UNIVERSITÉ DE SHERBROOKE

Faculté de génie

Département de génie électrique et de génie informatique

DISCRIMINATION PAROLE - MUSIQUE POUR LE CODAGE UNIVERSEL DE L'AUDIO

Mémoire de maîtrise es sciences appliquées
Spécialité : Génie électrique

Ludovic TANCEREL

Sherbrooke (Québec), CANADA

Décembre 1999

Résumé

Le codage large bande de l'audio à bas débit est un enjeu majeur pour les télécommunications. Il est utilisé dans des applications de radio-diffusion sur Internet, de vidéo-conférence, de visio-téléphonie, et sera prochainement utilisé dans les futurs standards de téléphonie mobile.

Il n'existe pas de modèles efficaces capables de coder à la fois les signaux de parole et de musique à bas débit, c'est à dire à moins de 1 bit/échantillon. Les codeurs de parole, basés sur la prédiction linéaire et la quantification vectorielle ont une mauvaise qualité pour la musique. Inversement, les codeurs de musique, basés sur un codage par transformée ou en sous-bande ont une mauvaise qualité pour les signaux de parole.

Le but de cette maîtrise était de proposer une approche pragmatique pour le codage de l'audio à bas débit pour une application de radio-diffusion. Le système est basé sur une discrimination parole/musique, et un codage bi-modal, utilisant un codeur de musique, et un codeur de parole. Le système a été élaboré de façon à pouvoir utiliser différents modèles de codeurs.

La discrimination parole/musique proposée dans ce mémoire s'appuie sur des techniques de reconnaissances de formes. Une analyse long-terme du signal est effectuée pour extraire 5 paramètres, basés sur les propriétés temporelles, fréquentielles et de stationnarité. Trois techniques de classification sont ensuite testées, les mélanges de gaussiennes, les K-plus proches voisins, et un perceptron multi-couches. Les performances de classification obtenues sont satisfaisantes, mais dépendent du délai que l'on se donne sur le signal.

Pour éviter les artefacts qui apparaissent lors des transitions entre les codeurs si celles-ci surviennent dans des segments à haute énergie du signal, les transitions sont strictement limitées. Les performances obtenues sont alors meilleures que celles d'un codeur pris individuellement.

Avant propos

Je souhaite remercier mon directeur, Professeur Roch Lefebvre, pour m'avoir proposé ce projet, et supervisé tout au long de cette maîtrise. Je tiens également à remercier très vivement Stéphane Ragot, qui m'a toujours soutenu et motivé pendant cette année de recherche, et qui m'a de plus énormément aidé pour l'intégration des codeurs dans le système.

Je remercie tous les chercheurs et étudiants du groupe de codage de parole et audio, et plus particulièrement Vesa, pour ses conseils notamment dans la rédaction des articles en anglais, ainsi que Francis et Milan.

Je tiens à remercier tous mes amis québecois, pour leur accueil, ils ont fait de ce séjour prolongé au Québec une expérience unique. Je souhaite également saluer la *french connection*, notamment Antonin, Matthieu, David, Jérome et Géraldine, Pascal et Stéphane.

Finalement, je tiens à remercier mes parents qui m'ont toujours supporté pendant mes études en France et au Québec, et Virginie, avec qui les heures passées au téléphone m'ont grandement aidé à achever la rédaction de ce rapport.

TABLE DES MATIÈRES

LISTE DES FIGURES

Chapitre 1

Introduction

Les communications numériques sont rapidement devenues indispensables dans notre société moderne. Les données transmises sont compressées pour occupper moins d'espace. Par exemple, la bande téléphonique transmet les données dont les fréquences sont comprises entre 300Hz et 3400Hz. La qualité de cette compression n'est pas parfaite, mais suffisante pour un dialogue entre deux personnes.

Cependant, la demande du marché et les progrès technologiques poussent à augmenter cette bande de fréquence. La transmission de l'audio large bande, dont les fréquences sont situées entre 50 Hz à 7000 Hz, est donc devenue un nouvel enjeu pour la tranmission de données audio. La qualité correspond à celle de la radio AM, elle donne une impression de communication face-à-face. Le son est plus réaliste, le locuteur semble plus présent et en outre, la qualité de données telles que la musique est beaucoup mieux conservée.

L'audio large bande est déjà très répandue en vidéo-téléphonie, vidéo-conférence et en radio-diffusion. Les standards de télécommunication mobiles vont également évoluer vers une communication bi-directionnelle en parole large bande. Le codage universel de données audio à bas débit pour des données échantillonnées à 16 KHz est donc devenu un défi majeur pour les télécommunications.

La motivation de cette maîtrise est essentiellement due au fait qu'il n'existe pas de technologie mature pour le codage universel de l'audio. Le contrôle de la qualité de la transmission nécessite souvent l'intervention de l'utilisateur pour sélectionner le meilleur codeur suivant le contexte de communication. Typiquement, il existe deux types de codage de l'audio à bas débit. D'un côté, les codeurs de parole utilisent l'analyse par synthèse et sont basés sur les techniques de quantification vectorielle, de masquage et de prédiction linéaire qui fournit un modèle de production de la parole. La plupart d'entre eux sont basés sur les modèles CELP (Code Excited Linear Prediction). De l'autre, les codeurs audio large bande sont conceptuellement plus simples. Ils effectuent une décomposition fréquentielle du signal, par analyse en sous-bandes ou par tranformées, utilisent un modèle perceptuel basé sur les propriétés psycho-accoustiques de l'oreille pour adapter les pas de quantification scalaire et se basent sur le codage entropique. Ces deux approches se révèlent complémentaires dans le sens que ni l'une ni l'autre ne sont capables de coder efficacement à la fois la parole et la musique.

Une approche déjà envisagée est d'utiliser deux codeurs en boucle fermée, et de choisir le plus approprié. Le principal inconvénient est la complexité puisque cela implique le calcul de la synthèse pour les deux codeurs. Notre approche est basée sur une décision en boucle ouverte, le critère de sélection du codeur est la discrimination parole/musique. L'efficacité d'une telle approche impose un retard trop grand pour une utilisation temps réel, mais envisageable dans des applications telles que la radio-diffusion. Il est important de noter que ce type de solution deviendra obsolète lorsqu'une technologie mature permettra le codage universel de l'audio, mais semble viable à court terme.

La discrimination parole/musique est un processus qui a déjà été utilisé dans diverses applications telles que l'archivage de fichiers audio, les statistiques, la reconnaissance de données audio. L'approche classique dans ce genre de problème est l'extraction des paramètres puis la classification. L'application prévue impose au système un contrôle strict de la complexité, et donc un choix restreint de paramètres.

La définition de l'audio à de la musique ou de la parole peut sembler restrictive. Dans le contexte de codage du système, il est donc nécessaire de préciser ces termes. Le codage de parole est le plus

sensible car il utilise un modèle de production de la parole (section 4.1). Il est adapté à de la parole non ou faiblement bruitée. Dans le reste de ce mémoire, nous appelerons donc parole, de la parole non ou faiblement bruitée. La parole très bruitée, la musique ou simplement du bruit seront considérés comme étant de la musique.

Le mémoire se présente comme suit; Le chapitre 2 présente une analyse discriminante des signaux de parole et de l'audio. Elle se base sur les travaux déjà effectués sur la discrimination parole/musique et sur nos propres conjectures. Elle aboutit à la définition des paramètres utilisés pour la discrimination. Le chapitre suivant traite de plusieurs techniques de classification et des résultats obtenus à partir de notre espace de paramètres. Enfin, le chapitre 4 présente l'application de la discrimination appliquée au codage combiné de la parole et de la musique, et les performances obtenues.

Chapitre 2

Analyse discriminante des signaux audio

L'objectif de ce chapitre est de définir des caractéristiques pour les signaux que nous souhaitons discriminer. L'analyse des mécanismes de production de la parole est par exemple un moyen intéressant pour la distinguer de la musique. Les signaux de parole et de musique sont décrits dans la section 2.1. Certains travaux ont déjà traité le problème de la discrimination. Ils ont servi de base pour la recherche qui a été effectuée dans ce projet et sont présentés dans la section 2.2. Les sections 2.3 et 2.4 définissent les paramètres qui seront utilisés pour la discrimination.

2.1 Signaux audio : parole vs. musique

Les différences entre les signaux de parole et de musique sont très importantes, mais ce qui rend la discrimination parole/musique difficile, c'est la grande variété des signaux de musique. Les différences entre les signaux de parole se limitent aux différences biologiques des appareils phonatoires et aux langues utilisées, tandis que la musique est extrêmement variée, par les instruments qui la produisent et la façon dont les sons sont mélangés entre eux. C'est pourquoi la discrimination

parole/musique tend à devenir une discrimination parole/non-parole. Néanmoins, il est possible d'identifier certaines caractéristiques propres à ces deux signaux.

2.1.1 La parole et son modèle de production

La parole est un signal extrêmement bien connu puisqu'il a été étudié depuis de longues années aussi bien pour le codage, la reconnaissance, ou encore la synthèse. [KP95, Cal89] sont deux ouvrages qui décrivent ce signal de manière précise. Les caractéristiques de la parole qui sont présentées ici s'inscrivent dans le contexte de la discrimination par rapport à la musique.

La parole est un signal non-stationnaire, mais peut être considérée comme quasi stationnaire sur des fenêtres de l'ordre de 20 ms. Sa structure est complexe : tantôt périodique (ou pseudo-périodique) pour des sons voisés, tantôt aléatoire pour des sons fricatifs, tantôt impulsionnelle dans les phases explosives des sons occlusifs. La figure 2.1 montre un signal de parole prononcé par un locuteur masculin. Cette structure reflète l'organisation temporelle du mécanisme de phonation.

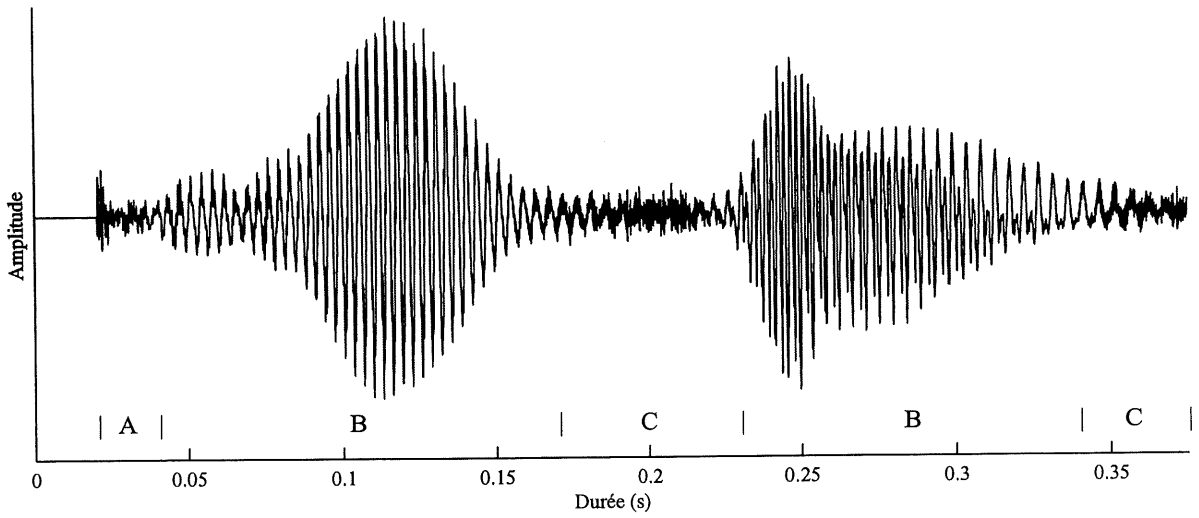

Figure 2.1 – Le signal vocal. Mots prononcés : /plus elle/, A : Impulsion, B : Pseudo-périodique, C : aléatoire

Modèle de production de la parole

Physiologiquement, la parole est produite par l'expulsion de l'air des poumons à travers les cordes vocales, le conduit vocal jusqu'à l'extrémité de la bouche. D'un point de vue traitement de signal, le mécanisme de production de la parole peut-être modélisé par un signal d'excitation, convolué par un filtre variant dans le temps, qui atténue ou amplifie certaines fréquences dans l'excitation. On dit que le conduit vocal est un système variant dans le temps puisqu'il consiste en une combinaison de la gorge, la bouche, la langue, le nez et les lèvres qui changent la réponse du filtre pendant l'élocution.

Les propriétés du signal d'excitation dépendent fortement du type de sons émis, soit voisés ou non-voisés. Dans le cas d'un son voisé, c'est un signal quasi-périodique généré par l'air qui fait osciller les cordes vocales. Les caractéristiques périodiques du signal dépendent de leur degré d'ouverture. Le conduit vocal étant généralement considéré comme linéaire, il ne modifie pas la périodicité du son émis. Dans le cas d'un son non-voisé, les cordes vocales sont complètement ouvertes, le signal peut alors être assimilé à du bruit. La figure 2.2a représente un exemple de signal voisé et de signal non-voisé. Dans le domaine spectral, en raison de la quasi-périodicité de l'excitation, le signal voisé a une structure harmonique régulière, comme illustré à la figure 2.2b. L'espacement entre les harmoniques est appelé fréquence fondamentale ou fréquence de pitch. L'enveloppe spectrale, appelée structure formantique est représentée par un ensemble de pics caractéristiques de la parole.

La figure 2.2c montre le spectre de puissance et la structure formantique d'un signal non-voisé. Contrairement au signal voisé, il contient beaucoup moins d'information spectrale. En outre, il ne possède pas de structure harmonique, et son énergie est plus faible.

2.1.2 Musique

Les signaux de musique sont difficiles à caractériser. En particulier, il n'existe pas de modèle de production simple à mettre en oeuvre. Il existe des modèles pour différents instruments, notamment en synthèse de signaux musicaux. L'ouvrage [Ols52] donne une définition des caractéristiques

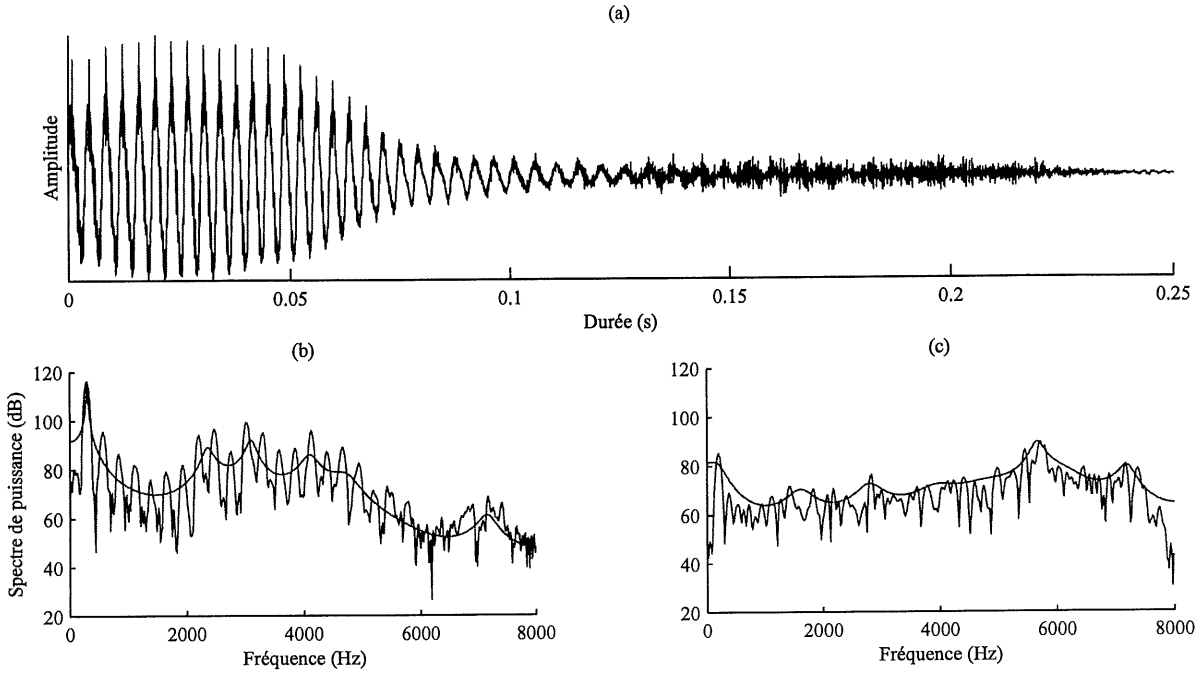

Figure 2.2 – Représentation temporelle et fréquentielle d'un segment de parole voisé et non-voisé. (a)Segment de parole voisée et non-voisée. (b)Spectre de puissance et structure formantique d'un segment de 20ms de parole voisée commençant à 15ms. (c)Spectre de puissance et structure formantique d'un segment de 20ms de parole non-voisée commençant à 15ms.

générales des signaux audio, et des instruments de musique. Leur diversité et leur enchevêtrement dans la musique rend impossible la définition d'un modèle général. Cependant, il est possible d'extraire certaines caractéristiques quasi-générales à ces signaux.

Forte dynamique énergétique

Généralement, la musique possède une puissance importante, c'est-à-dire que indépendemment du niveau sonore, le signal aura une grande amplitude, mais relativement homogène contrairement à la parole où la distribution énergétique dans le domaine temporelle varie énormément. La figure 2.3 nous montre une seconde d'un morceau de musique rock. Si on le compare à la figure 2.2 présentée

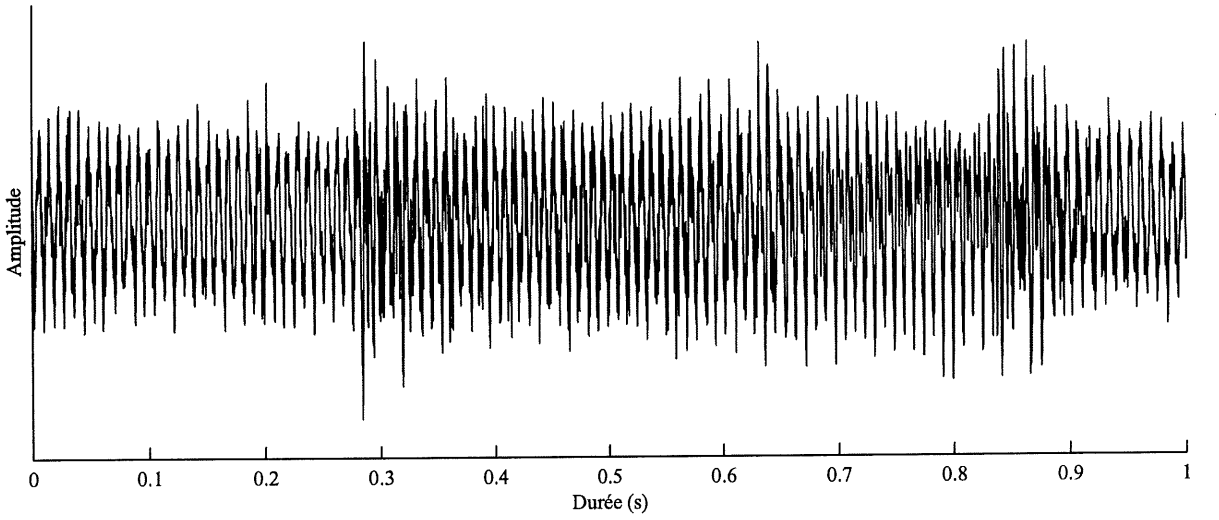

Figure 2.3 – Exemple de morceau de musique avec une dynamique importante et homogène.

à la section 2.1.1, on comprend tout à fait ce que l'on entend par dynamique forte et homogène en puissance.

Les signaux de musique ayant une forte composante rythmique ne possèdent pas cette caractéristique. La figure 2.4 représente une seconde de musique rap, les impulsions basse fréquence qui marquent le rythme de ce morceau sont ici très marquées. La dynamique du signal reste forte, mais elle n'est pas homogène. Il est donc nécessaire que ce type de signaux soit caractérisé par d'autres paramètres.

