons afin de valider l'utilisation de réseau de neurones profonds pour la reconnaissance de sons. La méthode de reconnaissance utilisée étant très proche de la reconnaissance d'images, nous nous attendons donc à avoir de bons résultats dans les niveaux de bruits faibles, de par les couches convolutives qui devraient estomper le bruit. Cependant dans les forts niveaux de bruit ces même couches pourraient amplifier ce bruit et donc dégrader la précision du système. La figure 5.7 présente l'évolution de la précision du réseau neuronal en fonction des différents niveaux de bruits. Les résultats sont conformes à nos prédictions, la meilleure précision dans les faibles niveaux de bruits est encourageante car les forts niveaux de bruits présentent surtout un intérêt théorique, mais en pratique ces derniers seront probablement rejetés avant leur analyse. Les résultats pour la reconnaissance de sons dans le bruit blanc sont consignés dans le tableau 5.1.

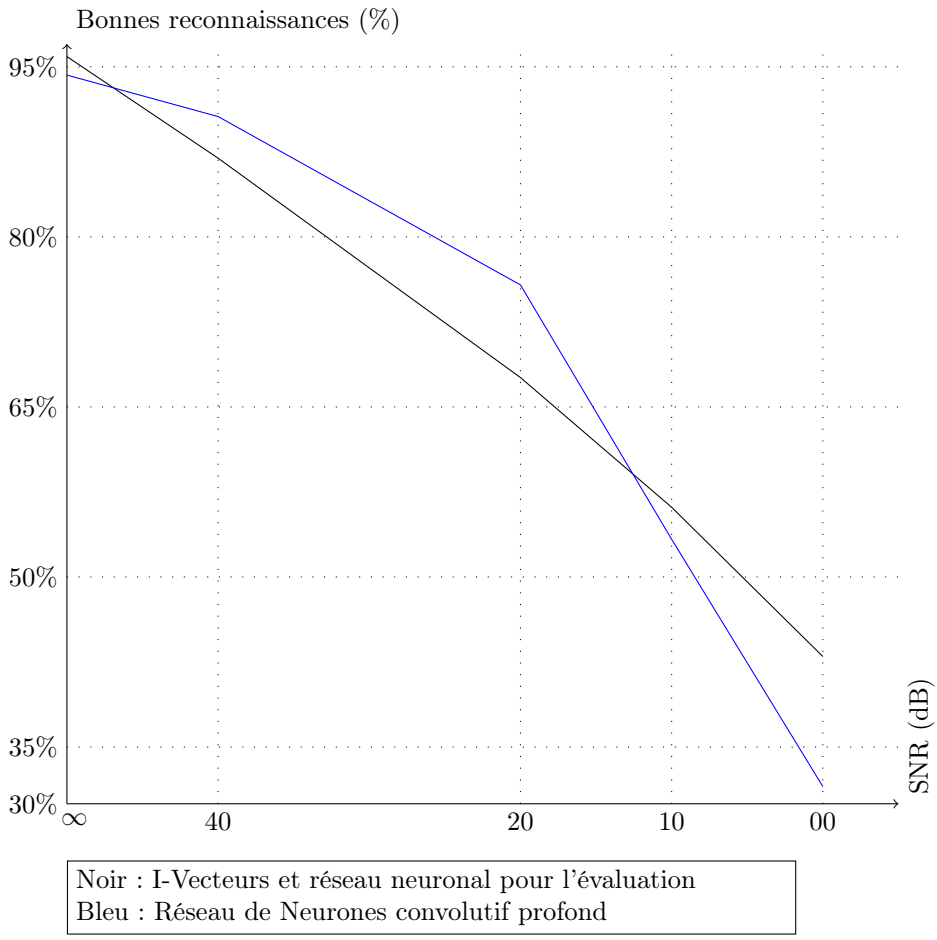


FIGURE 5.7 – Précision du réseau neuronal profond en fonction des différents niveaux de bruit blanc

Bruit	**40db**	**20db**	**10db**	**00db**
Blanc	90.61%	75.76%	53.33%	31.52%

Tableau 5.1 – Résultats du réseau neuronal profond selon les différents niveaux de bruit blanc

5.2.3 Évaluation sur des sons bruités - bruit réel

Dans les chapitres précédents nous avons pu observer de meilleurs résultats avec le bruit réel par rapport au bruit blanc, nous espérons la même évolution pour notre réseau de neurones profond. Les résultats sont consignés dans le tableau 5.2, nous observons de très bons résultats pour les niveaux de bruit 40db et 20db ce qui sont les niveaux que nous considérons les plus pertinents en pratique. La différence d'évolution entre ce système et le système présenté en 4 est présentée dans la figure 5.8. Nous observons donc une sorte de plateau pour les niveaux de bruits 40db et 20db et des résultats légèrement supérieurs pour 10db par rapports au systèmes précédents.

Bruit	**40db**	**20db**	**10db**	**00db**
Appartement	93.33%	91.21%	69.09%	38.80%
Machine à laver	92.73%	86.67%	72.42%	51.82%
Fenêtre ouverte	92.97%	84.11%	75.00%	46.88%

Tableau 5.2 – Résultats du réseau neuronal profond selon les différents niveaux de bruit réels