Structure harmonique

La structure harmonique est importante dans un signal de musique. Elle est le résultat de la superposition des harmoniques des instruments et des notes qui la composent. On peut l'observer aisément dans le domaine spectral. La figure 2.5 montre le spectre de puissance d'un accord d'orgue pour une trame d'une durée de 64 ms. On distingue un grand nombre d'harmoniques, mais même visuellement, il semble impossible de la définir de manière structurée. On en déduit que ce type de

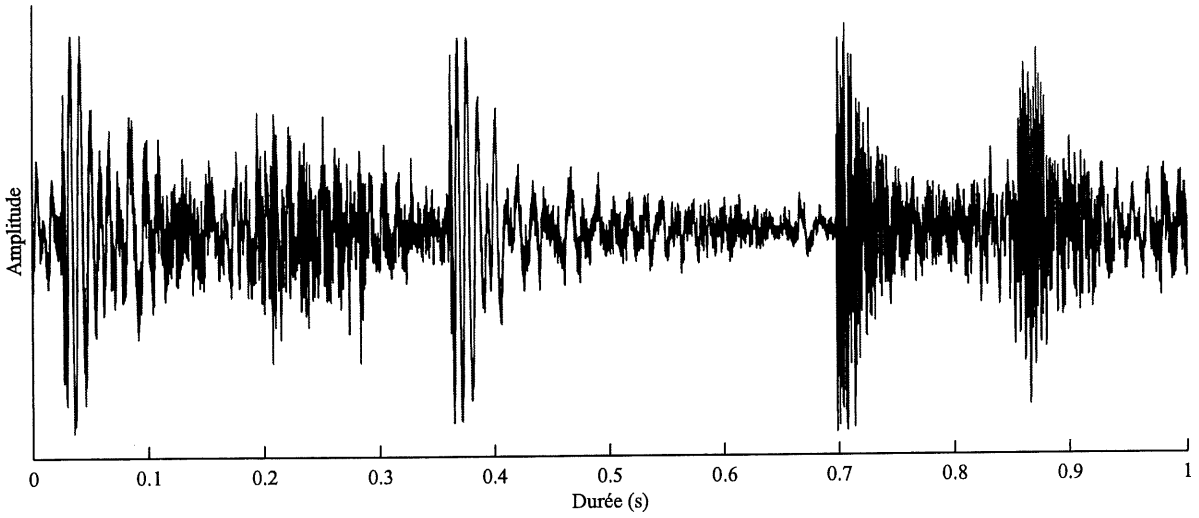

Figure 2.4 – Exemple de morceau de musique très rythmé.

signal est difficilement prédictible par rapport à un signal de parole voisée qui se représente aisément par une prédiction de pitch. Cette caractéristique devrait donc être discriminante, cependant le choix de la méthode à utiliser pour détecter cette structure harmonique est à définir.

Une fois encore, cette caractéristique ne comprend pas tous les signaux de musique. L'exemple en est donné sur la figure 2.6 qui illustre deux spectres de puissance, le premier pour une portion de musique live, l'autre de parole non-voisée pour une durée de 64 ms. L'information dans la structure harmonique n'est plus présente, il faut être en mesure de différencier d'une autre façon ces signaux, notamment en utilisant les propriétés de l'enveloppe spectrale.

Stationnarité

Une plus grande stationnarité de la musique par rapport à la parole a déjà été abordée dans le domaine temporel. Cette caractéristique devrait également être valable dans le domaine fréquentiel. Le signal de parole est considéré comme stationnaire sur des trames de 20ms (section 2.1.1). Pour différents signaux musicaux tels que le pop rock, la musique classique, le jazz, l'enveloppe spectrale

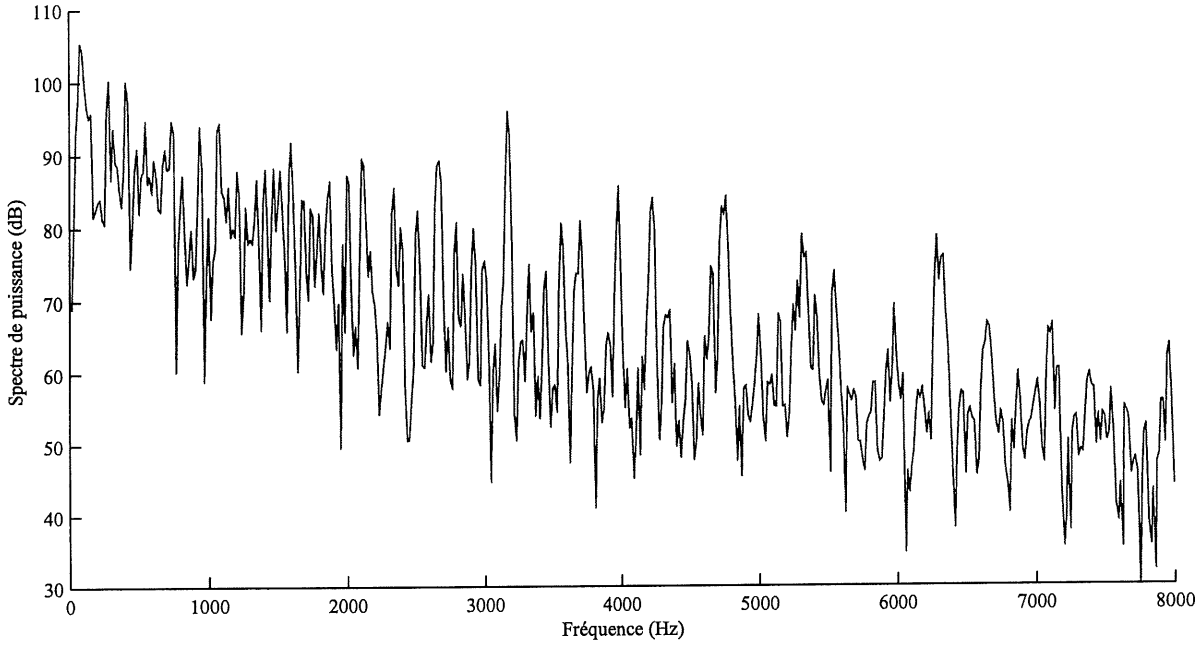

Figure 2.5 – Spectre de puissance d'un segment de 64ms d'un accord d'orgue.

évolue relativement lentement sur une durée supérieure à 100ms. La figure 2.7 présente l'évolution de l'enveloppe fréquentielle obtenue par analyse prédictive linéaire à 16 coefficients sur 10 trames de 20ms pour une signal de parole et un signal de musique classique.

L'allure des formants évolue assez rapidement pour la parole, tandis que pour la musique l'enveloppe reste relativement stable. Cette caractéristique est générale à la parole. Cependant, il est évident que certains signaux musicaux auront une enveloppe fréquentielle qui évoluera très rapidement. C'est une fois encore la complémentarité entre les paramètres qui permettra une discrimination générale.

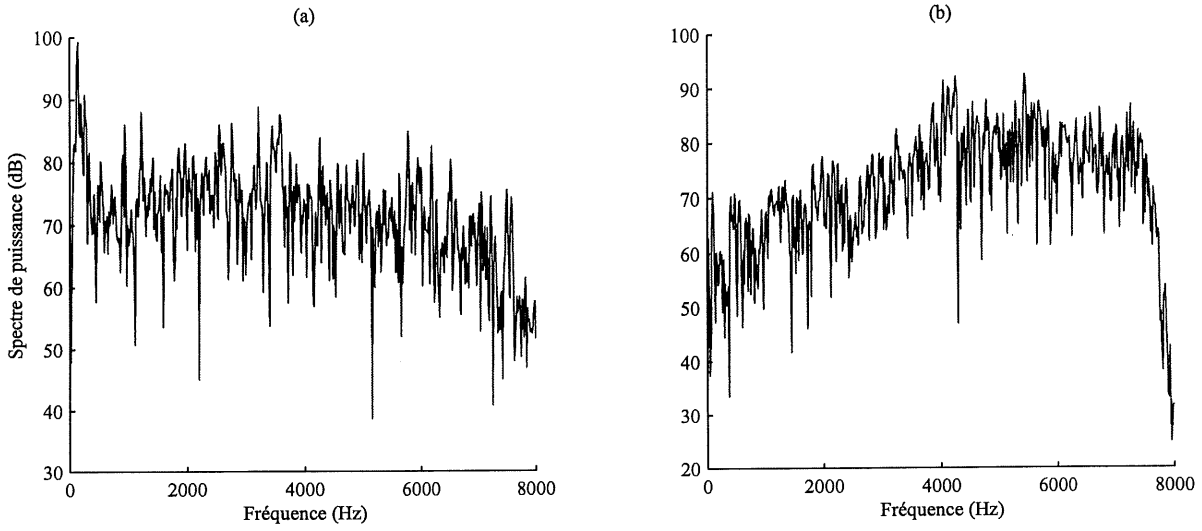

Figure 2.6 – Comparaison des spectres de puissance calculés sur 64ms d'un signal de musique non-harmonique (a) et d'un signal de parole non-voisée (b).

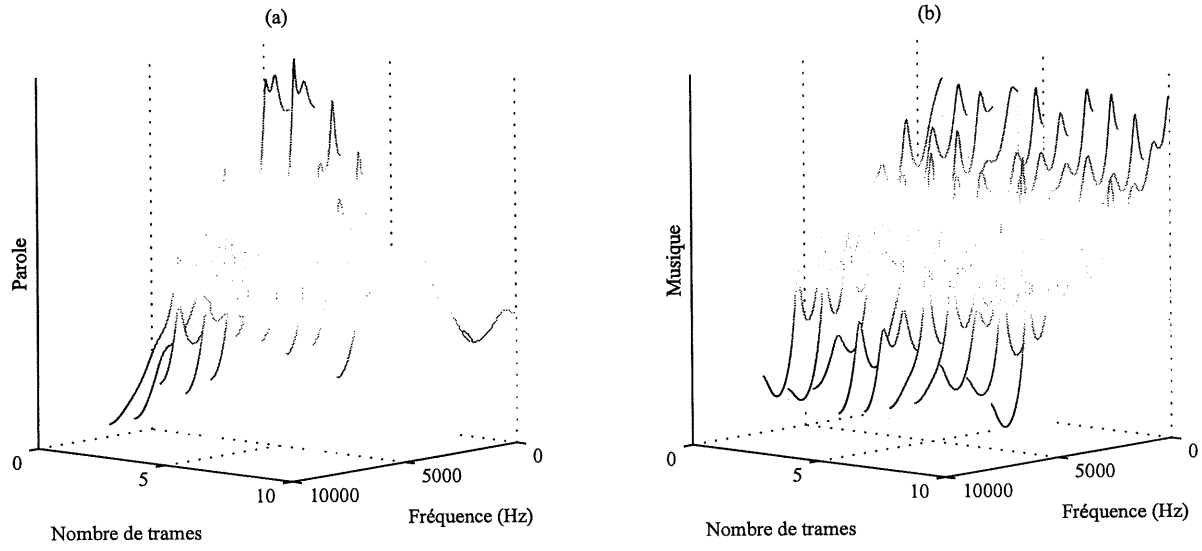

Figure 2.7 – Évolution de l'enveloppe prédictive linéaire à 16 coefficients sur 10 trames de 20ms. (a) Signal de parole. (b) Signal de musique.

2.2 Travaux précédents d'analyse

La discrimination parole/musique est un problème qui a déjà été abordé dans différents travaux. Ils ont permis de définir un nombre conséquent de paramètres. Les applications mentionnées dans ces travaux sont diverses et différentes de la notre. [Sau96] décrit une technique de discrimination pour la diffusion de radio FM essentiellement basée sur des paramètres temporels. [SS97] utilise la discrimination parole/musique pour un système de reconnaissance de parole pour des données audio générales. [SZ96] intègre la discrimination parole/musique au sein d'un système de transcription automatiques de données audio. Enfin, [CPLT99] fait une revue pour évaluer les performances de plusieurs paramètres pour la discrimination parole/musique.

Les résultats obtenus à partir de ces différents travaux montrent que la discrimination parole/musique nécessite de connaitre le signal sur une durée relativement longue pour obtenir des performances satisfaisantes. Cela n'est pas très gênant dans des systèmes de reconnaissance, mais cela constitue par contre un élément important dans le contexte du codage.

Il est nécessaire d'utiliser à la fois des paramètres temporels et fréquentiels dans la reconnaissance. Dans [SZ96], les paramètres cepstraux sont utilisés, comme dans la plupart des systèmes de reconnaissance de parole. Les paramètres utilisés dans les différents travaux sont décrits puis seront discutés ultérieurement.

2.2.1 Paramètres temporels

Les paramètres calculés dans [Sau96] sont exclusivement temporels. Ils sont essentiellement basés sur les propriétés statistiques du nombre de passage par zéro (NPZ). Ses propriétés sont décrites de façon précise dans [Ked86]. Le NPZ moyen fournit une mesure de la distribution de l'énergie spectrale. Les statistiques d'ordre supérieur permettent en outre de détecter rapidement et simplement les changements dans le spectre au cours du temps. [Sau96] complète ces statistiques en utilisant les propriétés de l'enveloppe temporelle. Ce dernier paramètre mesure le nombre de minima d'énergie à partir d'un certain seuil dans l'enveloppe. Les paramètres sont calculés sur des

fenêtres de 2.4 secondes, les performances moyennes de classification avoisinent les 96%.

Dans [SS97], on utilise 5 paramètres temporels parmi un ensemble de 13. La parole possède un pic d'énergie de modulation aux environs de 4Hz, qui résulte de l'articulation syllabique. Le premier paramètre temporel est donc obtenu en extrayant cette énergie en utilisant un filtrage en sous-bandes. Le second paramètre temporel mesure le nombre de trames ayant une puissance inférieure à la moitié de la puissance moyenne, sur une fenêtre d'une seconde. Deux paramètres sont basés sur les statistiques du NPZ, comme dans les travaux de [Sau96]. La moyenne et la variance de celui-ci sont calculées sur 1 seconde. Enfin, le dernier paramètre utilise la corrélation à long terme dans plusieurs bandes pour déterminer un niveau d'intensité rythmique sur une fenêtre de 5 secondes. Il permet de détecter des musiques rythmées tel que la musique techno, la salsa, ...

Dans [CPLT99], on a évalué les performances de trois paramètres temporels, l'amplitude, le NPZ et le pitch. Le paramètre d'amplitude est obtenu par un banc de filtres à échelle Mel. La variation de cette amplitude est également calculée sur une base de 5 trames consécutives. Le paramètre basé sur le NPZ consiste simplement à compter le nombre de passages par zéro toutes les 10ms, et estimer ensuite son évolution sur 5 trames successives. Le pitch, ou fréquence fondamentale a été introduit à la section 2.1.1. Il peut-être déterminé dans le domaine temporel et fréquentiel. C'est un paramètre qui n'avait jamais été utilisé précédemment, et pourtant il contient certainement beaucoup d'informations permettant de discriminer la parole de la musique. Il est calculé par corrélation temporelle du signal filté à 1kHz, et est affiné de manière fractionnaire dans le domaine spectral.

2.2.2 Paramètres fréquentiels

Dans [SS97], on présente plusieurs résultats concernant les caractéristiques fréquentielles. Les paramètres calculés sont les suivants :

1. Coefficient de chute de la distribution spectrale.
2. Centre de gravité de la distribution spectrale.
3. Amplitude de la variation du spectre.

4. Résidu cepstral.

Pour chacun de ces paramètres, la variance de celui-ci est évaluée également, sur une durée de 1 seconde. Le premier paramètre calcule la fréquence en dessous de laquelle le spectre possède 95% de son énergie. Ceci permet de distinguer les segments voisés des segments non-voisés. Le centre de gravité permet de différencier la musique à percussions de la parole, puisque celles-ci ajoutent du bruit dans les hautes fréquence, et donc élèvent le niveau du centre de gravité. L'amplitude de la variation du spectre est une mesure de différence d'amplitude spectrale trame par trame. Elle permet de donner une mesure de la stationnarité du signal. Elle est très variable pour la parole, tandis que pour la musique, le spectre évolue de façon relativement constante. Le dernier paramètre tente de distinguer les segments de parole non-voisée de la musique, en calculant les coefficients cepstraux réels et en effectuant un lissage de ceux-ci. La re-synthèse est alors plus fidèle à l'original pour les signaux de parole non-voisée.

2.2.3 Paramètres cepstraux

Les coefficients cepstraux sont souvent utilisés en reconnaissance de la parole ou du locuteur. En général, on utilise les coefficients cepstraux sur une échelle mel (MFCC, *Mel Frequency Cepstral Coefficients*). Ils ont la particularité de modéliser efficacement le système auditif humain. [SZ96, CPLT99] utilisent ces coefficients pour la discrimination . Le calcul des coefficients cepstraux est présenté sur la figure 2.8. Le module de la transformée de Fourrier discrète est calculé sur une trame de 10ms, une fonction non-linéaire est appliquée au spectre, puis on le filtre par un banc de filtres triangulaires à échelle mel. Le nombre de filtres utilisé est de 19 pour [CPLT99], et 14 pour [SZ96]. On calcule ensuite une tranformée de Fourier discrète inverse ou une transformée en cosinus discrète sur les coefficients issus du filtrage. Cette seconde transformée a pour propriété de décorréler les coefficients entre eux. Les coefficients cepstraux ainsi obtenus sont utilisés comme paramètres pour la discrimination.

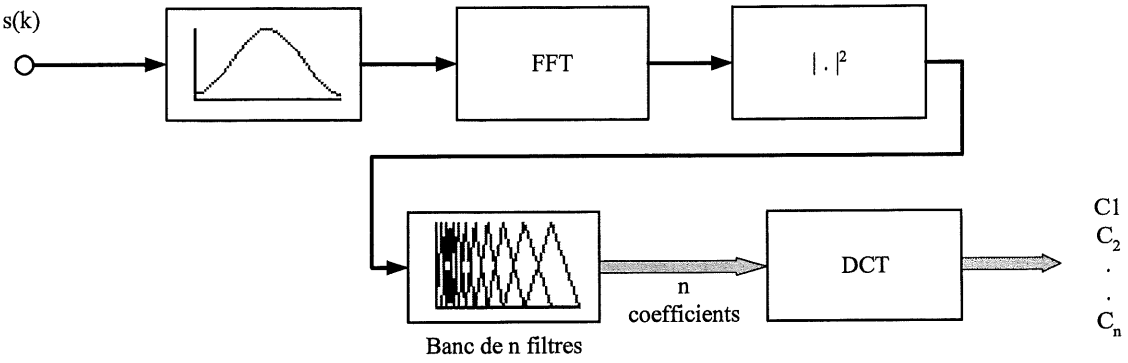

Figure 2.8 – Schéma bloc du calcul des coefficients cepstraux avec une échelle mel.

2.2.4 Discussion

Les différents paramètres utilisés dans les précédents travaux sont nombreux et plus ou moins performants pour la discrimination parole/musique. Certains paramètres paraissent même parfois peu justifiés quant aux caractéristiques des deux signaux. En revanche, il semble essentiel d'utiliser l'information relative à l'amplitude et au pitch, qui sont deux caractéristiques très discriminantes pour la parole et la musique. La stationnarité du signal dans le domaine fréquentiel est également une composante très importante qui a été utilisée dans [SS97]. Cependant, les paramètres qui sont définis pour extraire cette information de semblent pas très performant.

Les coefficients cepstraux sont en revanche très efficaces, mais complexes. De plus, ils correspondent plus à un contexte de reconnaissance de formes, qu'à celui de codage. C'est pourquoi ils ne seront pas retenus dans les paramètres pour la discrimination.

2.3 Réduction de dimensionalité, choix d'une représentation

La section 2.1 nous a permis de caractériser les signaux de parole et de musique. La section 2.2 nous donne un aperçu des paramètres qui possèdent des caractéristiques discriminantes. Pour des

raisons de complexité, il est important de définir un nombre minimal de paramètres. Ils devront de plus être complémentaires l'un par rapport à l'autre. En effet, il peut arriver que deux paramètres soient très discriminants, mais redondants entre eux. L'élimination d'un des deux permettra alors de diminuer l'espace de représentation. Par contre, il peut arriver qu'un paramètre soit faiblement discriminant, mais que combiné à un autre, il s'avère apporter beaucoup de nouvelles informations.

Les paramètres calculés dans le reste de la section ne seront pas tous retenus pour la discrimination pour les raisons citées précédemment. Les raisons pour lesquelles ces paramètres sont utilisés sont issus de nos propres constatations sur les signaux de parole et de musique, ainsi que des travaux précédemments effectués.

2.3.1 Énergie temporelle

L'utilisation de l'énergie temporelle a été justifiée à plusieurs reprises dans les sections précédentes. Elle représente une bonne approximation de l'enveloppe temporelle. Il existe plusieurs méthodes pour calculer cette enveloppe. Une d'entre elle est le filtrage passe-bas du signal redressé par une valeur absolue. Cependant, cette technique est relativement complexe puisque la fréquence de coupure du filtre doit être très basse et sélective. Une autre méthode est de calculer l'énergie dans des trames successives en utilisant une fenêtre de Hamming recouvrant les trames voisines. Le résultat obtenu est moins précis pour l'enveloppe, mais a l'avantage d'avoir une faible complexité.

Le paramètre d'énergie $e[k]$ est finalement calculé toutes les 5 ms sur une trame de 15 ms :

$$e[k] = \sum_{n=0}^{N-1} w_h[n]|s[n - kN]| \tag{2.1}$$

w_h est une fenêtre de Hamming de longueur 15ms, n est l'indice temporel, N la longueur de la trame et k son index. Le résultat pour un signal de parole de quelques secondes est présenté sur la figure 2.9b. L'amplitude de l'enveloppe étant dépendante du niveau du signal en entrée, il est nécessaire de le normaliser pour qu'il n'ait pas d'influence sur la décision. Deux types de normalisation sont possibles, la première en utilisant une base de fichiers tests pour fixer une échelle de normalisation, comme dans [CPLT99]. La deuxième est d'utiliser un coefficient adaptatif dépendant du signal en

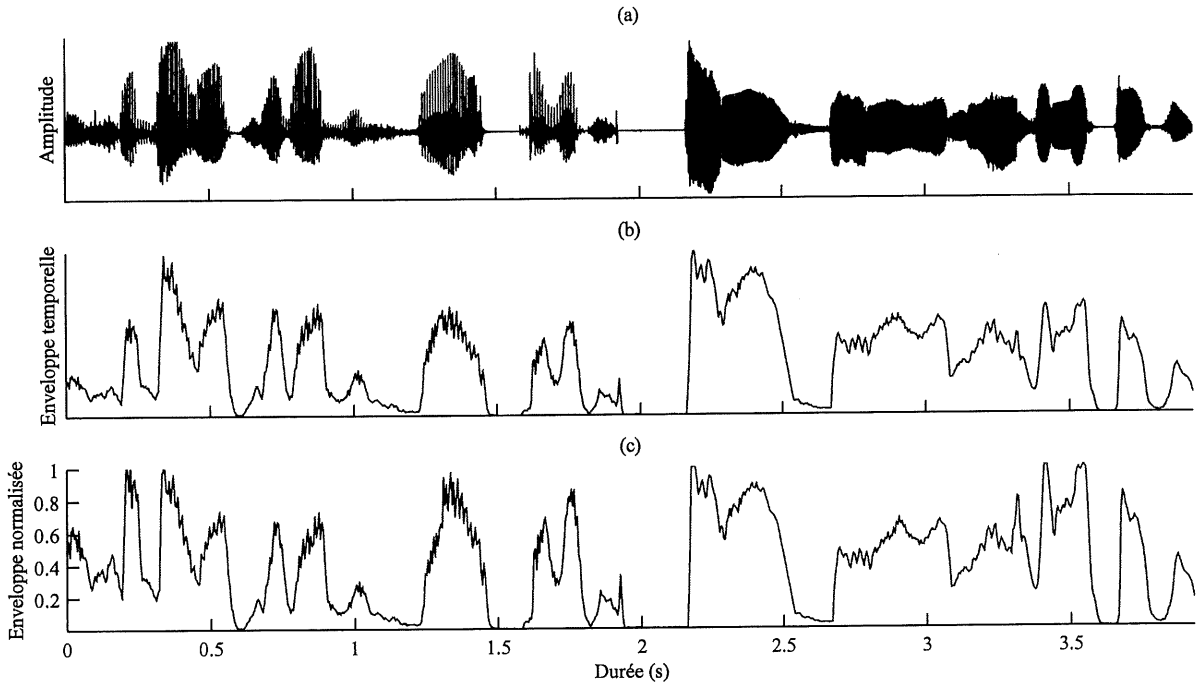

Figure 2.9 – (a) Signal de parole. (b) Enveloppe énergétique. (c) Enveloppe énergétique après normalisation.

entrée. Cette dernière solution semble plus robuste puisqu'elle ne dépend pas d'une base qui peut être insuffisante, en revanche elle est susceptible de modifier l'allure de l'enveloppe si la méthode n'est pas choisie avec précautions. Le coefficient de normalisation est obtenu en prenant le maximum de l'enveloppe sur les N_{trames} trames de 5ms précédentes. N_{trames} est égal au nombre de trames sur lesquelles seront estimées les statistiques des paramètres à la section 2.4.1. Le résultat de la normalisation est représenté sur la figure 2.9c. Elle est obtenue en divisant les résultats de l'enveloppe par le coefficient de normalisation. La forme de l'enveloppe temporelle est bien conservée, les valeurs prises par celle-ci sont toujours comprises entre 0 et 1.

2.3.2 Pitch et gain de corrélation normalisée

Il a déjà été montré par [CPLT99] que le pitch est un paramètre performant pour la discrimination parole/musique. La moyenne et la variance de celui-ci sont utilisés comme paramètres finaux. Cependant, il semble que plus d'informations peuvent être obtenues à partir du pitch. De plus, il paraît opportun de s'intéresser au gain de corrélation normalisée qui peut apporter de l'information supplémentaire sur le voisement.

Plusieurs techniques sont possibles pour déterminer le pitch, [Hes83]. L'algorithme utilisé est basé sur la corrélation croisée normalisée [JL00]. Le délai de pitch T est calculé toutes les 5ms. Le dédoublement de pitch est évité en vérifiant la cohérence de celui sur les 2 trames voisines de chaque côté de la trame courante. La figure 2.10 présente l'allure du délai de pitch et du gain de pitch pour un fichier de parole suivi de musique. On remarque très bien l'évolution caractéristique du pitch dans la parole voisée. Dans le cas de la musique, les variations du pitch sont sporadiques, elle est en effet souvent composée de plusieurs fondamentales. L'algorithme de recherche de pitch est de plus adapté aux caractéristiques de la parole. On remarque que les variations de pitch dans la musique sont souvent pratiquement nulles où très grandes. Pour cette raison, plutôt que de calculer des statistiques sur le pitch comme dans les travaux de [CPLT99], un coefficient de voisement v est calculé sur N_{trames}. Ce coefficient est basé sur l'évolution du pitch sur trois trames consécutives. Si cette évolution est comprise entre deux seuils déterminés empiriquement, alors le coefficient v est incrémenté, puisqu'on considère qu'on est en présence de parole voisée. La distribution obtenue pour ce paramètre est présentée à la figure 2.11.

Le gain de corrélation croisée normalisée g_p est obtenu de la façon suivante :

$$g_p(k) = \frac{\sum_{n=0}^{N-1} s(kN+n)s(kN+n-T)}{\sqrt{\sum_{n=0}^{N-1} s(k+n)^2}\sqrt{\sum_{n=0}^{N-1} s(k+n-T)^2}} \tag{2.2}$$

Il est très proche de 1 pour des segments de parole voisée, et prend des valeurs faibles pour des segments non-voisés, figure 2.10c. Les valeurs prises pour la musique sont diverses, mais rarement

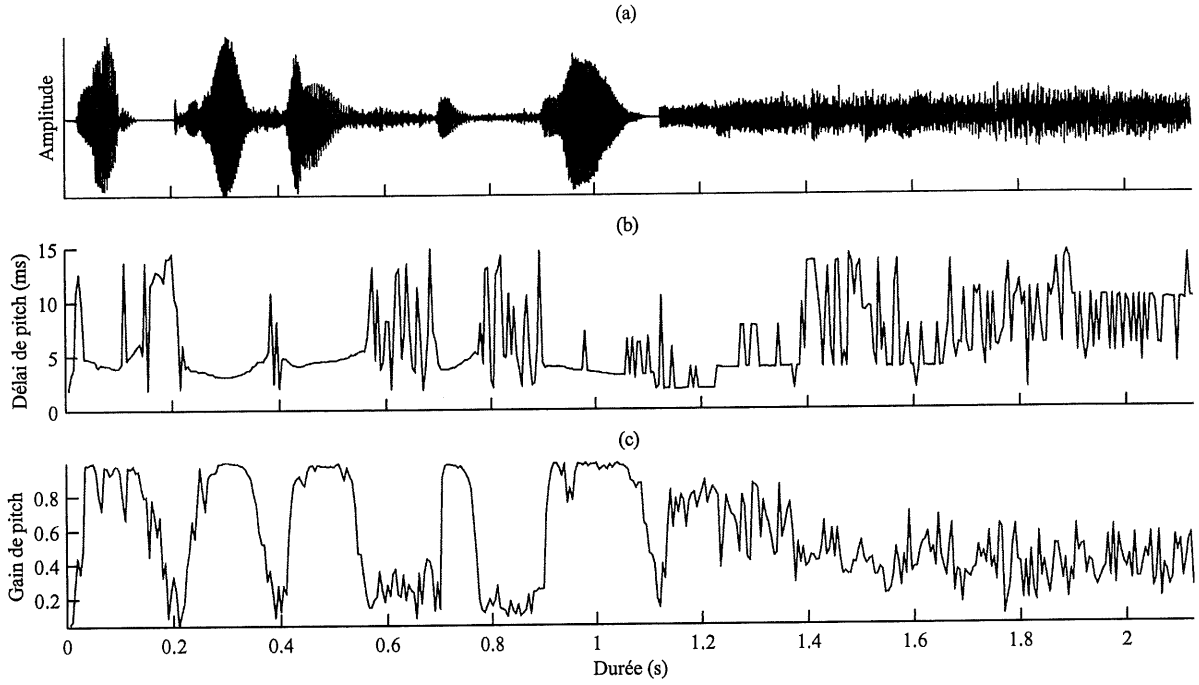

Figure 2.10 – (a) Signal de parole et de musique. (b) Délai de pitch obtenu par corrélation croisée. (c) Coefficient de corrélation croisée normalisée.

avec une variation d'amplitude aussi importante que la parole.

2.3.3 Énergie dans les harmoniques du pitch

Le signal de musique a une structure harmonique complexe (section 2.1.2). La parole, a en revanche une structure simple, constituée d'une fréquence fondamentale, et de ses harmoniques. Il semble donc intéressant de calculer l'énergie située dans les harmoniques du pitch. Celle-ci devrait avoir des valeurs importantes dans les segments de parole voisée. Une valeur robuste du pitch a déjà été calculée dans la section précédente. Nous proposons un algorithme chargé de calculer l'énergie dans ses harmoniques.

La figure 2.13 présente le schéma-bloc de l'algorithme. Une transformée de Fourier rapide sur

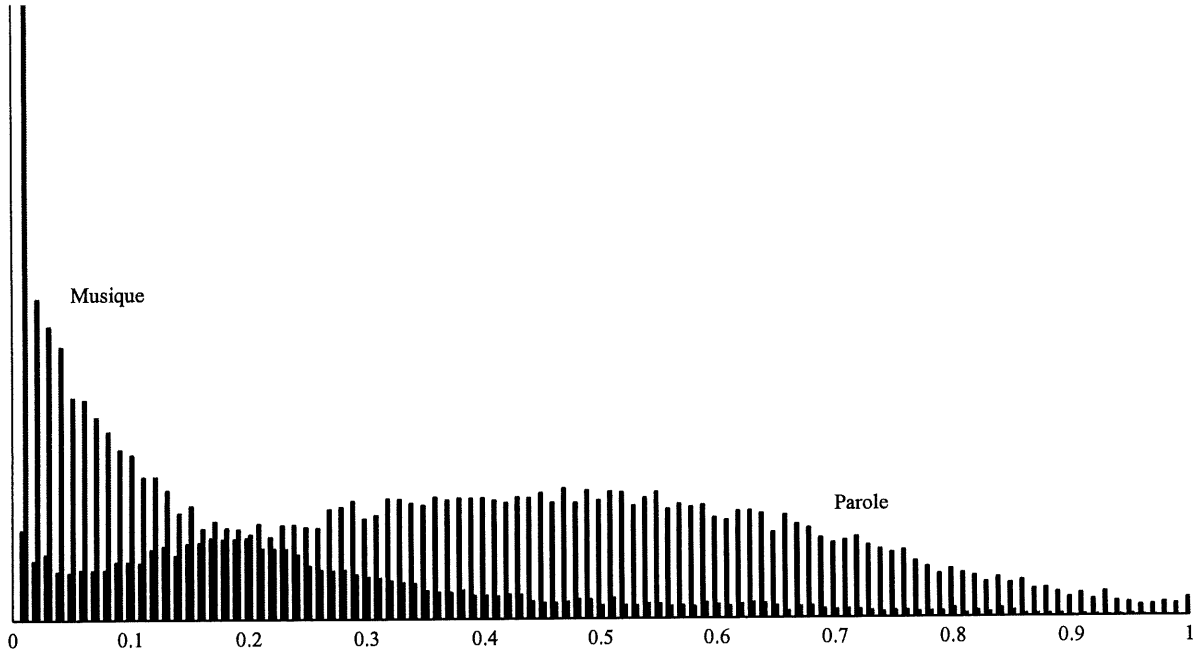

Figure 2.11 – Pdf du coefficient de voisement pour la parole et la musique.

1024 points est d'abord calculée toutes les 20ms sur le signal décimé à 4kHz et pondéré par une fenêtre de Hamming. Le signal est décimé pour avoir une plus grande résolution sur la transformée. On utilise le module au carré de la tranformée pour calculer l'énergie dans les harmoniques. Le pitch étant calculé toutes les 5ms, celui dont le gain de pitch est le plus grand est sélectionné. On utilise cette valeur pour déterminer les raies dans le spectre. Les raies sont déterminées en progressant vers les hautes fréquences du spectre. On recherche un maximum absolu dans une fenêtre de 6 échantillons autour de l'harmonique du pitch. Lorsqu'une harmonique est trouvée, la valeur du pitch est mise à jour. Une fois que ces raies sont déterminées, on calcule leur énergie en incluant 10 échantillons de chaque côté pour tenir compte de l'énergie répartie dans l'entourage des raies à cause du fenêtrage de Hamming. L'énergie ainsi obtenue est ensuite divisée par l'énergie spectrale totale. La figure 2.13 présente un exemple de peigne de raies. Pour des signaux de parole non-voisée, les raies n'existent pas. Il peut arriver tout de même que l'algorithme détecte des raies puisque une valeur de pitch lui est toujours donnée en entrée. Cependant, l'énergie obtenue n'est pas significative

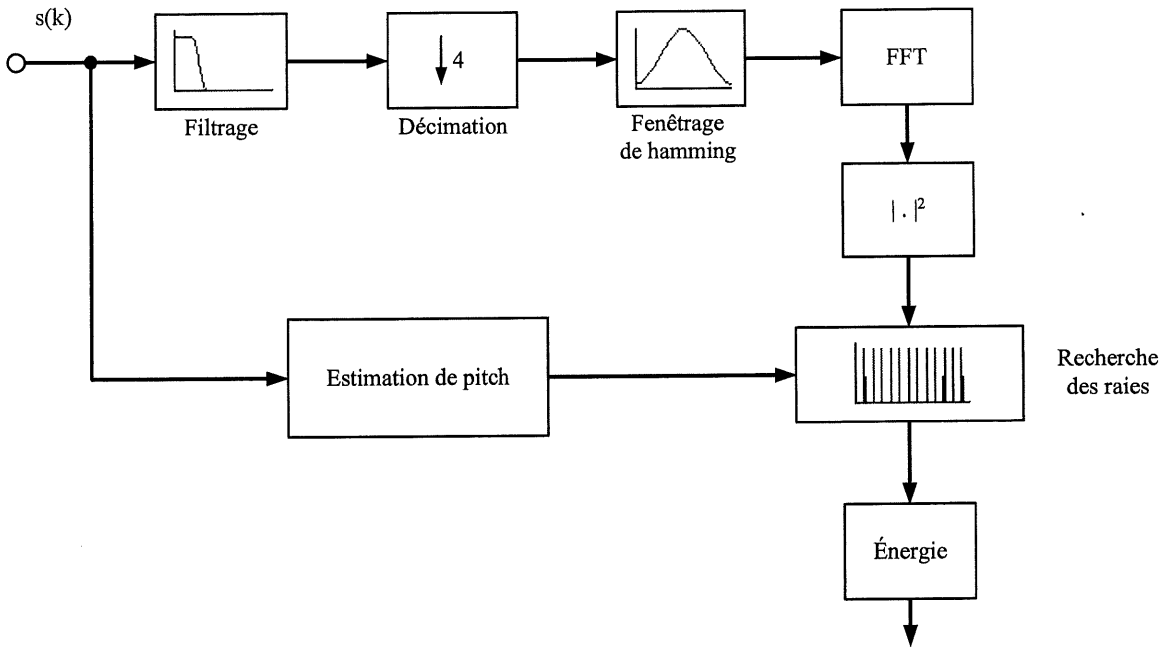

Figure 2.12 – Schéma-bloc du calcul de l'énergie dans les harmoniques du pitch.

puisque elle est répartie sur tout le spectre.

Le résultat obtenu pour des signaux de musique ayant une structure harmonique complexe est discutable, figure 2.13b. L'étalement provoqué par le fenêtrage de Hamming fait que les harmoniques vont se recouvrir entre elles, et donc rendre difficile la recherche de raies dans les harmoniques. D'un autre côté, ce manque de précision joue en notre faveur puisqu'on s'attend à ce que l'énergie dans les raies obtenues soit inférieure à celle de signaux de parole voisée. La figure 2.14b nous montre l'allure du paramètre obtenu pour un signal de parole suivi d'un signal de musique. L'énergie maximale obtenue ne dépasse pas 85% de l'énergie totale. Cela est vraissemblablement dû à l'étalement des raies dans le spectre. C'est essentiellement l'allure du paramètre qui est différente pour la parole et la musique. La performance de ce paramètre sera discutée à la partie 2.4.1, et notamment comparée au gain de corrélation croisée normalisée.

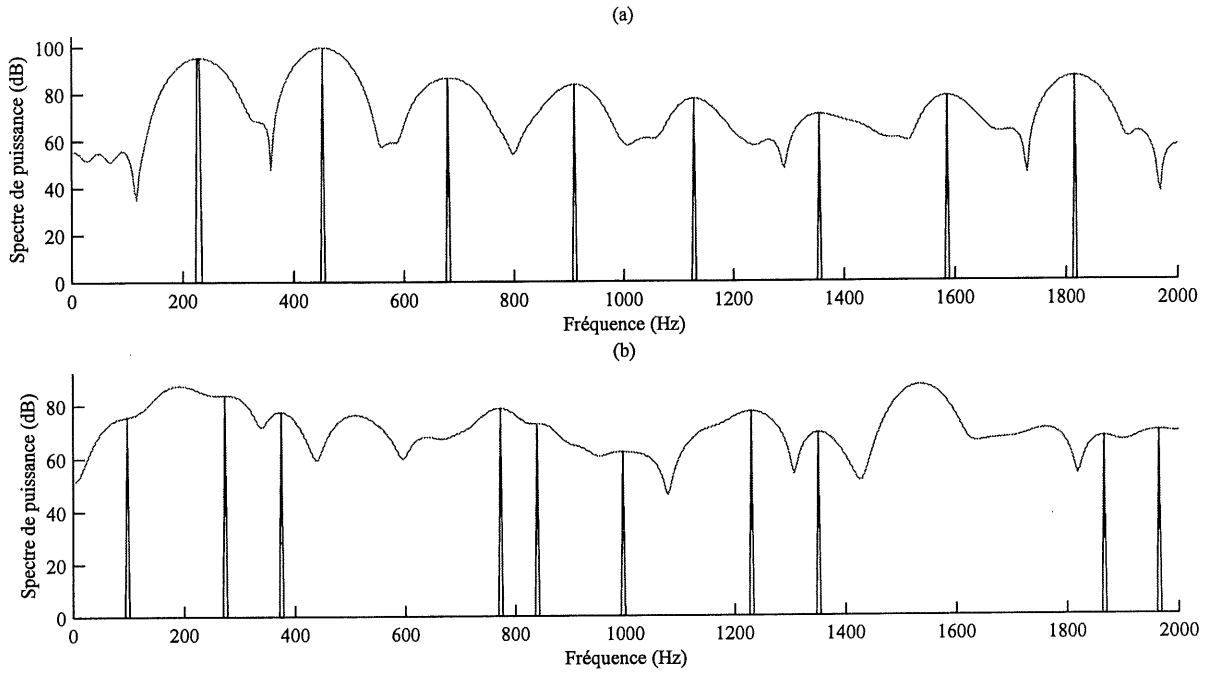

Figure 2.13 – Raies spectrales obtenues par recherche des harmoniques du pitch. (a) Segment de 20ms de parole voisée. (b) Segment de 20ms de musique.

2.3.4 Enveloppe spectrale estimée par prédiction lineaire

L'idée d'utiliser l'information dans le spectre et sa stationnarité a déjà été utilisée, notamment par [SS97]. Cependant, les paramètres tels que le flux spectral ne donnaient pas de résultats très satisfaisants. L'estimation de l'enveloppe spectrale par prédiction linéaire étant un concept fondamental du codage [Mor95], il semble intéressant de l'utiliser dans notre système.