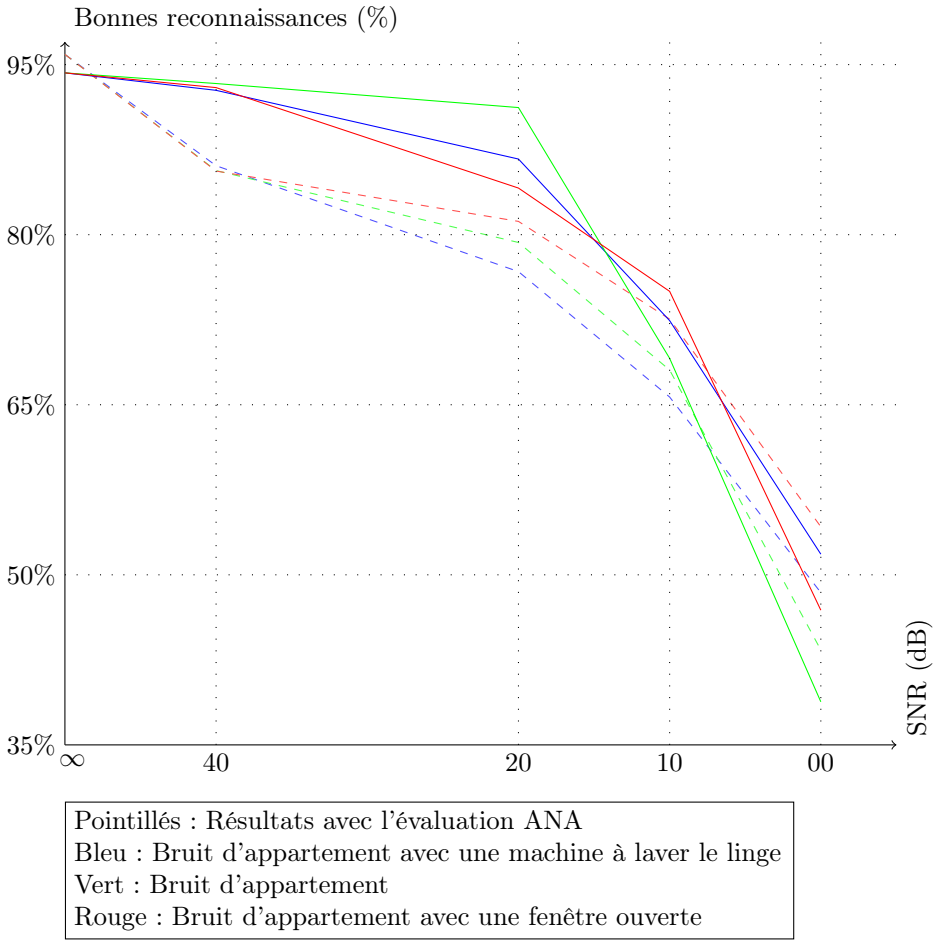


FIGURE 5.8 – Précision du réseau neuronal profond en fonction des différents niveaux de bruit réels comparés au système i-vecteurs hiérarchiques et réseau de neurones pour l'évaluation

5.3 Conclusion

L'utilisation d'un réseau neuronal profond nous a permis d'améliorer significativement les résultats du système précédent, surtout dans les niveaux de bruits faibles. Nous pouvons noter une amélioration moyenne de 7.17% par rapport au système de classification par i-vecteurs hiérarchiques avec évaluation par réseau de neurones ; mais aussi une amélioration moyenne de 12.23% par rapport au système i-vecteurs hiérarchiques. De plus comme nous pouvons le voir sur la figure 5.9 cette amélioration est très significative dans les niveaux de bruits faibles, ce qui nous permettra de créer un indice de confiance inversement proportionnel au niveau de bruit, plus le bruit étant fort plus le niveau de confiance sera faible.

Nous avons pu également remarquer que l'utilisation de réseau de neurones profonds permet d'améliorer significativement le temps nécessaire à la classification d'un son. Nonobstant l'utilisation de réseau de neurones profond reste encore difficile sur un système embarqué. C'est pourquoi il sera préférable de travailler avec un serveur, un micro-service, qui aura pour tâche de traiter les sons. Comme nous l'avons démontré, ce réseau de neurones est capable de traiter un son en 10ms sur une carte graphique *nVidia GeForce 860M* et donc nous ne pouvons qu'espérer d'encore meilleures performance sur un GPU (Processeur graphique – *Graphics Processing Unit*) spécialisé pour les réseaux de neurones, ou du moins un GPU n'étant pas fait pour les ordinateurs portables. Ce traitement déporté devra donc suivre les recommandations de la section 1.6 : les données devront transiter sous forme paramétrique, et le serveur ne devra pas garder de trace du demandeur et se limiter à sa tâche de reconnaissance de sons.

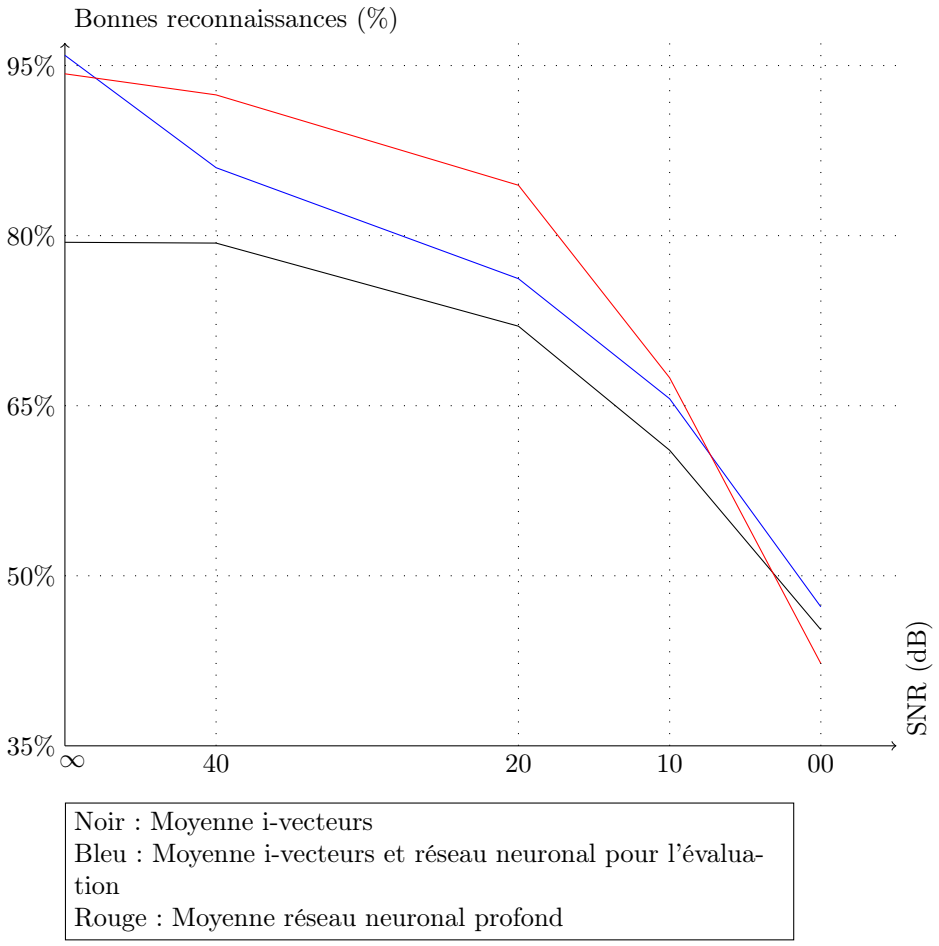


FIGURE 5.9 – Comparaison moyenne entre i-vecteurs et i-vecteurs à évaluation par réseaux de neurones, tous bruits confondus

Chapitre 6

Évaluation du système proposé en conditions réelles

6.1 Introduction et pistes de mise en œuvre

Suite aux résultats obtenus nous nous sommes proposés de tester le système en conditions réelles. En effet, bien que nous ayons testé le système avec du bruit réel et blanc, ces tests ont été effectués en laboratoire et ne peuvent par conséquent refléter l'utilisabilité du système. Les bruits que nous avons introduits pour nos tests ne sont pas exhaustifs, mais représentaient d'après nous des bruits suffisamment perturbateurs n'empêchant pas le système de fonctionner. En conditions réelles nous avons de grande chance de nous retrouver avec d'autres types de bruit : télévision, radio, voisins, etc., ces bruits créeront par conséquent une réponse différente du système que nous nous efforceront de qualifier et quantifier. De plus comme évoqué en section 2.2 les sons que nous cherchons à reconnaître sont loin de couvrir l'ensemble des sons que l'on peut déceler dans un appartement.

Afin de remédier aux limitations apparentes énoncées il est nécessaire d'apporter quelques modifications au système pour sa mise en application en conditions réelles. En effet il sera nécessaire d'ajouter un mécanisme de rejet ou du moins de pondérer les résultats en fonction d'une note de confiance du système. Nous avons identifié pour ce faire trois différents paramètres : le SNR, la note de probabilité d'appartenance à une classe déterminée du réseau neuronal, et le temps (plus spécifiquement la durée et fréquence de détection). Le SNR nous permettra de déterminer si la détection est suffisamment significative, et pourra nous permettre de discriminer les bruits provenant d'une télévision ou d'une radio, par exemple, afin de limiter leur impact sur la décision du système. Le score de sortie du réseau de neurones pourra nous donner une indication de ressemblance, notamment pour un son n'étant pas prévu dans l'apprentissage du système et ainsi élargir la reconnaissance du système ou bien rejeter le son

si le résultat et le SNR sont trop faibles. En effet nous n'avons pas implanté de mécanisme de rejet dans le système actuel. Le système est donc incomplet mais est créé comme exhaustif. C'est pourquoi ce mécanisme de rejet est à implanter pour préserver la stabilité de ses décisions. L'utilisation de la durée de détection peut être également un moyen de filtrer les sons. Cependant cette méthode semble quelque peut naïve et abrupte. La fréquence de détection quand à elle permettra probablement de mieux identifier une activité de la personne, même si les sons détectés ne font pas partie des sons que nous sommes en mesure de détecter. Mais leur succession par ressemblance avec les sons détectables nous permettra possiblement de reconnaître une activité.

Pour le test nous ne pourrons pas suivre l'intégralité des prérogatives éthiques proposées en section 1.6. Regrettablement pour valider les reconnaissances du système il sera nécessaire de permettre à un tiers de pouvoir écouter et labelliser un son détecté par le système. La personne ayant participé au test était volontaire et le système installé lui a été expliqué au préalable, cependant nous n'avons pas explicité à la personne les sons que nous cherchions à détecter, afin de ne pas biaiser l'expérience, cette dernière partie a été expliquée a posteriori.

6.2 Mise en place du système

Afin de tester le système proposé en conditions réelles et tout en respectant la vie privée de notre testeur volontaire, nous avons pris quelques mesures pour la mise en place du système. Afin de simplifier le traitement des données et leur stockage et du fait de la mémoire limitée de la carte mémoire utilisée pour notre *Raspberry Pi3*, nous avons opté pour un système déporté, c'est-à-dire le système installé chez la personne se chargera d'écouter et d'envoyer les sons détectés sur un serveur, nous utiliserons un microphone omnidirectionnel. Seulement trois types de données transiteront : le son, la date et l'heure de cette détection. L'échange de données se fera par HTTPS (*HyperText Transfer Protocol Secure*) afin de transférer le son sous forme cryptée. Une fois un son réceptionné nous pourrons effectuer le processus de classification et le calcul du score de confiance de cette classification. Le système étant encore en test nous n'enverrons pas les résultats de cette classification au système installé. Les sons réceptionnés seront également empaquetés, tous les quarts d'heure, dans une archive encryptée et protégée par mot de passe, le mot de passe étant différent pour chaque archive. Ainsi les sons ne seront pas accessibles aisément, même en cas d'intrusion sur le serveur recevant les détections acoustiques du système. Le stockage des sons réceptionnés nous permettra, a posteriori, de valider les reconnaissances du système. L'ensemble de ces différents processus est rassemblés dans la figure 6.1.

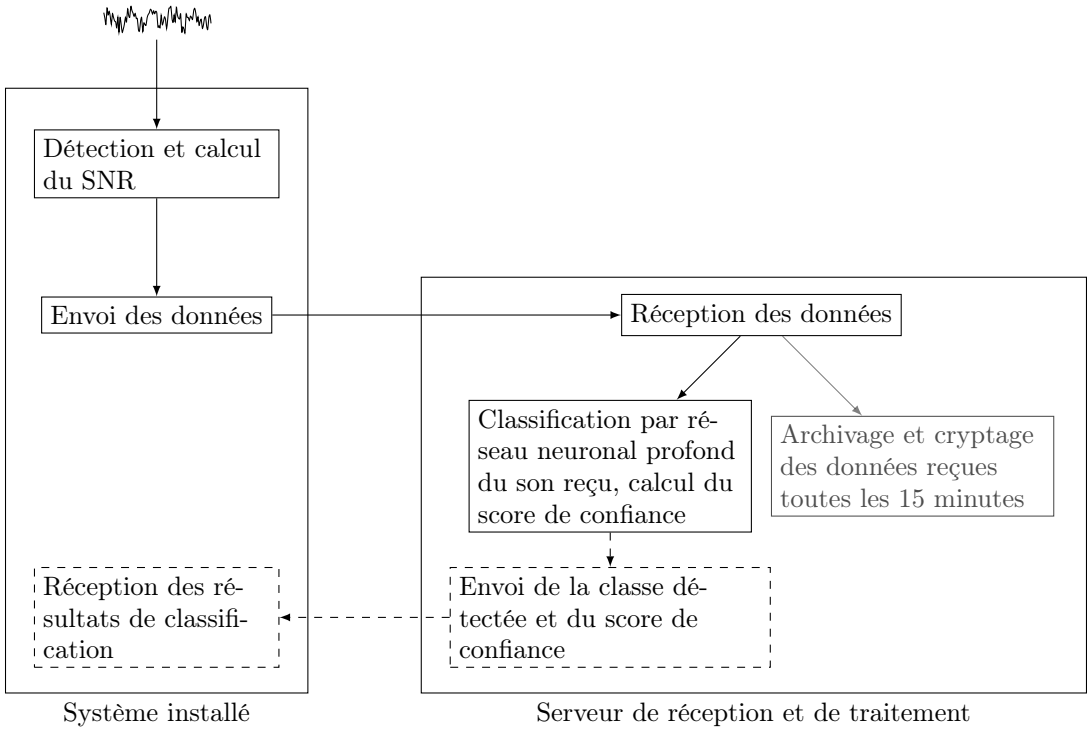


FIGURE 6.1 – Système de test proposé

6.3 Résultats du système

Nous avons installé le système proposé en section 6.2 pendant 24h chez un testeur volontaire. L'endroit d'installation étant une place centrale dans l'appartement à environ un mètre de la cuisine, du salon et de la salle à manger. Le test ayant commencé en début d'après-midi les graphiques présentés dans cette section auront donc pour origine cette date de départ. Durant les 24h de test nous avons relevé 566 événements acoustiques, d'une durée moyenne de 1.33 secondes et avec un SNR moyen de 10.15dB. Le calcul du SNR, a été effectué

à partir des fenêtres précédant la détection comme référence pour l'énergie du bruit.