La prédiction linéaire est celle utilisée dans le codeur ACELP présenté à la section 4.3.1. Le signal est d'abord décimé à 12.8kHz, on calcule son auto-corrélation toutes les 20ms, puis on en déduit 16 coefficients de prédiction linéaire par l'algorithme de Levinson-Durbin. Ces coefficients ayant de mauvaises propriétés de codage, on les tranforme dans l'espace appelé paires de raies spectrales (LSF, *Line Spectral Frequencies*) [Mor95]. Nous utilisons également cette représentation

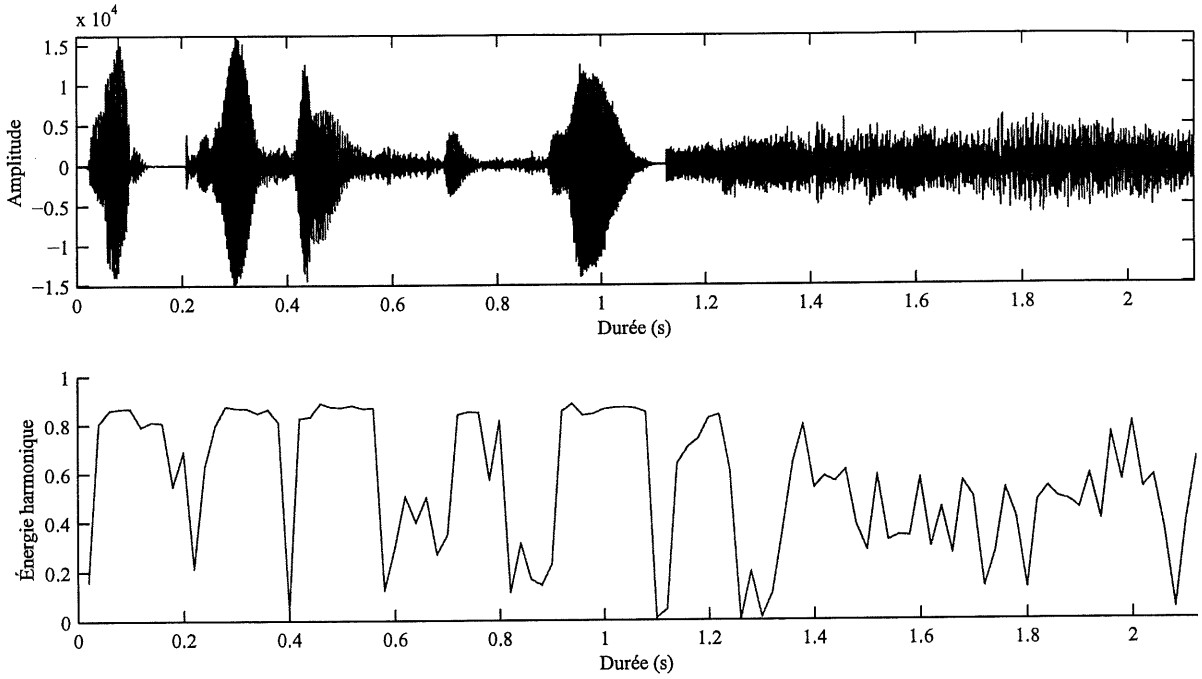

Figure 2.14 – (a) Signal de parole et de musique. (b) Énergie dans les harmoniques du pitch.

pour exploiter les caractéristiques des LSF. La figure 2.15 présente l'allure des LSF pour le signal de parole et de musique utilisé aux sections précédentes. Les coefficients LSF ont la particularité d'être distribués par ordre croissant, et sont compris entre 0 et π. Leur distribution est représentative de la forme de l'enveloppe spectrale. Par exemple, Les LSF d'une enveloppe plate seront uniformément réparties sur $[0, \pi]$.

L'enveloppe spectrale étant caractéristique de la structure formantique pour la parole, les LSF vont varier en fonction de l'évolution des formants, donc de l'articulation vocale. Les variations importantes des LSF prennent donc effet dans les phases transitoires de la parole. Pour la musique, la structure évolue, mais à une vitesse beaucoup moins grande. Ce paramètre est donc important puisqu'il devrait être complémentaire avec un paramètre comme le pitch, qui met plutôt en évidence les phases stationnaires voisées et non-voisées de la parole.

Pour caractériser cette variation, nous calculons la corrélation inter-trames des LSF. Ce pa-

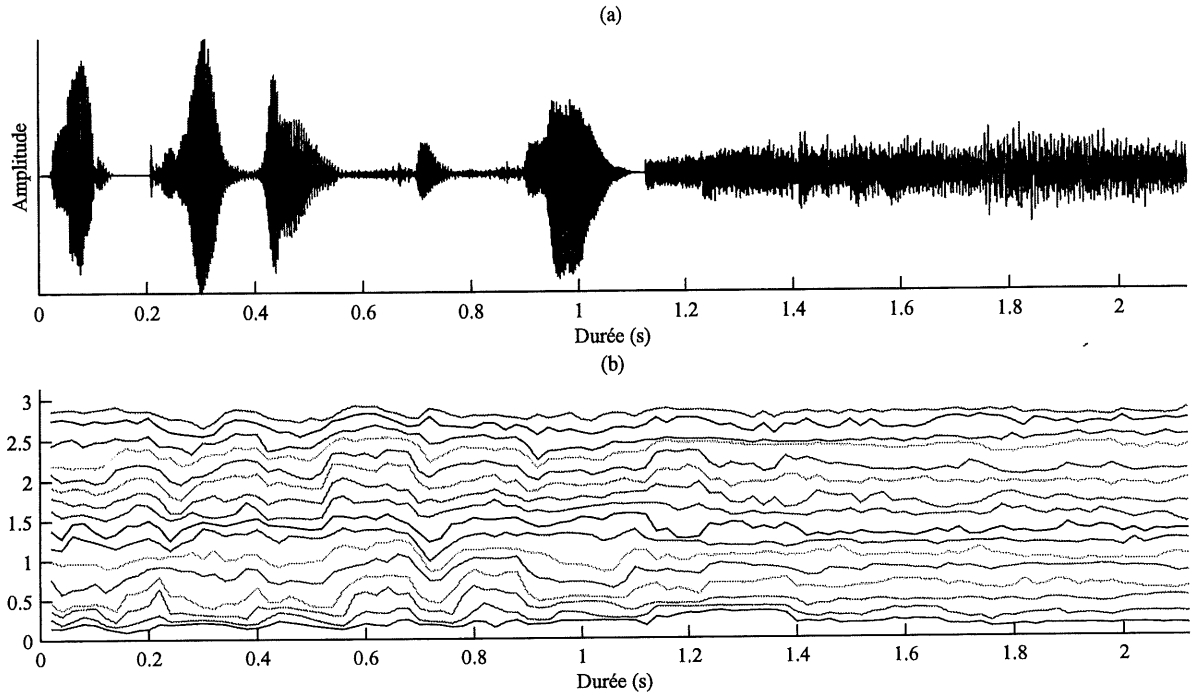

Figure 2.15 – (a) Signal de parole et de musique. (b) 16 paires de raies spectrales (LSF) calculées toutes les 20ms.

ramètre est calculé de la façon suivante,

$$r_n = \frac{\underline{\omega}(k)^T \, \underline{\omega}(k-n)}{\|\underline{\omega}(k)\| \cdot \|\underline{\omega}(k-n)\|}. \tag{2.3}$$

$\underline{w}(k)$ est le vecteur de LSF pour la trame k. n est l'indice de corrélation. La valeur de l'indice de la corrélation qui permet de discriminer le plus efficacement les signaux a été déterminé empiriquement, il s'agit de l'indice correspondant à r_2.

2.4 Choix des paramètres discriminants

Les paramètres calculés jusqu'à présent ont une faible valeur discriminante instantanée. Cela est compréhensible, puisque 20 ms de signal ne possèdent pas assez d'informations pour les distinguer,

cela est aussi vrai pour l'oreille humaine. Il faut utiliser l'information sur une durée plus longue. De plus, c'est souvent l'évolution du paramètre qui contient le plus d'information. C'est pourquoi nous calculons des statistiques sur les trajectoires. Il est nécessaire de trouver un compromis sur la durée de signal qui sera utilisée pour estimer les statistiques. Une durée longue permet d'améliorer le pouvoir disriminant, mais oblige à un grand retard sur la décision, ainsi qu'un manque de précision sur la décision instantanée.

2.4.1 Statistiques sur les trajectoires

Nous avons constaté que pour des paramètres tels que l'enveloppe temporelle, le gain de pitch, l'énergie dans les harmoniques du pitch et la corrélation inter-trames des LSF, c'est la trajectoire du paramètre qui est discriminante. Une façon simple de caractériser cette propriété, est d'estimer des statistiques sur ces trajectoires. Seules des statistiques du premier et du second ordre, c'est à dire des moyennes et des variances seront calculées. La durée sur laquelle sont estimés ces paramètres est importante pour réussir à discriminer efficacement les signaux. La figure 2.16 présente la distribution de la variance du gain de pitch pour un retard de 240ms, et de 480ms. La discrimination de ce paramètre passe de 27.8% d'erreur pour un retard de 240ms, et 20.2% pour un retard de 480ms. La section 4.3 présente les problèmes de transition dû à un retard trop grand. Un retard de 480ms correspondant à $N_{trames} = 24$ est finalement retenu pour le calcul de nos paramètres.

Statistiques sur l'enveloppe

La moyenne et la variance de l'enveloppe temporelle sont utilisées comme paramètre. La moyenne de l'enveloppe possède un pouvoir discriminant assez faible, mais conjuguée à la variance, elle améliore très sensiblement les performances. Cela peut se vérifier visuellement en observant la distribution obtenue en deux dimensions, figure 2.17. On remarque notamment que les segments de parole qui ont une variance de l'enveloppe aussi faible que les segments de musique ont généralement un moyenne également faible. Ces observations nous ont poussé à garder ces deux paramètres pour

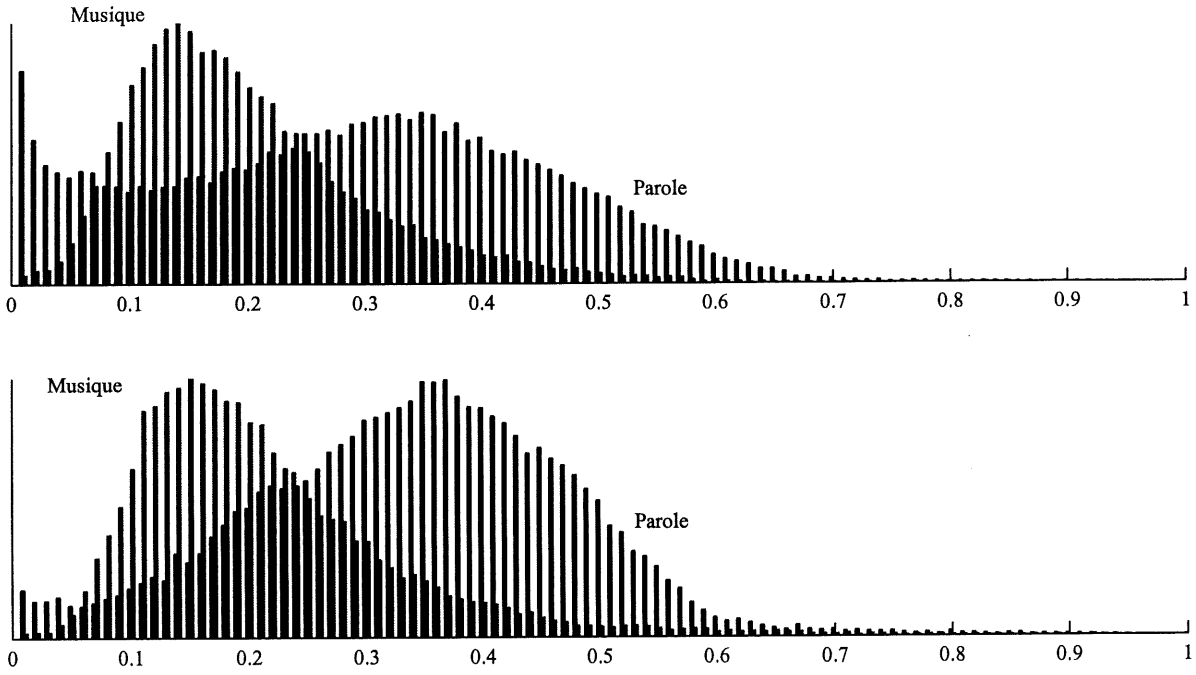

Figure 2.16 – Distribution de la variance de l'enveloppe temporelle pour des retards de 240ms et 500ms.

la discrimination.

Variance du gain de pitch vs variance de l'énergie dans les harmoniques du pitch

Nous avons remarqué que les trajectoires prises par le gain de pitch, et de l'énergie dans ses harmoniques sont assez similaires. Ces deux paramètres représentent le taux de voisement dans le signal. Pour éviter toute redondance dans les paramètres, il est nécessaire de n'en garder qu'un seul. La figure 2.18 représente la distribution conjointe de ces deux paramètres. Le pouvoir discriminant de la variance du gain de pitch étant supérieur, c'est ce paramètre qui sera conservé pour la discrimination.

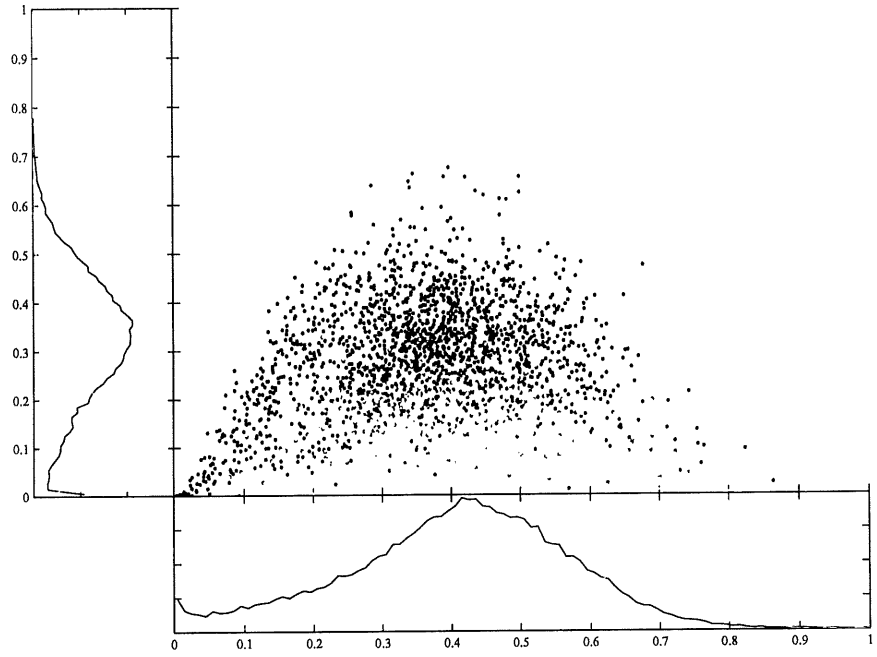

Figure 2.17 – Distribution de la moyenne et de la variance de l'enveloppe temporelle dans un espace à deux dimensions. En noir : Parole, en gris : Musique.

Variance de la corrélation inter-trames des LSF

Le dernier paramètre calculé est la variance de la corrélation inter-trames des LSF. Nous avons vu qu'il apporte de nouvelles informations notamment sur les transitions des signaux voisés à non-voisés. Le pouvoir discriminant de ce paramètre est à lui seul très intéressant, puisqu'il effectue une discrimination linéaire de 85.6%.

2.4.2 Autres possibilités

Les statistiques d'ordre 1 et 2 sont les techniques qui ont été utilisés dans les différents travaux de discrimination parole/musique. [Sau96] a par ailleurs exploité les statistiques d'ordre supérieur, il calcule notamment un moment d'ordre 3 sur le taux de passage par zéro. Cela permet en l'occurence

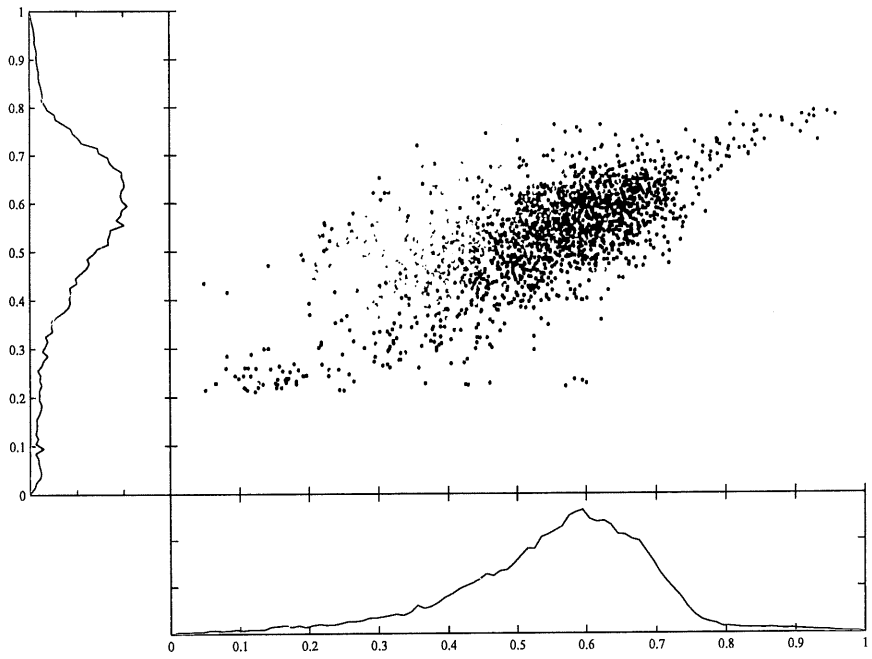

Figure 2.18 – Distribution de la variance du gain de pitch et de l'énergie dans les harmoniques du pitch. En noir : Parole, en gris : Musique.

de caractériser une distribution qui est assymétrique (*skewness*).

Il existe des techniques de classification qui permettent de retirer directement de l'information à partir de trajectoires de paramètres. Les chaînes de Markov cachées sont par exemple très utilisées en reconnaissance parole ou encore les algorithmes de classification par arbre. Cependant, ces techniques n'ont pas été retenues pour notre étude.

Chapitre 3

Discrimination et reconnaissance des formes

Le principe de la reconnaissance de formes s'apparente à classifier des objets dans un certain nombre de catégories ou de classes. L'intérêt est énorme puisqu'il peut être appliqué à toutes sortes de formes, et donc à des domaines aussi variés que la médecine, la biologie, le traitement de signal, ou tout autre application nécessitant l'analyse de données. La littérature est importante pour ce domaine, [Pat72, Fuk72] sont deux bons ouvrages de référence.

La reconnaissance de formes statistiques repose sur la théorie de la décision bayésienne, qui minimise la probabilité d'erreur. Malheureusement, cette décision nécessite des informations sur les données qui ne sont pas toujours disponibles. On utilise alors des classificateurs qui tentent d'estimer les informations nécessaires. Après une présentation de la théorie bayésienne, nous étudierons le principe des classificateurs par mélange de gaussiennes et par K-plus proches voisins. Nous aborderons ensuite une technique de classification moins classique, mais très utilisée, intéressante sur le plan pratique, les réseaux de neurones.

3.1 Formulation du problème de reconnaissance de formes

La reconnaissance de formes statistiques est une théorie qui repose sur une théorie de base très simple, et qui fonctionne bien en pratique. Elle s'applique lorsque le problème peut se représenter par un ensemble de paramètres $x(t_1), \ldots, x(t_n)$ de dimension n, à un instant donné t. Ces n données forment un vecteur X. Chaque élément $x(t_i)$ est une variable aléatoire, et X est appelé un vecteur aléatoire.

Par conséquent, chaque classe de données est représentée dans un espace à n dimensions par une distribution du vecteur X. Donc, pour élaborer un classificateur, il faut étudier les caractéristiques de la distribution de X pour chaque catégorie et en déduire une fonction discriminante.

Le choix des paramètres qui caractérisent les données est très important. Si chaque élément qui compose X contient peu d'informations, le nombre de dimensions nécessaires pour classifier les données sera grand. Cela rend très difficile le problème de reconnaissances de formes, puisque le nombre minimal d'éléments nécessaires pour définir correctement une distribution augmente exponentiellement avec la dimension des données.

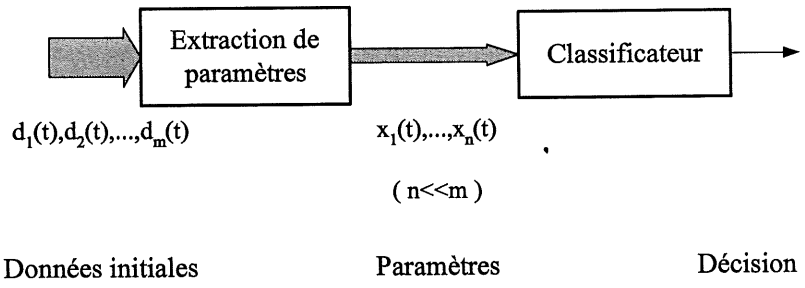

Figure 3.1 – Bloc diagramme du principe de la reconnaissance de formes.