Le différents événements acoustiques relevés ne sont pas équirépartis sur les 24h, le graphique 6.2 présente la répartition des détections au cours des 24h en fonction du niveau de bruit du relevé. Les fortes concentrations relevées au début et la fin de la période correspondant très probablement aux heures d'installation et de récupération du système par l'installateur. Les pics d'activités que l'on peut remarquer sont corrélés avec les activités de la personne, ces activités ayant été relevées par le système ambiant déjà présent chez la personne. Cependant par soucis éthique nous ne les présenteront pas ici, ces derniers apportant trop d'informations sur les habitudes de vie du testeur. On remarque également que lors des pics d'activités on retrouve également le plus fort taux de sons ayant un SNR inférieur à 10dB.

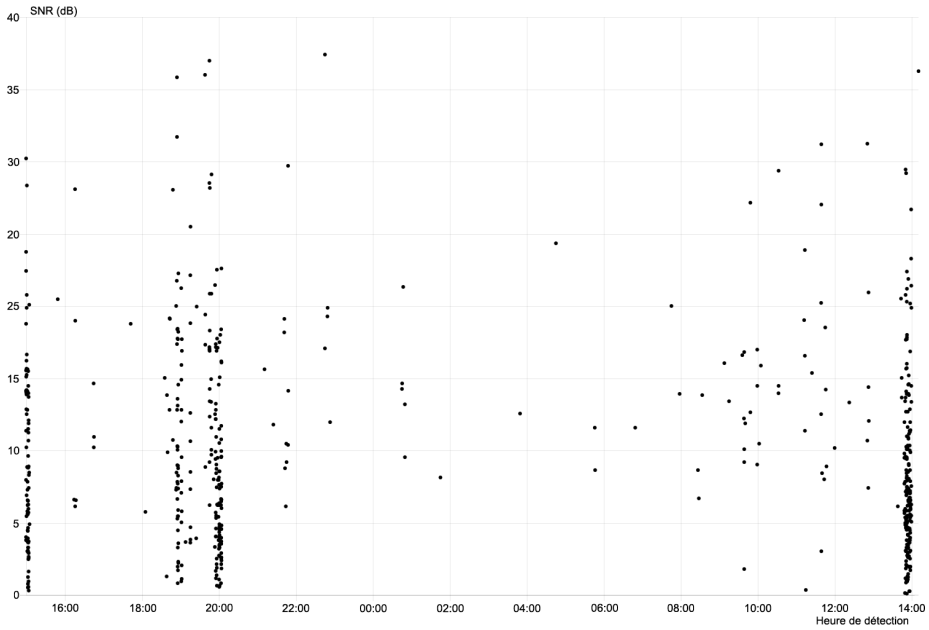


FIGURE 6.2 – Répartition des détections durant les 24 heures de test en fonction du niveau de bruit

Nous pouvons donc nous demander si ces détections avec un très faible SNR sont susceptibles d'être rejetées. Le graphique 6.3 [1] présente donc la durée d'un

1. Sur ce graphique le son le plus long n'est pas représenté, dans un soucis de commodité de lecture, ce son ayant une durée de 12.42 secondes et un SNR de 13.41dB

son en fonction du niveau de bruit, nous observons donc que les sons avec un faible SNR ne sont pas uniquement des détections courtes, ce qui aurait pu être un marqueur de mauvaise détection lors des pics de détection précédemment présentés. Nous remarquons cependant pour la plus courte durée de détection possible une forte fréquence de très faibles SNR.

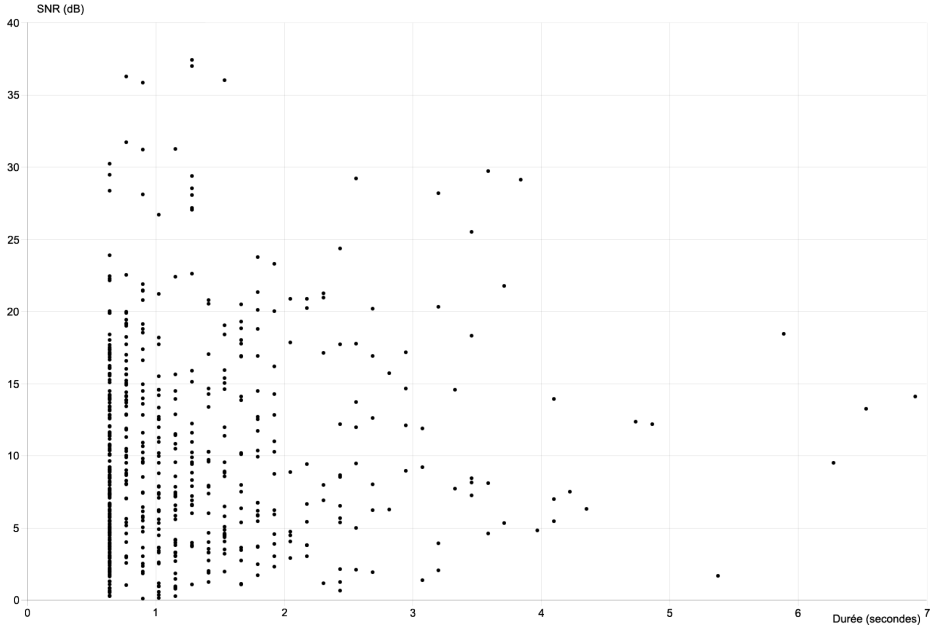


FIGURE 6.3 – Répartition de la durée de détection en fonction du niveau de bruit

Sur les 566 détections durant ces 24 heures, seules 58 ont été identifiables lors de l'écoute par un opérateur humain, les autres sons étant trop faibles ou ne comportaient rien de significatif. Parmi ces sons nous avons pu identifier 39 chocs, 11 chocs multiples, 5 bruits de verre, 1 comportant des tintements, et 2 sons que nous n'avons pu classifier.

Les sons que nous avons identifiés comme chocs correspondent d'après nous à des fermetures de placard ou des objets que l'on entrechoque en les posant sur une table ; ces sons correspondent tous à la classe *DoorClapping* d'après le système nous considérons donc que cette classification est cohérente par rapport à la classification que nous avons effectué. Nous pouvons cependant noter que la télévision utilisée en continu et à un volume élevée par notre testeur n'a pas été détectée par le système.

Les sons identifiés comme chocs multiples correspondent à une série rapprochée temporellement de chocs entre objets, ces dernier ont été reconnus en *FemaleCry* pour la plupart, avec un *GlassBreaking* et un *Cough*, ces dernières reconnaissances du système sont fausses.

Les sons de verre présentent principalement des chocs que nous avons identifiés comme provenant d'une source en verre ou plus exactement de la vaisselle, parmi les 5 sons reconnus nous avons d'après le système deux *GlassBreaking*, un *DoorClapping*, un *Yawn* et un *Paper* ; si les deux derniers sont inacceptables les autres restent cohérents. De plus le son identifié comme *DoorClapping* semble être produit par le choc entre une assiette et une surface en bois.

Le son de tintements a été identifié par le système comme *Water*, ce qui est une mauvaise classification.

Le tableau 6.1 présente une synthèse des reconnaissances que nous avons obtenu. Sur les 58 détections que nous avons pu identifier nous pouvons donc dire que nous sommes du même avis que le système pour 42 détections, soit 72,41% de bonnes reconnaissances en conditions réelles, ou 75% si l'on enlève les deux sons que nous n'avons pu identifier.

Classification	**Choc (39)**	**Chocs Multiples (11)**	**Verre (5)**	**Tintements (1)**
DoorClapping	39	0	1	0
Cough	0	1	0	0
GlassBreaking	0	1	2	0
FemaleCry	0	9	0	0
Paper	0	0	1	0
Water	0	0	0	1
Yawn	0	0	1	0

Tableau 6.1 – Matrice de reconnaissance du système par réseau de neurones par rapport à un opérateur humain

6.4 Conclusion

Les résultats obtenus sont très encourageants, en effet, les détections seules permettent de remarquer les zones d'activités de la personne. En effet ces der-

nières sont corrélables avec les activités relevées par le système ambiant déjà en place. Bien que n'apportant pas d'informations supplémentaires il pourrait agir comme un système moins invasif [2] pour relever les habitudes de vie de la personne.

Nous avons pu montrer que le système proposé était capable de bien reconnaître les sons reconnaissables par un humain avec 72.41%. Cependant nous pouvons dire que les classes de sons proposées ne sont pas en adéquation avec ce que le système est actuellement capable de détecter. Il est donc nécessaire d'améliorer la méthode de détection. De même il est important d'implanter une méthode de rejet. Les propositions que nous avions faites en section 6.1 après quelques tests permettent de réduire le nombre de mauvaises détections, mais réduisent dans une moindre mesure, les détections reconnaissables par un humain. Il semble donc indiqué de travailler également sur des paramètres internes au sons (stationnarité du signal par exemple) afin d'améliorer ce mécanisme. Il faudra probablement ajouter une classe à l'apprentissage avec les sons indéfinissables, afin de pouvoir les reconnaître et donc les exclure par le réseau de neurones. L'ajout d'une telle classe peut cependant avoir un effet néfaste et peut avoir tendance à absorber les résultats vers cette dernière. Un score de confiance sera donc à utiliser également en renfort de cette méthode. Ce score de confiance pourra être calculé à partir de la probabilité d'appartenance.

2. On parle ici des détection seules, sans analyse de sons. L'information étant réduite à l'activation de la détection.

Chapitre 7

Conclusions et perspectives

7.1 La reconnaissance de sons

Notre environnement est riche en informations sonores. Cependant de cette richesse la reconnaissance de parole et la reconnaissance du locuteur est le domaine le plus étudié. Le nombre de travaux et d'applications que nous utilisons au quotidien telles que : Amazon Alexa, Ok Google !, Siri, etc., en sont les témoins.

La richesse et l'utilité de l'environnement sonore ne sont néanmoins pas à prouver, il nous paraît en effet difficile d'appréhender notre quotidien sans l'utiliser. Bien que son utilisation et sa compréhension soient majoritairement inconsciente, il nous permet d'appréhender notre environnement et de nous apporter une information supplémentaire sur ce dernier. Les bruits de klaxons, sonneries diverses, sons et cris divers que nous produisons, toux et éternuements, nous permettent de mieux évoluer dans notre environnement quotidien, et sont porteurs de beaucoup d'informations.

Dans notre cas nous nous sommes proposés d'utiliser la reconnaissance de sons, pour la détection de dangers et d'activités, dans le cadre du maintien à domicile, de personnes âgées vivant seules dans les meilleures conditions possibles. Malheureusement pour ces dernières le dialogue n'est que peu présent dans leur vie. C'est pourquoi s'intéresser au son semble plus pertinent ou du moins une plus grande source d'informations sur le bien être de cette personne, car le son y est peu présent.

Nous avons donc proposé un système temps réel le plus efficace possible permettant de reconnaître divers événements acoustiques. Ces événements sont possiblement utilisables afin de reconnaître les activités quotidiennes d'une personne âgée.

7.2 Conclusions

Comme nous avons pu le remarquer dans l'état de l'art, les méthodes utilisées en reconnaissance de sons sont souvent des méthodes adaptées de la reconnaissance de parole ou de la reconnaissance du locuteur. De plus, nous avons pu remarquer que les méthodes adaptées de la reconnaissance du locuteur donnent bien souvent de meilleurs résultats que celles de la reconnaissance de parole. C'est pourquoi nous nous sommes proposés d'adapter les i-vecteurs à la reconnaissance d'événements de sons. Nous nous sommes aussi proposés d'utiliser une méthode de reconnaissance plus généraliste : l'apprentissage profond (*Deep Learning*). Cette méthode est utilisée désormais dans beaucoup de domaines et représente l'état de l'art dans chacun des domaines dans lequel cette technique est appliquée. Que ce soit en reconnaissance (image, vidéo, personnes, comportements, etc.), en génération (musique, texte, articles journalistiques, etc.) et en prédiction par le *q-learning* (apprentissage de marche pour un robot, AlphaGo, apprentissage des jeu Atari), les réseaux de neurones profonds ont permis une véritable révolution dans les domaines où ils ont été appliqués.

Nous avons donc proposé 3 différents systèmes, pour la reconnaissance de sons, au cours de cette thèse :

- les i-vecteurs ;
- les i-vecteurs, avec classification de ces derniers par réseau neuronal ;
- un réseau neuronal profond.