Par conséquent, le problème de reconnaissances de formes se sépare en deux sous-problèmes, figure 3.1 : D'une part, la définition et l'extraction des paramètres qui vont représenter les données, c'est le travail qui a été effectué au chapitre 2, d'autre part, l'élaboration du classificateur.

En ce qui concerne l'élaboration du classificateur, on peut supposer que le vecteur observé est un vecteur aléatoire dont la fonction de probabilité conditionnelle dépend de la classe à laquelle il appartient. Si cette fonction est connue pour chaque classe, alors la classification se résume à tester des hypothèses statistiques.

3.2 Théorie de la détection et de la décision

3.2.1 Théorie bayesienne

Dans cette section, nous discutons seulement d'un problème à deux classes, w_1 ou w_2. Ce n'est pas restrictif, puisqu'il peut directement se généraliser à un problème multi-classes. Les densités de probabilités conditionnelles et les probabilités *a priori* sont supposées connues. On les désigne respectivement par $P(X|w_i)$ et $P(w_i)$. Par probabilité *a priori*, on comprend la probabilité qu'un évènement X appartienne à w_1 ou w_2. Dans le cas de la discrimination parole/musique, puisqu'on a aucune information sur la source, on suppose que ces probabilités sont de $1/2$.

Le principe de décision simplement basé sur les probabilités est le suivant :

$$P(w_1|X) \gtrless P(w_2|X) \rightarrow X \in \begin{cases} w_1 \\ w_2 \end{cases} \tag{3.1}$$

Les probabilités *a posteriori* peuvent être calculées en utilisant le théorème de Bayes qui est

$$P(w_i|X) = \frac{P(X|w_i)P(w_i)}{P(X)}. \tag{3.2}$$

Étant donné que $P(X)$ est commun quelque soit la classe considérée, la règle de décision pour la classe w_i est celle qui maximise l'expression $P(X|w_i)P(w_i)$. La décision peut alors se définir de la façon suivante

$$P(X|w_1)P(w_1) \gtrless P(X|w_2)p(w_2) \rightarrow X \in \begin{cases} w_1 \\ w_2 \end{cases} \tag{3.3}$$

ou

$$l(X) = \frac{P(X|w_1)}{P(X|w_2)} \gtrless \frac{P(w_2)}{P(w_1)} \rightarrow X \in \begin{cases} w_1 \\ w_2 \end{cases} \tag{3.4}$$

$l(X) = P(X|w_1)/P(X|w_2)$ est appellé le rapport de vraissemblance, $P(w_2)/P(w_1)$ est alors le seuil du rapport de vraissemblance pour la décision. Les équations 3.3 et 3.4 sont appelées *décision Bayésienne pour l'erreur minimale.*

La figure 3.2 illustre le seuil du rapport de vraissemblance pour la classification de deux sources gaussiennes de dimension 1. Pour évaluer les performances d'une décision, on définit la probabilité

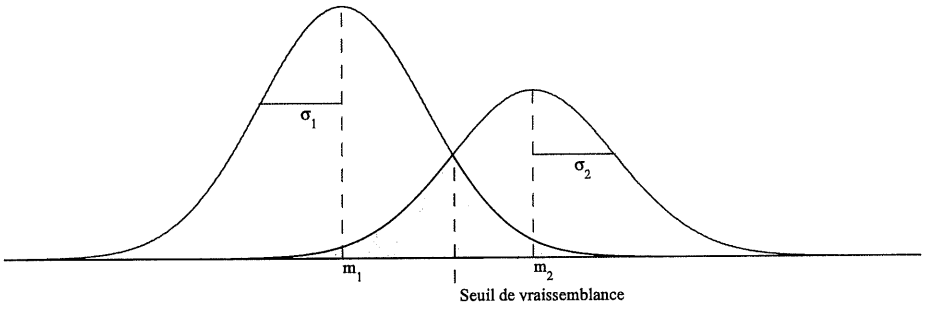

Figure 3.2 – Seuil de vraissemblance pour deux sources gaussiennes (m_1,σ_1) et (m_2,σ_2). L'aire de la surface en gris représente le taux d'erreur bayésien par rapport à l'aire totale sous les deux courbes.

d'erreur de classification. C'est la probabilité qu'un échantillon soit assigné à la mauvaise classe. On peut la calculer comme suit :

$$\epsilon = P\{erreur|w_1\}P(w_1) + P\{erreur|w_2\}P(w_2) \tag{3.5}$$

Si on nomme Γ_1 et Γ_2 Les régions de X telles que $P(w_1|X) > P(w_2|X)$ et $P(w_2|X) > P(w_1|X)$ respectivement. On a alors :

$$\epsilon = P(w_1)\int_{\Gamma_2} p(X|w_1)dX + P(w_2)\int_{\Gamma_2} p(X|w_1)dX \tag{3.6}$$

Cette expression correspond pour l'exemple de la figure 3.2, à l'aire de la zone grisée.

3.2.2 Estimation de la probabilité d'erreur bayésienne

L'analyse bayésienne standard est optimale au sens de la minimalisation de la probabilité d'erreur. Cependant, elle nécessite la connaissance des probabilités *a posteriori*, qui en pratique sont rarement connues. En effet, on ne dispose en général que d'un ensemble fini d'échantillons pour chaque classe, et nous devons utiliser ces échantillons pour définir notre classificateur. Il est possible d'établir théoriquement cette probabilité dans le cas de distributions particulières telles que une distribution gaussienne, mais une fois encore, en pratique, les distributions de points sont rarement totalement gaussiennes.

Pour estimer cette probabilité, on procède généralement par minoration, majoration. Dans [CH67], les auteurs établissent un théorème valable pour une analyse sur une grande quantité d'échantillons. si on appelle P_{Bayes} la probabilité bayesienne, et P_{NN} la probabilité d'erreur en utilisant la règle du plus proche voisin, ils montrent que pour une classification en M classes, on a :

$$P_{Bayes} > \frac{M-1}{M}(1 - \sqrt{1 - \frac{M}{M-1}P_{NN}}) \tag{3.7}$$

La règle du plus proche voisin sera définie à la section 3.3.2. La probabilité d'erreur bayésienne pour notre problème de classification sera estimée à la section **??**.

3.3 Méthodes de classification

Parmi les différentes techniques de reconnaissance de formes statistiques, on distingue les méthodes paramétriques et les méthodes non-paramétriques. Nous présentons ici une technique associée à chaque méthode, la modélisation des données par des gaussiennes et la règle des K-plus proches voisins. Une troisième technique utilisant les modèles connexionnistes sera finalement traitée.

3.3.1 Modélisation des densités de probabilité par des gaussiennes

L'objectif de la modélisation des données par des fonctions probabilistes est de représenter le processus aléatoire qui génère le vecteur X par un autre processus dont on connait le comportement. Cela revient donc à estimer les probabilités conditionnelles $p(X|w_i)$ par des fonctions qui seront paramétrées. Le but est alors de minimiser la fonction d'erreur entre les paramètres aléatoires, et leur estimée.

Les fonctions généralement utilisées sont des combinaisons linéaires de gaussiennes, en raison de la simplicité de manipulations de celles-ci. De plus, on sait que en général, des données générées aléatoirement ont naturellement une distribution proche de gaussiennes.

Il existe plusieurs méthodes pour minimiser l'erreur entre les paramètres et leur estimée : l'estimée Bayésienne, le *Maximum a posteriori*, ou le maximum de vraissemblance. Il est courant d'utiliser le maximum de vraissemblance dans le cas d'une modélisation par des gaussiennes. Nous expliquons le principe du maximum de vraissemblance dans le cas d'une seule gaussienne puis présentons sa généralisation dans le cas d'un mélange de gaussiennes.

Modélisation par une seule gaussienne

Nous souhaitons associer une gaussienne aux probabilités conditionnelles $p(X|w_i)$ de chaque classe. On appelle Θ et $\hat{\Theta}$ le vecteur paramètre de la gaussienne et son estimée. $\Theta = (\mu, \Sigma)$ sont le vecteur moyenne et la matrice de covariance de la gaussienne. La densité de probabilité de la gaussienne est

$$p(x|\Theta) = \frac{1}{(\sqrt{2\pi})^d \sqrt{|\Sigma|}} e^{-\frac{1}{2}(x-\mu)^T \Sigma^{-1}(x-\mu)} \tag{3.8}$$

$\hat{\Theta}$ est une fonction des vecteurs échantillons observés $Z = (X_1, X_2, \ldots, X_N)$ qui sont issus de la distribution à estimer. Une possibilité d'estimation est de chercher le Θ qui maximise $P(Z|\Theta)$ ou $\ln p(Z|\Theta)$. Cela veut dire qu'on sélectionne la valeur Θ pour laquelle Z est le résultat le plus vraissemblable. Le logarithme est introduit pour des raisons de simplicité de calcul, et ne change pas

le résultat, du fait de sa monotonie. Cet estimé est appelé le maximum de vraissemblance (*maximum likelihood estimate*). Il est solution des équation suivantes :

$$\left.\frac{\partial p(Z|\Theta)}{\partial \Theta}\right|_{\Theta=\hat{\Theta}(Z)} = 0 \quad ou \quad \left.\frac{\partial \ln p(Z|\Theta)}{\partial \Theta}\right|_{\Theta=\hat{\Theta}(Z)} = 0 \tag{3.9}$$

Recherchons dans un premier temps l'estimée $\hat{\mu}$ de la moyenne :

$$\frac{\partial \ln p(X_1, X_2, \ldots, X_N|\mu)}{\partial \mu} = \sum_{i=1}^{N} \frac{\partial \ln p(X_i|\mu)}{\partial \mu} \tag{3.10a}$$

$$= \sum_{i=1}^{N} \frac{\partial}{\partial \mu}[-\frac{1}{2}(X_i-\mu)^T\Sigma^{-1}(X_i-\mu) - \frac{1}{2}\ln|\Sigma| - \frac{1}{2}n\ln(2\pi)] \tag{3.10b}$$

$$= \Sigma^{-1}\{\sum_{i=1}^{N}(X_i-\mu)\} \tag{3.10c}$$

L'équation 3.9 nous permet donc d'en déduire que

$$\left.\sum_{i=1}^{N}(X_i-\mu)\right|_{\mu=\hat{\mu}} = 0. \tag{3.11}$$

L'estimée finale est donc

$$\hat{\mu} = \frac{1}{N}\sum_{i=1}^{N} X_i \tag{3.12}$$

qui est la moyenne des données échantillonnées. Le résultat de l'estimée de la matrice de covariance est donné dans [Pat72] :

$$\hat{\Sigma} = \frac{1}{N-1}\sum_{i=1}^{N}(X_i-\hat{\mu})(X_i-\hat{\mu})^T \tag{3.13}$$

Ce résultat est important puisqu'il permet de représenter simplement chaque classe par une gaussienne. Cependant, ce résultat estime les paramètres d'une gaussienne, et les données réelles n'ont pas exactement une distribution normale. Donc, une telle classification ne donnera pas les meilleurs résultats possibles.

Modélisation par un mélange de gaussiennes

Pour améliorer la modélisation paramétrique, on peut choisir de représenter les distributions par une combinaison linéaire de densités gaussiennes. Les probabilités conditionnelles pour chaque

classe i sont alors

$$P(x|w_i) = \sum_{k=1}^{N} \pi_k^{(i)} \mathcal{G}_{\mu_k^{(i)}, \Sigma_k^{(i)}}(x), \tag{3.14}$$

où $G_{\mu,\Sigma}$ est une fonction gaussienne, de vecteur moyenne μ et de matrice de covariance Σ. N est le nombre de gaussiennes utilisées. La figure 3.3 représente un exemple de modélisation d'une distribution à une dimension par plusieurs gaussiennes. Le nombre N de gaussiennes est en revanche

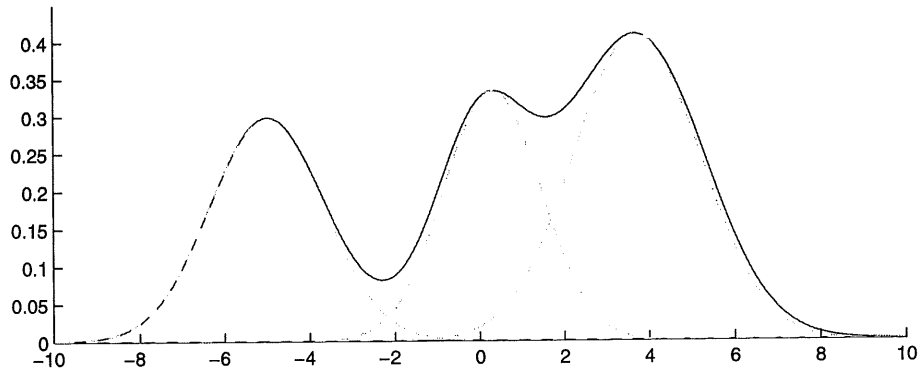

Figure 3.3 – Modélisation d'une distribution 1-d par un mélange de 3 gaussiennes.

à déterminer. En pratique, on fixe un nombre N empiriquement, puis on augmente sa valeur jusqu'à ce que le gain dans la classification ne soit plus significatif. Il reste ensuite à déterminer les paramètres μ et Σ des différentes gaussiennes qui composent le mélange.

L'algorithme EM (*Expectation-Maximization*) est un algorithme très utilisé en théorie de l'estimation [Moo96]. Il permet de converger vers le maximum de vraissemblance pour une distribution incomplète d'échantillons. Il est composé de deux étapes ; lors de l'étape E (*Expectation*), l'algorithme suppose que les paramètres estimés sont corrects, et cherche la distribution de données la plus vraissemblable par rapport à ceux-ci (dans notre cas, on détermine pour chaque vecteur de données son appartenance à telle ou telle gaussienne). Pour l'étape P, on suppose cette fois que la distribution est correcte, et maximise la vraissemblance des paramètres au regard de celle-ci. Cet algorithme s'applique a beaucoup de problèmes, nous donnons ici, son détail pour la maximisation de la vraissemblance dans le cas d'un modèle de mélanges de gaussiennes.

On initialise les N gaussiennes telles que les matrices de covariances Σ_i soient des matrices identités. Les pondérations π_i sont fixées égales à $\frac{1}{N}$. Les vecteurs moyennes sont choisis de la façon suivante : On recherche le vecteur moyenne de toutes les données. On choisit chaque vecteur μ_i à partir de celui-ci, dans la direction d'éléments de la distribution choisis aléatoirement.

Pour l'étape E, pour chaque point $x \in X$ de la distribution, et pour chaque $k \in (1, N)$, on calcule la vraissemblance pondérée (*weighted likelihood*) associée à la $k^{ième}$ gaussienne.

$$\mathcal{L}_k(x) = \pi_k \mathcal{G}_{\mu_k, \Sigma_k}(x), \tag{3.15}$$

On définit ensuite

$$\mathcal{P}_k(x) = \frac{\mathcal{L}_k(x)}{\sum_{i=1}^{N} \mathcal{L}_i(x)}. \tag{3.16}$$

Ces fractions permettent de séparer la distribution X suivant les N gaussiennes. En conséquence, on appelle

$$|\mathcal{X}| = \sum_{x \in X} \sum_{k=1}^{N} \mathcal{P}_k(x) \tag{3.17}$$

la cardinalité de la distribution X.

L'étape M consiste maintenant à réestimer les paramètres. Les coefficients de pondération sont calculées de la façon suivante pour tout k,

$$\pi_k = \frac{\sum_{x \in X} \mathcal{P}_k(x)}{|\mathcal{X}|}. \tag{3.18}$$

De même, on obtient ensuite les vecteurs moyennes,

$$\mu_k = \frac{\sum_{x \in X} \mathcal{P}_k(x)\, x}{\sum_{x \in X} \mathcal{P}_k(x)}, \tag{3.19}$$

et enfin les matrices de covariance

$$\Sigma_k = \frac{1}{\sum_{x \in X} \mathcal{P}_k(x) \; - 1} \sum_{x \in X} \mathcal{P}_k(x) \, [(x - \mu_k)(x - \mu_k)^T]. \tag{3.20}$$

La figure 3.4 représente le résultat de l'algorithme pour une distribution théorique en 2 dimensions.

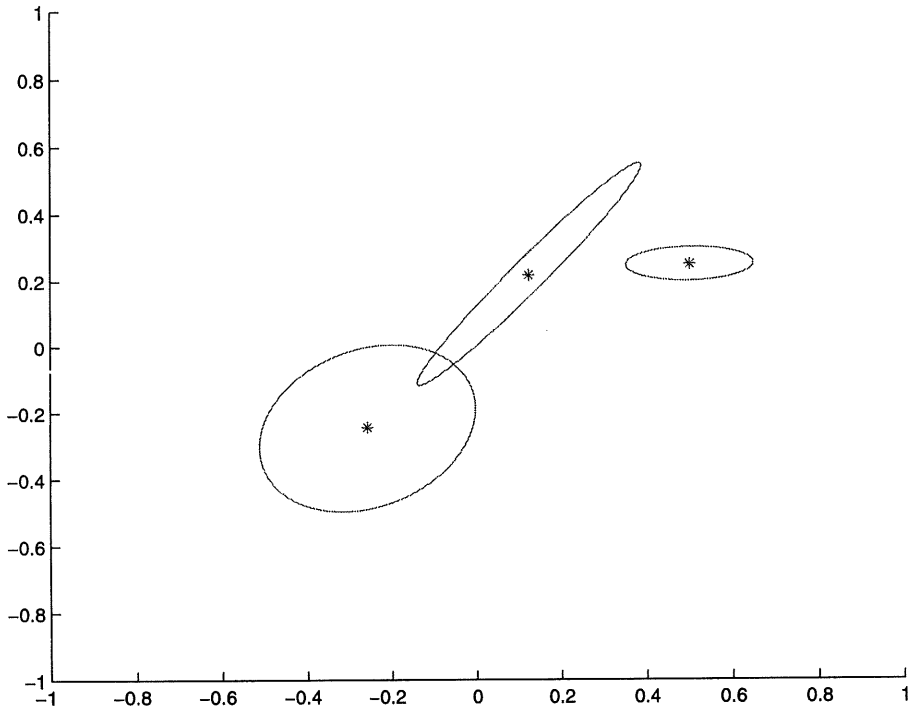

Figure 3.4 – Représentation des paramètres (μ, Σ) pour une modélisation d'une distribution en deux dimensions par un mélange de trois gaussiennes.

On peut montrer que cet algorithme converge vers un maximum de vraissemblance pour le mélange de gaussiennes, mais en pratique, l'étape d'initialisation est critique pour l'algorithme, et il peut converger vers un maximum local [Moo96]. Dans ce cas, la modélisation peut être très mauvaise.

La projection des données suivant les axes de la distribution peut permettre de se rendre compte de telles erreurs. Cela permet également d'évaluer le nombre N de gaussiennes nécessaires pour modéliser correctement la distribution, et de déduire des possibilités raisonnables pour l'initialisation des moyennes. L'algorithme EM peut devenir complexe lorsque le nombre d'échantillons à modéliser

est grand. Cependant, une fois les paramètres des gaussiennes estimées, la classification consiste seulement à calculer la probabitité conditionnelle pour chaque classe, ce qui est relativement peu coûteux.

3.3.2 K plus proches voisins

Il arrive parfois que les distributions associées aux classes de données soient difficiles à modéliser paramétriquement. On fait donc appel à des techniques différentes. On parle alors de classificateurs non-paramétriques. L'estimation non-parametrique est basée sur une estimation locale des densités, il en existe des différentes, estimée de Parzen [Fuk72]. Cependant, la méthode de classification non-paramétrique la plus utilisée est celle des K plus proches voisins (K-PPV), elle a la particularité de ne pas estimer les densités de probabilités, mais de les comparer, et est donc bien adaptée à un problème de classification en deux classes.

Dans notre cas, l'approche paramétrique semble adaptée, cependant les K-PPV permettent de comparer deux approches différentes. En outre, ils donnent une approximation de la probabilité d'erreur bayésienne. Nous allons d'abord présenter la règle des K ($K \geq 1$) plus proche voisin, pour ensuite considérer quelques résultats théoriques et jeter un regard critique sur cette méthode.