Les i-vecteurs étant reconnus comme ayant une bonne robustesse aux bruits pour la reconnaissance du locuteur, nous nous sommes également proposés de mettre à l'épreuve cette robustesse aux bruits pour les sons. Nous avons pu montrer que les i-vecteurs étaient également robustes aux bruits et que les sons peu bruités (SNR $\geq$ 40dB) ne dégradaient que très peu la précision du système. Nous avons dû utiliser un système de classification hiérarchique, en utilisant des macro-classes, afin d'améliorer la précision du système, en limitant le nombre de confusions possibles, et en spécialisant les différentes couches du système à un type de classification. En effet, plus le nombre de classes à reconnaître est grand, plus le nombre de confusions augmente. C'est pourquoi la création de macro-classes, qui seront donc discriminées par une autre couche, nous permet de réduire le nombre de classes à reconnaître par couche et donc d'augmenter le taux de bonnes reconnaissances.

Nous avons également travaillé sur des méthodes de fusion des scores des i-vecteurs. En effet chacune des méthodes d'évaluation retenues pour la classification des i-vecteurs présentait des avantages et inconvénients. Nous souhaitions donc, au lieu d'accepter les inconvénients d'une méthode, permettre de tirer le meilleur de chaque méthode pour améliorer les résultats. Nous avons commencé par utiliser des méthodes mathématiques déterministes, en proposant le NFD

et l'ANA.

Nous nous sommes alors proposés d'utiliser un réseau neuronal, pour remplacer les méthodes mathématiques précédemment évoquées. L'amélioration par rapport aux méthodes précédentes n'étant pas significatives, nous avons proposé d'évaluer directement les i-vecteurs en utilisant un réseau neuronal. L'augmentation du taux de bonnes reconnaissances a été très significative. Cependant nous avons pu noter que par rapport aux i-vecteurs, la dégradation des résultats dès l'introduction de bruit était plus grande. Cette dégradation restant tout de même inférieure à l'augmentation apportée par l'utilisation d'un réseau neuronal pour l'évaluation des i-vecteurs.

Nous souhaitions rendre plus accessible et plus modulaire le système, en effet l'introduction de nouvelles classes entraînerait une nouvelle phase d'investigation, pour définir les macro-classes et la hiérarchie des différentes couches du système précédents. Nous nous sommes alors proposés, d'après les résultats précédents, d'utiliser seulement un réseau neuronal profond pour effectuer la tâche de classification. Nous avons pu retrouver la résistance aux sons peu bruités apportés par les i-vecteurs tout en gardant l'amélioration de l'utilisation des réseaux neuronaux en évaluation. Bien que la précision en absence de bruit soit légèrement plus faible, la résistance aux bruits avec un SNR $\geq$ 20dB est nettement plus efficace.

Les réseaux de neurones apportent donc une très bonne solution aux systèmes de reconnaissance de sons, qu'il soient couplés à une autre méthodes, ou bien qu'ils soient utilisés seuls. De plus leur vitesse de traitement et le peu de calculs complexes à effectuer les rendent aisés à utiliser en temps réel. En conditions réelles, nous avons également obtenus des résultats très encourageants en utilisant les réseaux de neurones, ce qui conforte donc leur choix comme solution efficace pour la reconnaissance de sons.

7.3 Perspectives

Nous avons montré que l'utilisation d'un réseau neuronal profond était la meilleure alternative à ce jour, pour la reconnaissance de sons. Cependant l'utilisation de ce réseau implique un pré-traitement, même léger ; et la volonté des réseaux de neurones profonds est de ne pas utiliser de pré-traitement pour utiliser les données brutes en tant qu'entrée d'un réseau neuronal.

Il pourrait être également intéressant de créer de nouveaux paramètres ou de faire varier les paramètres présentés en entrée du réseau neuronal profond, dans le but d'améliorer la précision et la résistance au bruit. Dans cette optique il pourrait être judicieux de faire apprendre au réseau neuronal des sons bruités en spécifiant le niveau de bruit et le type de bruit, si ce dernier est détectable

et interprétable, en amont du réseau neuronal, et en conditions réelles.

La base de données présentée et utilisée pour l'évaluation du système, est de taille relativement réduite et très hétérogène structurellement. En effet les enregistrements varient de moins d'une seconde à plus de 30 secondes. De plus les classes de sons représentées ne sont pas suffisantes et ne permettent de détecter qu'une faible portion des activités de la vie courante ainsi que des potentielles situations de détresse. Il serait donc judicieux d'étoffer la base de données ainsi que d'homogénéiser la structure des données en présence.

Les systèmes proposés dans cette thèse peuvent avoir une application plus large. Par exemple d'un point de vue domotique ce système permettrait de déterminer les activités d'une personne et donc d'enclencher les systèmes connectés en rapport avec ce dernier. Par exemple le bruit de la porte d'entrée qui s'ouvre déclencherait, sous conditions, l'allumage de la lumière de l'entrée puis proposerait à l'utilisateur d'effectuer l'action la plus probable (exemple : allumer la cuisine, démarrer la bouilloire, etc.).

Les systèmes de reconnaissance de sons, posent problème d'un point de vue éthique. Nous avons proposé diverses méthodes pour rendre le système le moins invasif et le plus éthique possible. Il est toutefois nécessaire tous domaines confondus de continuer à proposer des solutions éthiques à un problème. Ces dernières étant souvent proche de la solution apportée et bien souvent sans perte de fiabilité dans le système proposé.

Bibliographie

[1] Chutes chez les personne âgées. http ://institutdeprevention.com/index.php ?option=com_content&view=article&id=2&Itemid=3. Accessed : 11-09-2017.

[2] Gartner says 8.4 billion connected "things" will be in use in 2017, up 31 percent from 2016. https ://www.gartner.com/newsroom/id/3598917. Accessed : 26-10-2017.

[3] Olly wright - une architecture de l'information éthique pour les entreprises. https ://hackmd.io/s/rJsSHk-ZW. Accessed : 26-01-2018.

[4] Maria Andersson, Stavros Ntalampiras, Todor Ganchev, Joakim Rydell, Jörgen Ahlberg, and Nikos Fakotakis. 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. IEEE, 2010.

[5] Jean-Louis Baldinger, Jérôme Boudy, Bernadette Dorizzi, Jean-Pierre Levrey, Rodrigo Andreao, Christian Perpère, François Delavault, François Rocaries, Christophe Dietrich, and Alain Lacombe. Tele-surveillance system for patient at home : the mediville system. In *International Conference on Computers for Handicapped Persons*, pages 400–407. Springer, 2004.

[6] Younes Bennani and Patrick Gallinari. On the use of tdnn-extracted features information in talker identification. In *Acoustics, Speech, and Signal Processing, 1991. ICASSP-91., 1991 International Conference on*, pages 385–388. IEEE, 1991.

[7] Ned Block. On a confusion about a function of consciousness. *Behavioral and brain sciences*, 18(2) :227–247, 1995.

[8] René Boite. *Traitement de la parole*. PPUR presses polytechniques, 2000.

[9] Pierre-Michel Bousquet, Anthony Larcher, Driss Matrouf, Jean-François Bonastre, and Oldrich Plchot. Variance-spectra based normalization for i-vector standard and probabilistic linear discriminant analysis. In *Odyssey*, pages 157–164, 2012.

[10] Pierre-Michel Bousquet, Driss Matrouf, Jean-François Bonastre, et al. Intersession compensation and scoring methods in the i-vectors space for speaker recognition. In *Interspeech*, pages 485–488, 2011.

[11] Albert S Bregman et al. *Auditory scene analysis*, volume 10. Cambridge, ma : mit press, 1990.

[12] Jianfeng Chen, Alvin Kam, Jianmin Zhang, Ning Liu, and Louis Shue. Bathroom activity monitoring based on sound. *Pervasive Computing*, pages 65–76, 2005.

[13] Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv :1406.1078*, 2014.

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

[15] Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv :1412.3555*, 2014.

[16] Patricia S Churchland and Terrence J Sejnowski. *The computational brain*. MIT press, 2016.

[17] Djork-Arné Clevert, Thomas Unterthiner, and Sepp Hochreiter. Fast and accurate deep network learning by exponential linear units (elus). *arXiv preprint arXiv :1511.07289*, 2015.

[18] Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine learning*, 20(3) :273–297, 1995.

[19] Stanislas Dehaene, Hakwan Lau, and Sid Kouider. What is consciousness, and could machines have it ? *Science*, 358(6362) :486–492, 2017.

[20] Najim Dehak, Reda Dehak, James R Glass, Douglas A Reynolds, and Patrick Kenny. Cosine similarity scoring without score normalization techniques. In *Odyssey*, page 15, 2010.

[21] Najim Dehak, Patrick J Kenny, Réda Dehak, Pierre Dumouchel, and Pierre Ouellet. Front-end factor analysis for speaker verification. *IEEE Transactions on Audio, Speech, and Language Processing*, 19(4) :788–798, 2011.

[22] Arthur P Dempster, Nan M Laird, and Donald B Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the royal statistical society. Series B (methodological)*, pages 1–38, 1977.

[23] Alain Dufaux. Detection and recognition of impulsive sounds signals. *Institute de Microtechnique Neuchatel, Switzerland*, 2001.

[24] Alain Dufaux, Laurent Besacier, Michael Ansorge, and Fausto Pellandini. Automatic sound detection and recognition for noisy environment. In *Signal Processing Conference, 2000 10th European*, pages 1–4. IEEE, 2000.

[25] Brian S Everitt. *Mixture Distributions—I*. Wiley Online Library, 1985.

[26] Kevin R Farrell, Richard J Mammone, and Khaled T Assaleh. Speaker recognition using neural networks and conventional classifiers. *IEEE Transactions on speech and audio processing*, 2(1) :194–205, 1994.

[27] Felix A Gers, Jürgen Schmidhuber, and Fred Cummins. Learning to forget : Continual prediction with lstm. 1999.

[28] Benjamin Graham. Fractional max-pooling. *arXiv preprint arXiv :1412.6071*, 2014.

[29] Klaus Greff, Rupesh K Srivastava, Jan Koutník, Bas R Steunebrink, and Jürgen Schmidhuber. Lstm : A search space odyssey. *IEEE transactions on neural networks and learning systems*, 2017.

[30] Richard Wesley Hamming. *Digital filters*. Courier Corporation, 1989.

[31] Andrew O Hatch, Sachin S Kajarekar, and Andreas Stolcke. Within-class covariance normalization for svm-based speaker recognition. In *Interspeech*, 2006.

[32] AL Higgins, LG Bahler, and JE Porter. Voice identification using nearest-neighbor distance measure. In *Acoustics, Speech, and Signal Processing, 1993. ICASSP-93., 1993 IEEE International Conference on*, volume 2, pages 375–378. IEEE, 1993.

[33] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8) :1735–1780, 1997.

[34] Dan Istrate. *Détection et reconnaissance des sons pour la surveillance Médicale*. PhD thesis, 2003.

[35] Rafal Jozefowicz, Wojciech Zaremba, and Ilya Sutskever. An empirical exploration of recurrent network architectures. In *Proceedings of the 32nd International Conference on Machine Learning (ICML-15)*, pages 2342–2350, 2015.

[36] Patrick Kenny. Joint factor analysis of speaker and session variability : Theory and algorithms. *CRIM, Montreal,(Report) CRIM-06/08-13*, 215, 2005.

[37] Patrick Kenny, Gilles Boulianne, and Pierre Dumouchel. Eigenvoice modeling with sparse training data. *IEEE transactions on speech and audio processing*, 13(3) :345–354, 2005.

[38] David Lyon. Facing the future : Seeking ethics for everyday surveillance. *Ethics and information technology*, 3(3) :171–180, 2001.

[39] Gary T Marx. Ethics for the new surveillance. *The Information Society*, 14(3) :171–185, 1998.

[40] Vinod Nair and Geoffrey E Hinton. Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th international conference on machine learning (ICML-10)*, pages 807–814, 2010.

[41] Aran Nayebi and Matt Vitelli. Gruv : Algorithmic music generation using recurrent neural networks. *Course CS224D : Deep Learning for Natural Language Processing (Stanford)*, 2015.

[42] Andrew Y. Ng. Preventing "overfitting" of cross-validation data. In *Proceedings of the Fourteenth International Conference on Machine Learning*, ICML '97, pages 245–253, San Francisco, CA, USA, 1997. Morgan Kaufmann Publishers Inc.

[43] Heinrich Niemann. *Klassifikation von mustern*. springer-Verlag, 2013.

[44] OECD. Oecd labour force statistics 2016.

[45] Aaron van den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, Andrew Senior, and Koray Kavukcuoglu. Wavenet : A generative model for raw audio. *arXiv preprint arXiv :1609.03499*, 2016.

[46] William H Press. *Numerical recipes 3rd edition : The art of scientific computing*. Cambridge university press, 2007.

[47] Simon JD Prince and James H Elder. Probabilistic linear discriminant analysis for inferences about identity. In *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*, pages 1–8. IEEE, 2007.

[48] Douglas Reynolds. Gaussian mixture models. *Encyclopedia of biometrics*, pages 827–832, 2015.

[49] Douglas A Reynolds, Thomas F Quatieri, and Robert B Dunn. Speaker verification using adapted gaussian mixture models. *Digital signal processing*, 10(1-3) :19–41, 2000.