La règle des K plus proches voisins

Le principe des K plus proches voisins est extrêmement simple, puisqu'il consiste à chercher les K vecteurs de la base d'apprentissage les plus proches du vecteur à classifier, et d'attribuer à celui-ci la classe la plus représentée. La figure 3.5 illustre ce principe pour une distribution en deux dimensions et une distance euclidienne. Les K-PPV fournissent une estimation locale de la différence de densité entre les classes. En effet, si r est la distance entre l'échantillon X à classifier et le K^{ieme} plus proche voisin. On estime la densité de probabilité locale par

$$\hat{p}_n(X) = \frac{K-1}{N} \frac{1}{A(K,N,X)} \tag{3.21}$$

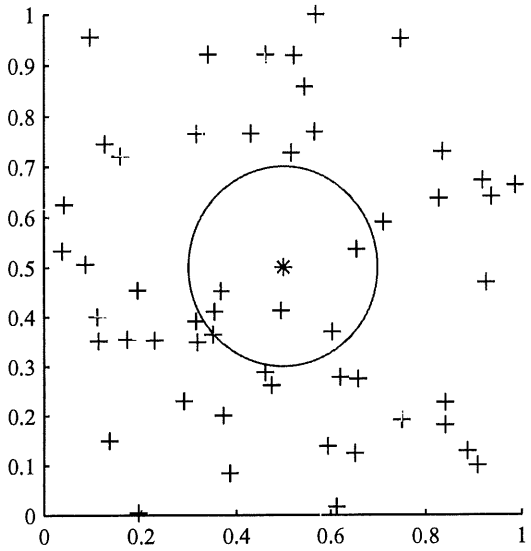

Figure 3.5 – Illustration du principe de la classification par K plus proches voisins pour un cas de deux classes.

où A, dans le cas d'une distance euclidienne, est une hypersphère de rayon r. A est une variable aléatoire dépendante des K échantillons sélectionnés et N est le nombre total d'éléments dans la distribution.

Dans une classification en deux classes. Les K échantillons contenus dans A consistent en K_1 échantillons de w_1 et K_2 échantillons de w_2. La densité de probabilité conditionnelle est estimée par

$$\hat{p}_{K_i}(X|w_i) = \frac{K_i - 1}{N_i}\frac{1}{A} \quad (i = 1, 2) \tag{3.22}$$

La décision bayésienne pour minimiser l'erreur devient alors

$$\frac{K_1}{N}\hat{p}_{K_1}(X|w_1) \gtrless \frac{K_2}{N}\hat{p}_{K_2}(X|w_2) \Rightarrow X \in \begin{cases} w_1 \\ w_2 \end{cases} \tag{3.23}$$

Étant donné que A est identique pour les deux classes, on en déduit la décision,

$$k_1 \gtrless k_2 \Rightarrow X \in \begin{cases} w_1 \\ w_2 \end{cases} \tag{3.24}$$

Probabilité d'erreur du 1-PPV et bornes sur la probabilité Bayésienne

Malgré l'apparence simpliste de la règle des K-plus proches voisins, celle-ci donne de bons résultats pratiques. En outre, il est possible de déterminer des bornes théoriques sur sa probabilité d'erreur. Nous ne démontrons ici que le calcul des bornes pour la règle du 1-PPV. Le calcul des bornes pour $K > 1$ est décrit dans [CH67]. Le résultat s'applique pour une distribution comportant un grand nombre d'échantillons.

x est l'élément à classifier, et x^k son plus proche voisin, appartenant à la classe w_k, parmi M. La vraie classe de x est supposée être w_p. La probabilité d'erreur est donc $P_{NN}(x) = P(w_p \neq w_k|x, x^k)$. Les évènements étant supposés indépendants, on en déduit que

$$P_{NN}(x) = \sum_{i=1}^{M} P(w_p = w_i|x)(1 - p(w_k = w_i|x^k)). \tag{3.25}$$

Finalement, si les données sont homogènes et en grand nombre, x^k sera très proche de x, et on peut raisonnablement supposer que pour tout i

$$P(w_k = w_i|x^k) \approx P(w_p = w_i|x) = P(w_i|x). \tag{3.26}$$

L'équation 3.25 devient alors

$$P_{NN}(x) = \sum_{i=1}^{M} P(w_i|x)(1 - p(w_i|x)) \tag{3.27a}$$

$$= 1 - \sum_{i=1}^{M} (P(w_i|x))^2. \tag{3.27b}$$

Il faut maintenant relier ce résultat avec la probabilité d'erreur Bayésienne $P_{Bayes}(x)$. On appelle w_{Bayes} la classe choisie par la décision Bayésienne, on a donc

$$P_{Bayes}(x) = \max_i P(w_i|x). \tag{3.28}$$

La probabilité d'erreur conditionnelle pour la règle de Bayes est donc

$$e_{Bayes}(x) = \sum_{i \neq Bayes}^{M} P(w_i|x) = 1 - P(w_{Bayes}|x). \tag{3.29}$$

Pour obtenir une limite sur l'équation 3.25, on écrit

$$\sum_{i=1}^{M}(P(w_i|x))^2 = (p(w_{Bayes}|x))^2 + \sum_{i\neq Bayes}^{M}(P(w_i|x))^2. \tag{3.30}$$

Pour une valeur fixe de $P(w_{Bayes}|x)$, on remarque que cette expression est minimisée lorsque toutes les probabilités restantes sont égales. On en déduit de l'équation 3.29 que ces valeurs sont pour tout $i \neq Bayes$

$$P(w_i|x) = \frac{e_{Bayes}(x)}{M-1}. \tag{3.31}$$

En remplaçant les résultats des équations 3.29 et 3.31 dans l'équation 3.30, on obtient la limite

$$\sum_{i=1}^{M}(P(w_i|x))^2 \geq (1-e_{Bayes}(x))^2 + \frac{e^2_{Bayes}(x)}{M-1}. \tag{3.32}$$

Enfin, on utilise cette inégalité dans l'équation 3.27b et par simplification, on a

$$e_{NN}(x) \leq 1-(1-e_{Bayes}(x))^2 - \frac{e^2_{Bayes}(x)}{M-1} \tag{3.33a}$$

$$e_{NN}(x) \leq 2e_{Bayes}(x) - \frac{M}{M-1}e^2_{Bayes}(x). \tag{3.33b}$$

Ce résultat s'étend à la probabilité d'erreur moyenne e_{Bayes} et e_{NN}

$$e_{Bayes} \leq e_{NN} \leq 2e_{Bayes} - \frac{M}{M-1}e^2_{Bayes}. \tag{3.34}$$

Cela montre que dans le cas d'une distribution importante, la probabilité d'erreur du plus proche voisin est toujours plus grande que la probabilité d'erreur Bayésienne, et toujours plus petite que le double de celle-ci.

Algorithmes de réduction de complexité de calcul

Les résultats théoriques pour les K-PPV sont très intéressants, mais le grand nombre de données nécessaires pour bien représenter les distributions posent des problèmes en terme de stockage et de complexité de calcul. Il faut en effet stocker tous les échantillons de la base d'apprentissage, et chercher à chaque étape les K éléments les plus proches du vecteur à classifier. Pour une base de n éléments, dans un espace de paramètres à d dimensions, cela fait $n * d$ distances scalaires à calculer.

Deux solutions se présentent pour palier à cet inconvénient. La première vise à regroupper les données qui ont des positions voisines et remplacer les groupes par des représentants (*clustering*), ce genre de technique est développé dans [Koh88]. La deuxième solution est de développer un algorithme de recherche rapide qui permet de limiter le calcul des distances à certains échantillons.

Il existe de nombreux algorithmes développés pour la recherche rapide. Celui présenté, est tiré de [aPMN75], il a l'avantage de ne pas exclure d'échantillons lors de la recherche (recherche exhaustive). Il est basé sur la recherche par arbre. Dans un premier temps, les données sont classées. La recherche se fait ensuite en descendant un arbre, et seules les données contenues dans les branches retenues sont utilisées pour le calcul des K-PPV.

L'algorithme de rangement des données est quelconque, la technique suggérée est l'algorithme de classification par k-moyenne, [GG92]. Le vecteur d'échantillons $(x_1, x_2, \ldots, x_n)$ est divisé en i sous-ensembles, chaque sous-ensemble est de nouveau divisé en i, et ainsi de suite. Cette décomposition peut-être représentée par un arbre tel que celui de la figure 3.6 ($i = 3$). Chaque noeud p est

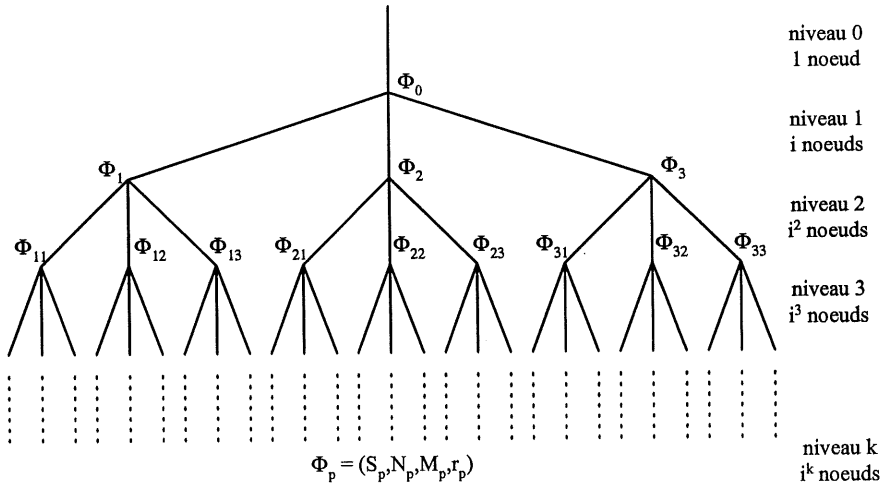

Figure 3.6 – Décomposition par arbre des données. Chaque noeud possède i branches et des paramètres utilisés pour la recherche.

caractérisé par les paramètres suivant :

S_p	ensemble de données associées à p,
N_p	nombre de données associées à p,
M_p	Vecteur moyenne de S_P,
r_p	$= \max_{X_i \in S_p} d(X_I, M_p)$, plus grande distance entre M_p et $X_i \in S_p$.

Une fois l'arbre élaboré, la recherche pour le 1-PPV se fait suivant la règle suivante à chaque noeud p de l'arbre pour savoir si x peut appartenir à S_p : Un échantillons x_i peut être le plus proche voisin de x si

$$B + r_p < d(x, M_p). \tag{3.35}$$

B est la distance du plus proche voisin courant de x parmi les échantillons considérés jusqu'alors, il est initialisé à ∞.

L'extension de l'algorithme pour la recherche des K-PPV est immédiate. L'inconvénient de l'algorithme est que le nombre de branches retenues finalement est variable, par conséquent le nombre de distances à calculer également. Dans le pire des cas, on peut même être amené à calculer toutes les distances. Le temps moyen de recherche est en revanche beaucoup plus faible que pour la recherche exhaustive parmi tous les éléments de la base.

3.3.3 Réseaux de neurones

Les réseaux de neurones ont été dans les années 1980 l'espoir de l'intelligence artificielle. Depuis, les limites de ces systèmes ont été mieux définies, et ils sont reconnus comme un outil performant de classification. Les ouvrages concernant les réseaux de neurones sont nombreux, le lecteur pourra se référer à [Hay94, Bis95] pour une introduction complète de ceux-ci. Cette partie a pour but de présenter deux types de réseaux utilisés en classification, le perceptron multi-couches, et le réseau RBF. L'accent est mis sur la description des algorithmes et des heuristiques susceptibles d'améliorer leurs performances, et sur le choix des paramètres des réseaux qui est la plus grande difficulté pour obtenir un système performant.

Le perceptron multi-couches

Le perceptron multi-couches décrit ici, et utilisé dans notre système comporte seulement une couche cachée, 3.7. La couche d'entrée $X = (x_1, \ldots, x_n, x_{n+1})$ est constituée des valeurs des n

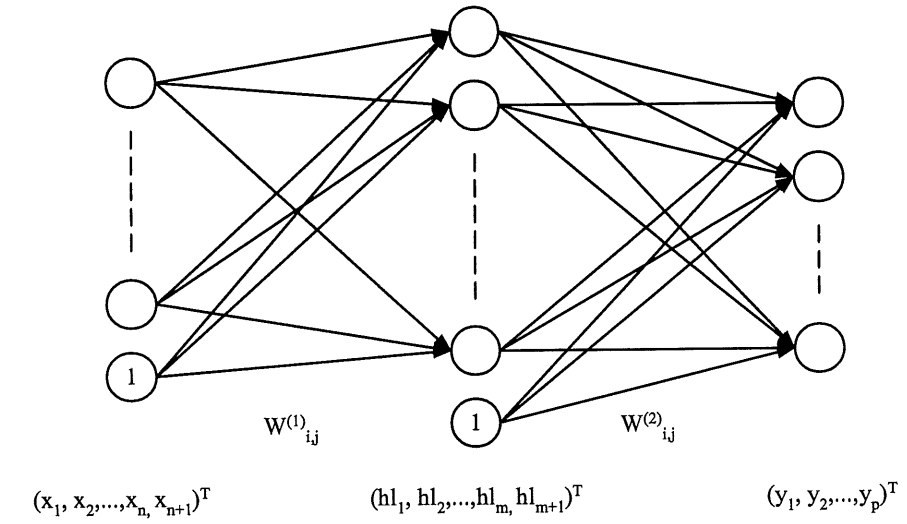

Figure 3.7 – Architecture d'un perceptron multi-couches à une couche cachée.

paramètres calculés au chapitre 2. $HL = (hl_1, \ldots, hl_m, hl_{m+1})$ est la couche cachée constituée de $m+1$ neurones. x_{n+1} et hl_{m+1} sont des biais, ils sont toujours égaux à 1. La couche de sortie est appelée $Y = (y_1, \ldots, d_p)$. Les matrices $W^{(1)} = (w^{(1)})_{n+1,m}$ et $W^{(2)} = (w^{(2)})_{m+1,p}$ sont les coefficients d'apprentissage. Ce sont eux qui vont être modifiés pour "apprendre" la sortie désirée en fonction des données entrées.

Algorithme de rétropropagation du gradient : Cet algorithme résulte de la minimisation d'une fonction d'erreur définie par

$$E = \sum_{i=1}^{p} (y_i - d_i)^2. \tag{3.36}$$

où $D = (d_i, \ldots, d_p)$ est la sortie désirée du réseau correspondant à X. Le problème consiste à trouver les erreurs commises par les neurones de sortie et les neurones cachées. Cela se fait par la méthode de descente du gradient. L'algorithme se fait en deux étapes, une première étape de propagation, où on calcule la sortie du réseau pour un X en entrée. La seconde consiste à calculer le gradient d'erreur pour les différentes couches.

La propagation se fait couche par couche. On calcule d'abord les sorties de la couche cachée pour $i \in (1, m)$:

$$hl_i = f(A_i^{(1)}) \; avec \; A_i^{(1)} = \sum_{j=1}^{n+1} w_{i,j}^{(1)} x_j, \tag{3.37}$$

ou

$$HL = f(W^{(1)}.X). \tag{3.38}$$

$f()$ est une fonction non-linéaire. On utilise généralement la fonction échelon, sigmoïde ou la fonction tanh. La sortie est calculée de la même façon pour $i \in (1, p)$:

$$y_i = f(A_i^{(2)}) \; avec \; A_i^{(2)} = \sum_{j=1}^{m+1} w_{i,j}^{(2)} hl_j, \tag{3.39}$$

ou

$$Y = f(W^{(2)}.HL). \tag{3.40}$$

Le gradient $\Delta^{(2)} = (\delta_i^{(2)}, \ldots, \delta_p^{(2)})$ de sortie est obtenu de la façon suivante :

$$\delta_i^{(2)} = 2.(y_i - d_i).f'(A_i^{(2)}), \tag{3.41}$$

On en déduit le gradient de la couche cachée $\Delta^{(1)} = (\delta_i^{(1)}, \ldots, \delta_p^{(1)})$

$$\delta_i^{(1)} = f'(A_i^{(1)}) \sum_{j=1}^{p} w_{j,i}^{(2)} \delta_j^{(2)}, \tag{3.42}$$

L'apprentissage se fait alors de la façon suivante,

$$\Delta w_{i,j}^{(1)} = \alpha \delta^{(1)} x_j, \;\; \forall i \in (1, m), j \in (1, n+1) \tag{3.43a}$$

$$\Delta w_{i,j}^{(2)} = \alpha \delta^{(2)} hl_j, \;\; \forall i \in (1, p), j \in (1, m+1) \tag{3.43b}$$

α est le taux d'apprentissage. Sa valeur est fixée empiriquement (souvent 0.1). L'algorithme se répète sur l'ensemble des échantillons X d'apprentissage jusqu'à ce que

- la fonction d'erreur soit inférieure à un seuil pré-établi,
- ou le gradient de la fonction d'erreur soit inférieur à un seuil pré-établi,
- ou un nombre prédéterminé d'itérations soit atteint.

L'algorithme de rétro-propagation du gradient est l'un des plus populaires dans le domaine des réseaux de neurones. Il n'est pourtant pas très efficace. Deux raisons principales expliquent cette lacune. D'une part, il est fortement dépendant du taux d'apprentissage α. À chaque étape d'amélioration dans la direction du gradient, les paramètres avancent trop ou pas assez car il est difficile de trouver les pas d'apprentissage optimaux. D'autre part, la direction du vecteur gradient ne pointe pas vers le minimum absolu de la surface d'erreur, on peut donc converger vers des minimas locaux.

Il existe plusieurs méthodes heuristiques pour accélerer la convergence de l'algorithme et éviter le problème des minimas locaux. Celles présentées ici concernent essentiellement le taux d'apprentissage.

Taux d'apprentissage individuel et variable : Cette méthode est basée sur deux idées. Si en deux étapes successives t et $t-1$, les variations des poids $\Delta_{i,j}^{(k)}(t)$ et $\Delta_{i,j}^{(k)}(t-1)$ ont un signe opposé, cela signifie que le poids $w_{i,j}^{(k)}$ oscille, et le taux d'apprentissage doit être diminué. Dans le cas contraire, le taux doit être augmenté. Ensuite, pour que l'évolution des poids soit compétitive, la somme des taux pour tous les poids doit être constante. Les points suivants permettent de résumer ces heuristiques :

- Chaque poids doit avoir son taux individuel d'apprentissage.
- Il faut autoriser la variation du taux d'apprentissage pour chaque dimension des poids.
- On augmente ou on diminue la variation des poids respectivement si $\Delta_{i,j}^{(k)}(t)$ et $\Delta_{i,j}^{(k)}(t-1)$ ont le même signe, ou un signe opposé.

Momentum : Pour éviter les oscillations des coefficients d'apprentissage, on introduit un terme d'inertie dans l'apprentissage. Il est proportionnel à la variation du gradient à l'itération

précédente. L'équation d'apprentissage devient alors

$$\Delta w_{i,j}^{(1)}(t) = \alpha\delta^{(1)}x_j + \gamma\Delta w_{i,j}^{(1)}(t-1) \ , \ \forall i \in (1,m), j \in (1,n+1) \tag{3.44a}$$

$$\Delta w_{i,j}^{(2)}(t) = \alpha\delta^{(2)}hl_j + \gamma\Delta w_{i,j}^{(2)}(t-1) \ , \ \forall i \in (1,p), j \in (1,m+1). \tag{3.44b}$$

$0 \leq \gamma < 1$ est un nouveau coefficient d'apprentissage. $\Delta w_{i,j}^{(1)}(t)$ est alors représentée comme une somme exponentiellement pondérée des gradients consécutifs. On peut donc voir que si ces gradients ont le même signe, la valeur absolue de la somme aura plutôt tendance à être augmentée. Si les gradients oscillent, elle sera diminuée.

Réseaux RBF

Les réseaux à base de fonctions radiales (*radial basis function*) sont des réseaux de deux couches dont la fonction réalisée peut être exprimée sous la forme

$$y_i(x) = \sum_{j=1}^{m} w_{i,j}\Phi_j(x) + w_{i,0}. \tag{3.45}$$

m est le nombre de neurones intermédiaires. La fonctions Φ est radiale, et paramétrable. Elle doit être fonction de $\|x - x_0\|^i$, $i \geq 1$, x_0 étant appelé le centre de la fonction. De plus, Φ doit vérifier $\lim_{\|x-x_0\|\to+\infty} \Phi = 0$. Les fonctions généralement utilisées sont

$$\Phi_j(x) = exp(-\frac{\|x-\mu_j\|^2}{2\sigma_j^2}), \tag{3.46a}$$

$$\Phi_j(x) = \frac{1}{(\|x-\mu_j\|^2 + \sigma_j^2)^2}. \tag{3.46b}$$

La démarche de l'algorithme d'apprentissage pour ces réseaux est la suivante :

- Détermination des paramètres des fonctions radiales Φ. Dans le cas des deux fonctions précédentes, on utilise une méthode basée sur la k-moyenne.
- Détermination des poids $w_{i,j}$ par des méthodes basées sur la correction d'erreur.

Les principales propriétés des réseaux RBF sont

- une capacité d'approximation d'une fonction continue quelconque,
- la vitesse d'apprentissage,

– la capacité de généralisation.

Ces réseaux sont par ailleurs très intéressants par les similitudes qu'ils ont avec les méthodes de classification statistiques présentées dans les sections précédentes, [Low99].

3.4 Résultats de discrimination

3.4.1 Base d'apprentissage et base de test

Le choix de la base d'apprentissage et de la base de test est important. Elles doivent être le plus exhaustives pour pouvoir évaluer correctement les performances des paramètres. La base d'apprentissage est utilisée pour l'élaboration du classificateur. La robustesse de la classification sera donc énormément dépendante de la diversité de celle-ci. Les séquences qui ont été utilisées pour construire notre base d'apprentissage sont les suivantes. Dans le cas de la musique, des séquences d'une dizaine de seconde ont été enregistrées à partir de disques numériques, et décimées à un taux d'échantillonnage de 16kHz. La plus grande variété possible a été utilisée, la musique pop, rock, techno, rap, classique, chantée ou non chantée. Les séquences de parole utilisées sont des locuteurs de sexe différents, et dans plus de 24 langues. La base d'apprentissage était environ constituée de 120000 trames de signal, ce qui correspond à 40mn de parole et de musique.

La base de test est également très importante. Elle a été choisie de façon à être aussi diversifiée que la base d'apprentissage, mais bien sûr avec des séquences différentes. Des test ont également été effectués pour observer le comportement de la classification lors de transitions parole/musique ou musique/parole. Ces résultats sont discutés dans la section 4.2.4. La base de test utilisée était composée d'environ 100000 trames de parole et de musique. Cela permet d'avoir une certaine confiance dans les statistiques qui sont présentées ensuite.

3.4.2 Limite bayésienne

La théorie de la classification bayésienne nous donne une borne inférieure pour la probabilité d'erreur. Celle-ci est démontrée dans la section 3.2.2, elle utilise la classification par 1-plus proche voisin. Cette limite nous dit que théoriquement, on ne pourra avoir une classification qui a une probabilité d'erreur inférieure à cette limite. Elle est vraie pour une base de données suffisamment grande. Il est raisonnable de penser que ce sera vrai pour les bases que nous utilisons. La limite obtenue est :

$$P_{Bayes} > 2.51 \tag{3.47}$$

3.4.3 Résultats de classification

Le tableau suivant présente les résultats de classification obtenus pour les différents classificateurs. Ils sont commentés ensuite.

Classificateur	Probabilité d'erreur (%)
Maximum de vraissemblance	5.93
5-GMM	3.28
10-GMM	3.17
20-GMM	3.24
1-PPV	4.92
3-PPV	3.17
Perceptron à 10 neurones cachés	8.27
Perceptron à 25 neurones cachés	8.12

Les résultats obtenus sont tous supérieurs à la limite Bayésienne, ce qui prouve que celle-ci semble correctement approchée. Les résultats obtenus par les mélanges de gaussiennes sont assez performants. On s'aperçoit que le résultat est moins bon avec 20 gaussiennes que 10. Ceci peut sembler étonnant, mais il faut se rappeler que la convergence est plus difficile pour un nombre de gaussiennes plus grand. Ce résultat est dû essentiellement à l'initialisation de l'algorithme EM.

Les résultats obtenus avec les K-plus proches voisins sont relativement bons. Nous avons constaté qu'augmenter K plus grand que 3 n'apportait pas d'intérêt aux résultats. Ce classificateur ne sera pas utilisé dans la classification pour des raisons de complexité.

Les résultats obtenus par le perceptron multi-couches n'atteint pas les résultats des autres classificateurs. Nous avons constaté qu'augmenter le nombre de neurones cachés au dessus de 25 n'augmentait pas les performances. En outre, nous nous sommes aperçus que la majorité des erreurs survenaient dans des segments de musique, bien que l'apprentissage ait été fait sur un nombre identique d'éléments de musique et de parole, et que ceux-ci aient été appris aléatoirement. Ces résultats sont sans doute les conséquences de l'inoptimalité de l'algorithme de rétro-propagation du gradient, d'un mauvais choix de la structure ou des vecteurs d'apprentissage. Il faut rappeler que le choix des paramètres est difficile pour un perceptron, et c'est sans doute l'inexpérience qui fait que les résultats obtenus pour le perceptron sont inférieurs aux autres classificateurs. Il serait intéressant de tenter d'améliorer les performances du perceptron en utilisant des algorithmes plus performants, ne convergeant pas vers un minimum relatif.

Généralement, les erreurs qui se produisent sont communes aux différents classificateur, c'est pourquoi l'amélioration du système réside plutôt dans les paramètres.

3.4.4 Hysteresis

La reconnaissance de formes statistiques est basée sur le fait que les vecteurs à classifier sont des processus aléatoires. Cependant, les vecteurs à classifier sont des statistiques à long-terme, réestimés toutes les trames. Ils possèdent donc une inertie qui fait qu'ils seront dépendants d'une décision à l'autre. Pour éviter des erreurs de classification, on propose une décision basée sur une hysteresis. Elle rend plus difficile le changement de décision du classificateur en pondérant la décision en faveur de la trame précédente. Cette pondération peut se comprendre comme l'introduction d'une probabilité *a priori* dans la classification statistique. Nous l'utilisons pour les réseaux de neurones et les mélanges de gaussiennes. Les résultats sont les suivants.

Classificateur	Probabilité d'erreur avec hysteresis (%)
Maximum de vraissemblance	4.44
5-GMM	2.30
10-GMM	2.17
20-GMM	2.11
Perceptron à 10 neurones cachés	4.03
Perceptron à 25 neurones cachés	4.03

Les performances en sont nettement améliorées. Cependant, ce genre de logique amène un problème puisqu'elle a tendance a retarder la décision, et pose donc des problèmes pendant les transitions. C'est pourquoi finalement, cette idée n'a pas été retenue dans l'intégration des codeurs.

Chapitre 4

Intégration dans un codage multimode

L'objectif du codage est de compresser des signaux à des fins de transmission ou de stockage. Les applications de l'audio à 16kbit/s sont la vidéo-téléphonie, la vidéoconférence, la diffusion sur Internet et dans un futur proche, les téléphones mobiles. L'information long-terme nécessaire pour obtenir des performances satisfaisantes pour la discrimination parole/musique nous empêche d'envisager une communication bi-directionnelle. L'application de codage pour ce système est la diffusion sur Internet, ou encore le stockage de données audio.

Dans un premier temps, les principes de base du codage de la parole et de l'audio sont décrits. Le système global intégrant le discriminateur et les codeurs est ensuite présenté. Les problèmes qui émergent de ce modèle sont discutés, et une solution utilisant un VAD est proposée pour obtenir des performances de codage satisfaisantes.

4.1 Techniques de codage de la parole et de l'audio

4.1.1 Codage de parole

Codage à débit élevé : Le codage différentiel

Généralités : Les techniques de codage à haut débit sont pratiquement toujours du type codage temporel parce qu'elles cherchent à conserver l'allure temporelle des signaux.

Auparavant, les transmissions numériques dans le réseau téléphonique utilisaient exclusivement la loi de codage PCM (Pulse Code Modulation), mais les besoins en transmission numérique ayant augmenté considérablement, il a fallu normaliser un second algorithme de réduction de débit pour la transmission dans le réseau téléphonique, le choix du CCITT (Comité Consultatif International pour la Téléphonie et la Télégraphie) s'est porté sur le codage ADPCM (Adaptative Differential PCM), pour lequel une qualité de codage satisfaisante a été obtenue grâce à la technique de codage différentiel en adaptant les prédicteurs en fonction des caractéristiques spectrales des signaux à coder et les quantificateurs en fonction de la dynamique du signal de différence.

Principe du codage différentiel : Le principe du codage différentiel consiste à quantifier non pas le signal lui-même $S(n)$ mais la différence $e(n)$ entre le signal et une prédiction $\tilde{S}(n)$ de sa valeur à partir des valeurs précédentes des signaux $\tilde{S}(n)$ et $e_q(n)$, [Mor95]. Le bruit de quantification sera d'autant plus faible que la différence à quantifier sera petite (signal très prédictible). Pour un rapport signal sur bruit, on peut donc diminuer le nombre de bits du quantificateur, et donc réduire le débit du codeur. Le quantificateur fait correspondre à chaque valeur de $e(n)$ le numéro de la plage dans lequel cette valeur apparaît. Le quantificateur inverse transforme ce numéro en un signal quantifié. Le prédicteur au décodeur calcule la prédiction $\tilde{S}(n)$ de l'échantillon $S(n)$ à partir des échantillons précédents du signal reconstitué $\tilde{S}(n-1)$, $\tilde{S}(n-2)$, ..., et du signal d'erreur quantifié $e_q(n-1)$, ...,.

La quantification adaptative : Compte tenu de la non-stationnarité du signal de parole, le signal de différence $e(n)$ à l'entrée du quantificateur présente des variations importantes en dynamique qu'un quantificateur à seuils fixes ne saurait prendre en compte. L'adaptation du quantificateur consiste à rendre les seuils dépendants du niveau estimé du signal de différence. Étant donné que le quantificateur inverse doit être adapté aussi bien côté émission que réception, l'estimation du niveau à l'instant n doit se faire à partir du code binaire transmis.

La prédiction adaptative : Le rapport signal sur bruit d'un codeur différentiel étant égal à la somme du rapport signal sur bruit du quantificateur et du gain de prédiction, il faut s'efforcer de modéliser au mieux le gain de parole pour maximiser le gain de prédiction. Le modèle le plus général du signal de parole est le modèle auto-régressif à moyenne ajustée (ARMA). La prédiction adaptative consiste à modifier les valeurs des coefficients du filtre ARMA à chaque instant d'échantillonnage par un algorithme du gradient.

Codage à débit réduit : Le vocodeur

Généralités : Les procédés de codage temporel (PCM, ADPCM) permettent de préserver fidèlement l'évolution temporelle du signal de parole. Cependant, la qualité excellente de ces codeurs est obtenue au prix d'un débit élevé. En effet, la diminution du débit nécessite la réduction du nombre de niveaux du quantificateur. Aux environs de 16 kbit/s, l'intelligibilité reste acceptable, mais pour des débits plus faibles, leur bruit de quantification crée une gêne trop importante pour envisager leur utilisation. On a donc recours à d'autres techniques de codage, une solution possible consiste à modéliser l'appareil de production de la parole tout entier. Cependant, l'appareil phonatoire est tellement complexe que ces modélisations sont encore impropres à des applications concrètes.

La solution utilisée dans les vocodeurs est plus raisonnable, elle consiste à modéliser la source vocale et l'incidence des diverses parties de l'appareil phonatoire sur le signal acoustique qui se propage dans celui-ci.

Principe des vocodeurs : Pour parvenir à un débit inférieur à 5 kbit/s les vocodeurs font appel à un modèle simplifié de la phonation en ne retenant que les paramètres les plus significatifs du signal de parole (codage paramétrique).

La partie synthèse d'un vocodeur restitue le spectre à court terme du signal : Le signal d'excitation à spectre plat est mis en forme par un filtre ou un banc de filtres de synthèse qui reproduit les résonances, ou formants du conduit vocal. La partie analyse consiste à déterminer le signal d'excitation du filtre et ses coefficients.

La source vocale peut être soit voisée (caractérisée par la période de vibration des cordes vocales), ou non voisée. Sa modélisation dans les vocodeurs consiste donc à détecter le voisement. S'il est présent, on remplace la source par un signal à spectre plat et de même périodicité, sinon, elle est modélisée par un bruit blanc.

Codage à débit moyen

Pour conserver un codage de bonne qualité pour un débit inférieur à 16 kbit/s, les techniques de codage à débit moyen utilisent de façon complémentaire les avantages des techniques temporelles qui ne modélisent pas de façon excessive la source vocale, et ceux des techniques de codage paramétriques qui ont la particularité de coder efficacement l'enveloppe spectrale. De ces derniers, ils reprennent la modélisation de l'ensemble ou d'une partie du spectre sur des tranches de signal de 10 à 20 ms, pendant lesquelles les caractéristiques spectrales du signal évoluent lentement. Les techniques employées sont à base de prédiction linéaire ou de filtrage pas banc de filtres. La modélisation de la source des vocodeurs est elle avantageusement remplacée par le codage temporel représentant de façon plus précise la source ou du moins permettant d'en reconstituer les principales caractéristiques.

Le codage CELP (Code Excited Linear Prediction) est à la base du codage à débit moyen en parole. Il n'est pas décrit dans cette partie puisque le principe du codage ACELP (Algebraic Code Excited Linear Prediction) utilisé dans notre système est présenté dans la section 4.2.1.

Bilan des codeurs de parole

Nous avons présenté les différents codeurs utilisés en codage de la parole. Nous n'avons pas abordé les techniqes de codage par tranformée qui sont traités dans la section suivante et qui concernent le domaine audio. La figure 4.1 présente les performances des codeurs en fonction de leur débit, elle est tirée de [Cal89]. Elle résume parfaitement la qualité obtenue par le codage selon

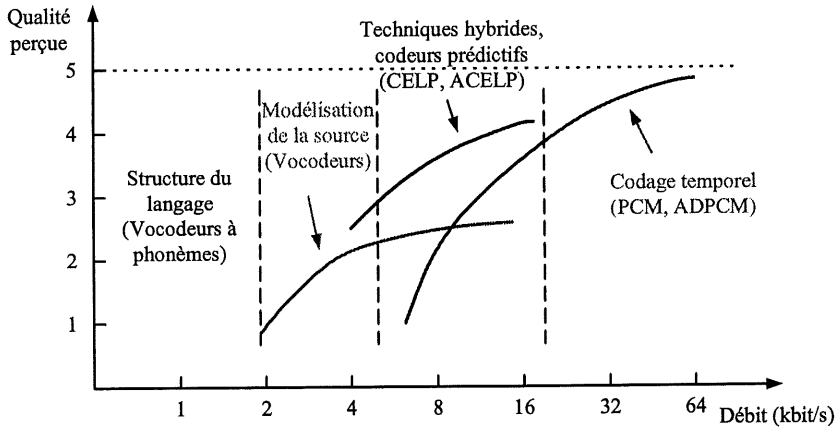

Figure 4.1 – Performances des différents codeurs en termes de notes d'opinion en fonction de leur débit.

la technique utilisée.

4.1.2 Codage audio

Le codage de l'audio à haut débit est basé sur la prédiction adaptative, de même que pour la parole, le lecteur peut donc se référer à la section 4.1.1. Pour un codage à un débit plus faible ou pour une largeur de bande plus élevée (MPEG, MP3), les techniques utilisent le domaine fréquentiel qui permet d'intégrer les propriétés psychoacoustiques de l'oreille. Elles sont brièvement présentées dans la partie suivante.

Notion de masquage

Le masquage auditif décrit le fait qu'un signal d'amplitude faible (signal masqué) devient inaudible lorsqu'un signal plus fort (masqueur) est émis simultanément. Ce phénomène peut être exploité en codage par une mise en forme de bruit (*noise shaping*) appropriée. Le masquage dépend de la distribution fréquentielle du signal masqueur et masqué, ainsi que leur distribution temporelle. Le masquage temporel n'a jusqu'à présent pas été exploité dans le codage. Seules les propriétés du masquage fréquentiel ont été utilisées.

Ce phénomène se passe lorsque le signal masqué et le signal masqueur sont suffisamment proches en fréquence. Le seuil d'audition absolu est le niveau minimal qui doit être atteint par un signal pour être audible. Quand l'environnement n'est pas silencieux, ce seuil est modifié, on obtient alors le seuil de masquage. La figure 4.2 présente un seuil de masquage pour un bruit à bande étroite (90 Hz), de fréquence centrale de 1 KHz. Il dépend de la puissance du signal masqueur et de sa fréquence. Il correspond à un masquage d'un bruit à bande étroite. Le masquage fréquentiel est très

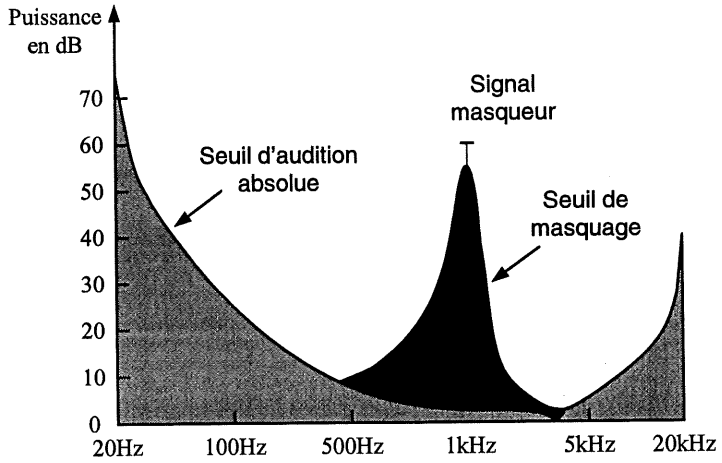

Figure 4.2 – Seuil d'audition absolu et seuil de masquage pour un son à bande étroite. Les sons à bande étroite dont la puissance ne dépasse pas les zones grisées sont inaudibles.

intéressant pour le codage, le signal dans le seuil de masquage peut être du bruit de quantification

ou être émis par la source. Dans ce dernier cas, il n'a pas besoin d'être codé et transmis. Un codage de source efficace essaiera de supprimer tous les éléments du signal qui ne sont pas perceptibles par l'oreille.

Mise en forme du bruit et codage perceptuel

Des techniques de mise en forme dynamique du bruit permettent d'augmenter le bruit de codage dans les bandes fréquentielles qui n'ont pas d'importance perceptuellement, [Pai92]. L'allocation fixe ou dynamique de bits dans les sous-bandes du domaine fréquentiel offrent le moyen le plus simple de prendre en compte les propriétés du système auditif. La figure 4.3 décrit le principe d'un codeur perceptuel. L'encodeur est controlé par un bloc perceptuel basé sur une analyse fréquentielle qui

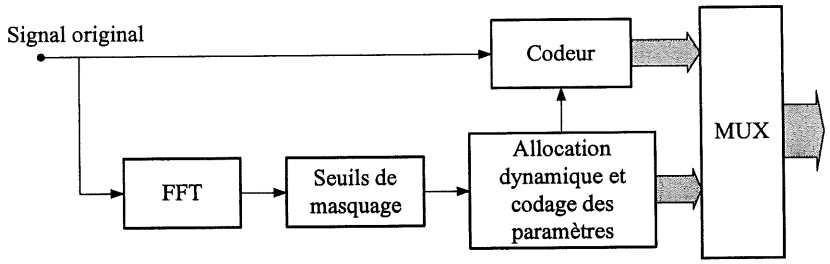

Figure 4.3 – Schéma bloc d'un codeur perceptuel.

met en forme le bruit par une courbe de pondération de coefficients de la tranformée. Ce principe peut être adapté à toutes sortes de codeurs. Si le nombre de bits nécessaire pour représenter la courbe de masquage est disponible, le codage sera alors transparent, c'est à dire que la différence entre le signal mis en forme et le signal source sera inaudible.

Codage dans le domaine fréquentiel

Le codage fréquentiel offre un moyen direct pour la mise en forme du bruit et la suppression des composantes inaudibles. Le spectre du signal est décomposé en différentes bandes qui sont codées

séparément. Par conséquent, le bruit de quantification associé à chaque bande est contenu dans celle-ci. Deux techniques existent, le codage en sous-bandes, et le codage par transformée. Dans les deux cas, le codeur est basé sur une analyse par banc de filtres pour produire des composantes spectrales sous-échantillonnées.

Dans le codage en sous-bandes, le signal est introduit dans un banc de M filtres en sous-bandes. Chaque sortie est ensuite décimée par un facteur M. Les échantillons ainsi obtenus sont ensuite quantifiés. Au décodeur, le taux d'échantillonnage est rétabli pour chaque sous-bande en introduisant un nombre approprié de zéros (interpolation), et les signaux sont ensuite traités par le filtre de synthèse. Dans le cas de filtres à reconstruction parfaite, et en absence de quantification, la somme des sous-bandes permet de retrouver exactement le signal original. Dans le cas opposé, le recouvrement spectral entre filtres voisins provoque une distorsion fréquentielle (*aliasing*)

Dans le codage par tranformée, un bloc de N échantillons est traité par une tranformation discrète pour produire N nouveaux échantillons dans le domaine fréquentiel. Les tranformations typiquement utilisées sont la DFT (*Discrete Fourrier Transform*) ou la DCT (*Discrete Cosinus Transform*). Les coefficients obtenus sont ensuite quantifiés. Le décodage consiste simplement à appliquer la transformée inverse. De même que pour le codage en sous-bandes, la tranformation introduit des erreurs dans le signal appelées effets de bords. Ces effets peuvent être réduits en utilisant un recouvrement entre les différentes trames codées, au prix d'un nombre de bits plus grand.