[50] Douglas A Reynolds and Richard C Rose. Robust text-independent speaker identification using gaussian mixture speaker models. *IEEE transactions on Speech and Audio Processing*, 3(1) :72–83, 1995.

[51] Maxime Robin, Dan Istrate, and Jérome Boudy. Remote monitoring, distress detection by slightest invasive systems : Sound recognition based on hierarchical i-vectors. In *Engineering in Medicine and Biology Society (EMBC), 2017 39th Annual International Conference of the IEEE*, pages 2744–2748. IEEE, 2017.

[52] Maxime ROBIN, Grégoire NICOLLE, and Alexandre ROTA. Automatic sounds clustering approach based on a likelihood measure computation. In *JetSan 2015*, 2015.

[53] Mohammed Sehili. *Reconnaissance des sons de l'environnement dans un contexte domotique*. PhD thesis, 2013.

[54] Frank K Soong, Aaron E Rosenberg, Bling-Hwang Juang, and Lawrence R Rabiner. Report : A vector quantization approach to speaker recognition. *Bell Labs Technical Journal*, 66(2) :14–26, 1987.

[55] Andrey Temko and Climent Nadeu. Classification of meeting-room acoustic events with support vector machines and variable-feature-set clustering. In *Acoustics, Speech, and Signal Processing, 2005. Proceedings.(ICASSP'05). IEEE International Conference on*, volume 5, pages v–505. IEEE, 2005.

[56] Tijmen Tieleman and Geoffrey Hinton. Lecture 6.5-rmsprop : Divide the gradient by a running average of its recent magnitude. *COURSERA : Neural networks for machine learning*, 4(2) :26–31, 2012.

[57] D Michael Titterington, Adrian FM Smith, and Udi E Makov. *Statistical analysis of finite mixture distributions*. Wiley,, 1985.

[58] Michel Vacher, Dan Istrate, Jean-François Serignat, and Nicolas Gac. Detection and speech/sound segmentation in a smart room environment. In

Trends in Speech Technology, The 3rd International Conference on Speech Technology and Human-Computer Dialogue, IEEE, SpeD2005, 2005.

[59] Nancy VanDerveer. Ecological acoustics : human perception of environmental sounds, 1979. Thesis (Ph.D.) – Cornell University, 1979.

[60] Oriol Vinyals, Alexander Toshev, Samy Bengio, and Dumitru Erhan. Show and tell : A neural image caption generator. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3156–3164, 2015.

[61] Li Wan, Matthew Zeiler, Sixin Zhang, Yann L Cun, and Rob Fergus. Regularization of neural networks using dropconnect. In *Proceedings of the 30th international conference on machine learning (ICML-13)*, pages 1058–1066, 2013.

[62] Wayne Xiong, Jasha Droppo, Xuedong Huang, Frank Seide, Mike Seltzer, Andreas Stolcke, Dong Yu, and Geoffrey Zweig. Achieving human parity in conversational speech recognition. *arXiv preprint arXiv :1610.05256*, 2016.

[63] Yipin Zhou, Zhaowen Wang, Chen Fang, Trung Bui, and Tamara L Berg. Visual to sound : Generating natural sound for videos in the wild. *arXiv preprint arXiv :1712.01393*, 2017.

[64] Barret Zoph and Quoc V Le. Neural architecture search with reinforcement learning. *arXiv preprint arXiv :1611.01578*, 2016.

Appendices

Annexe A

Rappels mathématiques

Les sons que nous enregistrons sont stockés dans un espace discret. Ce qui implique que la variation des paramètre acoustiques est connues seulement en des instants définis, pour calculer les dérivées premières et secondes le calcul doit être fait par approximations [46]. Ces approximation nécessitent la connaissance de deux valeurs de la fonction autour du point concerné. Pour simplifier les calcul il est également possible d'utiliser 5 valeurs régulièrement espacées. Nous utiliserons donc les deux valeur précédant la valeur courante (c_k) appelées c_{k-2} et c_{k-1} ainsi que les deux suivant la valeur courante appelées c_{k+1} et c_{k+2}.

A.1 Calcul de la dérivée première

La formule d'approximation de la dérivée première (Équation A.1) s'obtient donc à partir de la décomposition en série de Taylor de la fonction.

$$\Delta c_k = \frac{(-(c_{k+2} - c_{k-2}) + 8 \cdot (c_{k+1} - c_{k-1}))}{12} \tag{A.1}$$

A.2 Calcul de la dérivée seconde

De façon analogue, pour le calcul de la dérivée seconde à partir de 5 valeurs régulièrement espacées, on obtient la formule A.2 à partir du développement en série de Taylor.

$$\Delta\Delta c_k = \frac{-(c_{k-2} - 16 \cdot c_{k-1} + 30 \cdot c_k - 16 \cdot c_{k-1} + c_{k+2})}{12} \tag{A.2}$$

Annexe B

Bibliothèques développées

B.1 Création d'une bibliothèque C++ d'enregistrement de sons

La création d'un système de reconnaissance de sons en temps réel nécessitait une fonctionnalité d'enregistrement non bloquante. C'est pourquoi nous avons proposé une librairie permettant d'enregistrer le son et/ou de l'utiliser directement de façon non bloquante.

La bibliothèque présente les méthodes suivantes :

— Diffuser ce qui est capté par le microphone vers le code
— Diffuser et enregistrer ce qui est capté par le microphone vers le code et un fichier
— Enregistrer ce qui est capté par le microphone vers un fichier
— Arrêter la diffusion et l'enregistrement
— Enregistrer durant un certain temps
— Récupérer les paramètre de la carte son
— Forcer le paramétrage de la carte son

Cette bibliothèque est disponible à l'adresse : `https://github.com/Waxo/ALSA_encapsulation`.

B.2 Création d'une bibliothèque d'extraction de paramètres acoustiques à partir d'un fichier wav en JavaScript

Il existe quelques bibliothèques d'extraction de paramètres acoustiques, on peut citer par exemple SPRO `https://www.irisa.fr/metiss/guig/spro/`. Cependant ces bibliothèques bien qu'efficaces sont peut modulables et difficilement adaptables. C'est pourquoi nous avons proposé une bibliothèque d'extraction de paramètres acoustiques et du calcul de dérivées utilisable le plus simplement possible.

La bibliothèque présente les méthodes suivantes :

— Extraire les paramètres acoustiques : MFCC, FFT, ZCR, SC, SRF, RER
— Enregistrer les paramètre acoustiques dans un format lisible par Alize `https://github.com/ALIZE-Speaker-Recognition`
— Calculer les dérivées premières et secondes selon deux méthodes différentes

Cette bibliothèque est disponible à l'adresse : `https://github.com/Waxo/sound-parameters-extractor`.