4.2 Détails du système

L'élaboration du système global est simple puisqu'on a des modules qui sont indépendants entre eux. Le schéma bloc du système est présenté à la figure 4.4. Le choix des codeurs est commandé par le système de discrimination parole/musique (ou encore parole/non-parole). La fiabilité de la discrimination n'étant pas totale, il est nécessaire de modifier le système pour prévenir d'éventuelles erreurs. Dans un premier temps, nous allons décrire d'une manière plus détaillée les codeurs qui

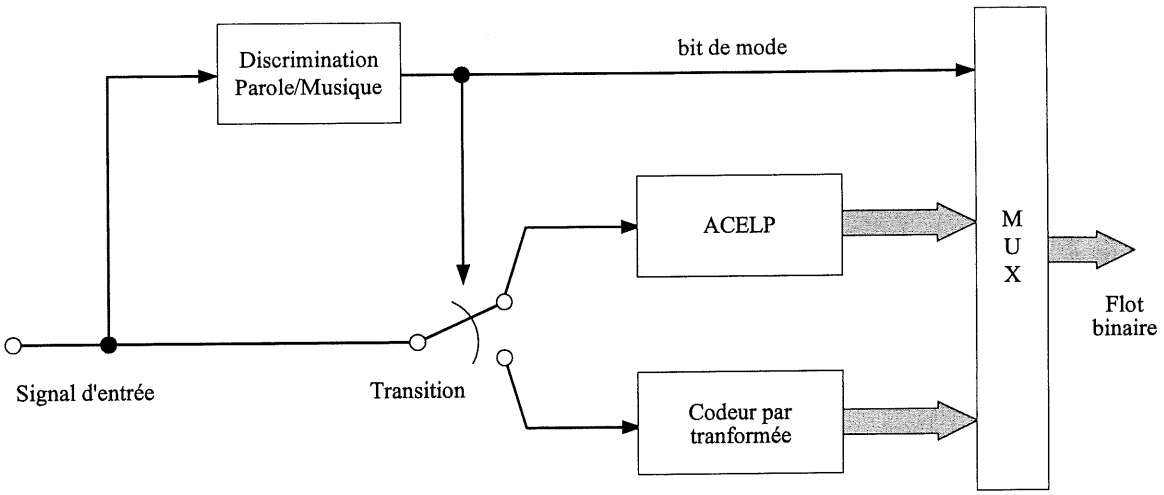

Figure 4.4 – Schéma-bloc du système global

ont été utilisés dans notre système, puis décrire les influences d'une mauvaise transition sur la qualité de l'audio. Il restera alors à expliciter les solutions qui permettent de résoudre les erreurs qui surviennent lors du codage, et enfin, décrire le système global qui en découle.

4.2.1 Codeur ACELP

Le codage ACELP (Algebraic Code Excited Linear Prediction) est basé sur le codage CELP qui a été proposé par [SA85]. La bonne qualité de codage à bas débit est obtenue pas analyse par synthèse utilisant à la fois la prédiction à court terme et à long terme comme montré à la figure 4.5. Ce codeur permet d'obtenir un codage bas débit aussi bien pour des signaux échantillonnés à 8 kHz ou 16 KHz. La procédure d'analyse consiste à trouver la séquence du dictionnaire optimale respectivement à un critère d'erreur subjectif. Chaque mot de code c_k est multiplié par un facteur de gain G_k et filtré par les filtres $1/B(z)$ (prédicteur de pitch) et $1/A(z)$ (filtre prédictif linéaire inverse). La différence $x_n = s_n - \hat{s}_n$ entre le signal et le signal synthétisé est filtré par le filtre perceptuel $W(z)$, et la meilleure séquence est choisie de façon à minimiser l'énergie du signal d'erreur perceptuelle y_n.

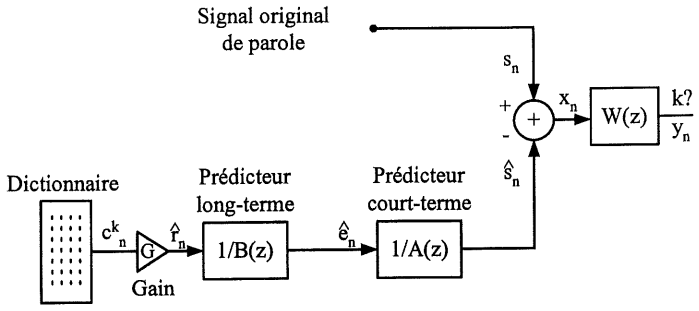

Figure 4.5 – Schéma-bloc du codage CELP.

Originalement, le signal d'excitation utilisé dans l'analyse par synthèse est issu d'un grand dictionnaire stochastique, pondéré par un facteur gain. Le meilleur vecteur d'excitation choisi est celui qui minimise l'erreur quadratique moyenne de la différence entre l'original et la synthèse. La recherche est exhaustive, donc la complexité est importante.

Pour remédier à ce problème, il a été proposé dans [AMDM87] d'utiliser les propriétés des codes algébriques pour générer l'excitation. Un vecteur d'excitation est représenté par $c_k = Fa_k$. le dictionnaire algébrique $\{a_k\}$ est composé de vecteurs $a_0, a_1, \ldots, a_{L-1}$, et la matrice F est dépendante de la prédiction linéaire. Son rôle est de mettre en forme le vecteur d'excitation dans le domaine fréquentiel pour concentrer l'énergie dans les bandes de fréquences importantes.

Un vecteur algébrique a_k est de la forme

$$a_k = \sum_{i=0}^{p-1} b_i \delta(k - m_i) \tag{4.1}$$

p est le nombe d'impulsions, b_i sont les impulsions (1 pour i paire, et -1 pour i impaire), les m_i sont les positions des impulsions. Les mots de code de ce dictionnaire peuvent être représentés par une distribution uniforme de points sur une hyper-sphère. C'est cette représentation qui rend le codage algébrique économique en nombre de bits. Une recherche sélective des pulses permet d'obtenir une complexité réduite, tout en gardant des performances très proches d'une recherche exhaustive [LAMM90].

4.2.2 Codeur G.722.1

Cette partie présente le principe du codeur G.722.1 développé par la compagnie PictureTel pour le codage large bande de l'audio de 50Hz à 7kHz à 24kbit/s et 32kbit/s, et détaillé par une recommandation de l'ITU-T (*International Telecommunication Union*) [1696].

L'algorithme est basé sur une transformée, la MLT (*Modulated Lap Transform*), opérant sur des trames de 20ms (320 échantillons). Le recouvrement est utilisé sur la trame suivante, le délai total du codeur est donc de 40ms (trame courante + recouvrement). Chaque trame est codée indépendamment, le débit est de 480 et 640 bits par trame pour des débits de 24kbit/s et 32kbit/s respectivement.

La figure 4.6 représente le bloc diagramme de l'encodeur. La tranformée donne 320 coefficients MLT. Ils sont d'abord appliqués à un bloc qui calcule l'enveloppe de la tranformée et la quantifie. La tranformée est divisée en blocs de 20 coefficients MLT appelés régions. Chaque région représente une largeur de bande de 500Hz. L'enveloppe est obtenue en calculant le RMS (*Root Mean Square*) de chaque région. Les bits représentant l'enveloppe transmise au multiplexeur, les bits restants sont utilisés pour le bloc de catégorisation.

La procédure de catégorisation utilise l'amplitude quantifiée et le nombre de bits restants dans la trame pour générer 16 ensembles de catégorisations. Chacune nécessite un nombre différent de bits pour encoder les mêmes coefficients MLT. Chaque catégorisation consiste en un ensemble de 14 allocations de catégories, pour chacune des régions de la tranformée. Chaque catégorie représente des paramètres de codage et de quantification pour chaque région. Il leur est associé le nombre de bits nécessaires pour le codage d'une région. Le nombre total de bits utilisés pour ce codage est variable puisqu'il est basé sur le codage de Huffman.

Ensuite, les coefficients MLT sont quantifiés et codés différemment pour chacune des catégorisations. Le nombre de bits nécessaires est déterminé à chaque fois. Les coefficients sont d'abord normalisés par l'amplitude de l'enveloppe quantifiée puis quantifiés scalairement. Les indices scalaires obtenus sont combinés en vecteurs d'indices, et codés par Huffman. La catégorisation sélectionnée est celle

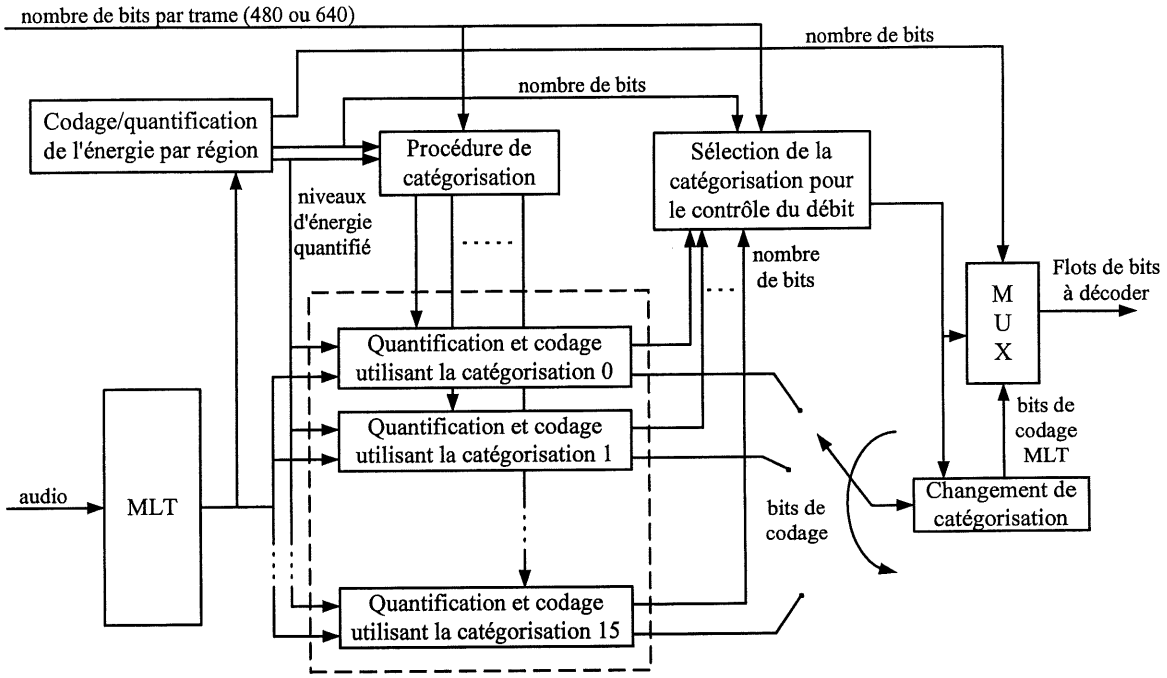

Figure 4.6 – Schéma-bloc du codeur G.722.1

qui a le nombre de bits le plus proche du nombre de bits restants pour le canal. 4 bits de contrôle de catégorisation identifient la catégorisation choisie au décodeur.

Pour notre application, le débit souhaité étant 16kbit/s, le nombre de bits alloués à chaque trame est 320. Le reste du codeur est identique.

4.2.3 Différences de codage entre l'ACELP et le G.722.1

L'illustration des performances de codage entre le codeur ACELP et le G.722.1 permet de démontrer l'intérêt d'un système utilisant les deux techniques. Nous nous attachons à montrer les différences qui sont générales aux codeurs de parole et de musique, et non aux codeurs qui ont été choisis en particulier.

Les codeurs de musique ont le défaut de ne pas coder efficacement les signaux de parole voisée. Cela est du au fait qu'ils n'ont pas de prédicteur de pitch. La figure 4.7 illustre ce phénomène en comparant la transformée de Fourier de la synthèse du codeur ACELP, et du G.722.1 avec celle de l'original pour 40ms de parole voisée masculine. On observe que la structure harmonique est mieux

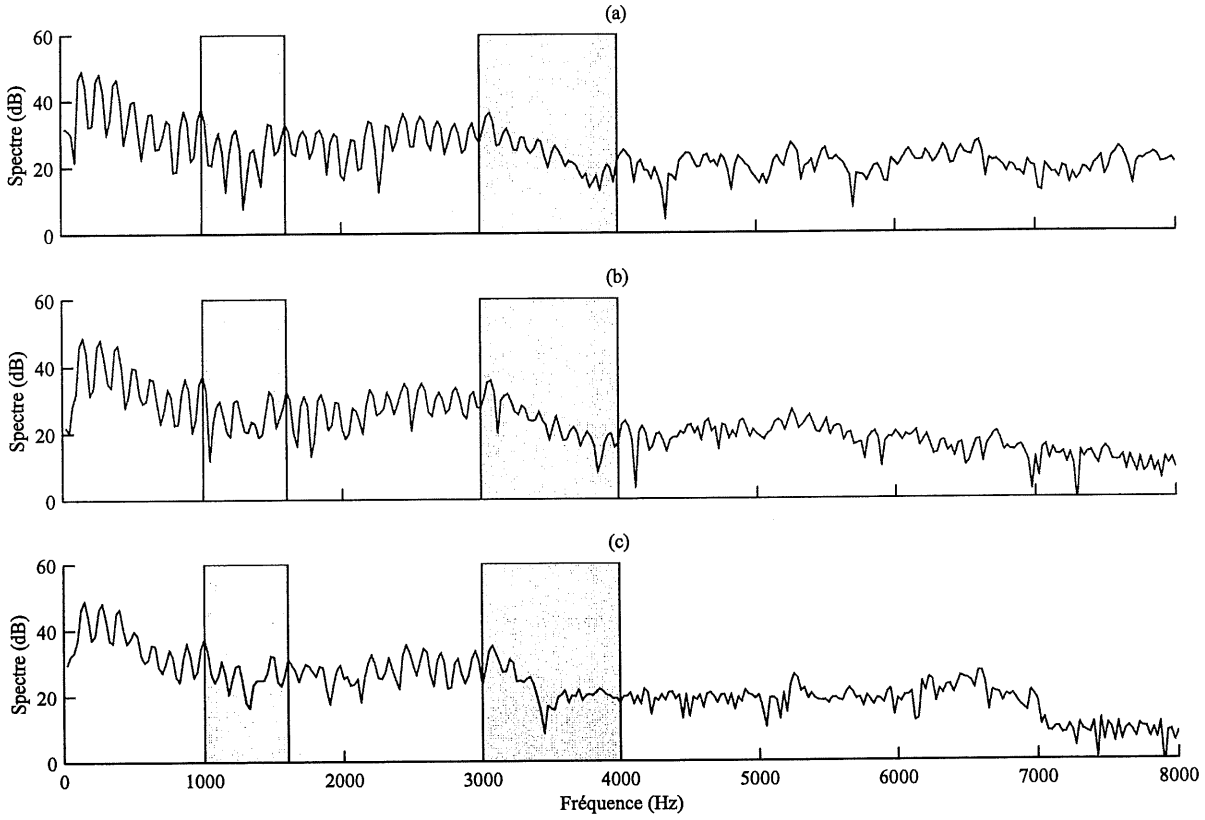

Figure 4.7 – Transformée de Fourier de 40ms de parole voisée. (a) Signal original. (b) Signal codé avec l'ACELP. (c) Signal codé avec le G.722.1.

conservée pour le codeur de parole. De plus, le fait que le locuteur soit masculin oblige le G.722.1 à allouer beaucoup de bits dans les basses fréquences, d'où une reproduction très mauvaise des hautes fréquences (il ne faut pas considérer les fréquences dépassant 7kHz, puisque le signal est préfiltré).

Dans le cas d'un signal musical ayant une forte structure harmonique, et complexe, le codeur

G.722.1 va réussir à mieux représenter le signal dans le domaine fréquentiel. La figure 4.8 nous montre la tranformée de Fourier pour le signal original, le signal codé par l'ACELP, et le signal codé par le G.722.1 de 40ms d'harmonica. La particularité du signal original est d'avoir des raies

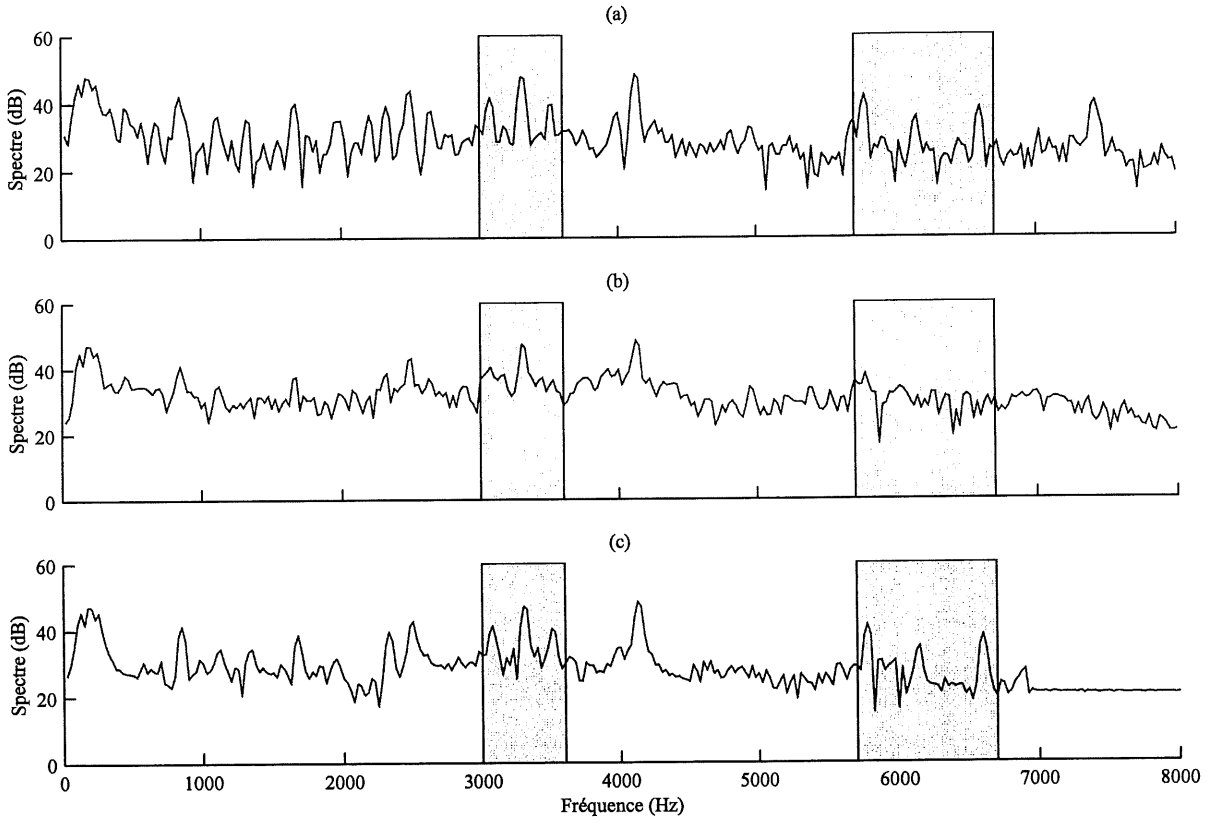

Figure 4.8 – Transformée de Fourier de 40ms de musique. (a) Signal original. (b) Signal codé avec l'ACELP. (c) Signal codé avec le G.722.1.

très énergétiques dans les hautes fréquences. L'ACELP est incapable de les reproduire. Le résultat perceptuel qui en résulte est très mauvais. En ce qui concerne le G.722.1, il réparti ses bits de façon à coder les raies sur toute la bande. Il réussi donc à représenter les raies à hautes fréquences, et à obtenir une qualité perceptuelle acceptable, nettement meilleure que l'ACELP.

4.2.4 Comportement des codeurs dans une transition

Cette section décrit les comportements des codeurs ACELP et G.722.7 pour des transitions, c'est à dire lorsque le classificateur indique un changement de nature du signal. On peut raisonnablement penser qu'ils seront reproductibles pour d'autres codeurs. Le but d'une décision en boucle ouverte étant de proposer un système adaptable à tout codeur, les solutions qui seront envisagées n'entreront pas dans la structure même des codeurs. C'est donc une politique très différente d'une décision en boucle fermée proposée dans [BSLL99] ou d'une approche hybride proposée par [Ram99] qui utilise un mode parole, un mode musique, et un mode transitionnel. Lors d'une transition, les codeurs sont simplement initialisés, comme si on les redémarrait.

Transition parole/musique

La transition parole/musique met en cause l'initialisation du codeur G.722.1. On sait que celui-ci utilise un recouvrement sur la trame précédente. Cette information n'étant pas disponible au décodage puisque la trame précédente aura été codée par le codeur de parole, l'initialisation revient à considérer que la trame précédente était nulle. La figure 4.9 représente les deux mêmes séquences audio. La première (a) a été obtenue en codant la première trame par l'ACELP, et les suivantes par le G.722.1 en initialisant ce dernier. La seconde (b) a été obtenue en codant toute la séquence par le G.722.1. L'effet de la mise à zéro de la trame précédente a beaucoup d'impact sur le signal codé. Il faut une trame au signal pour que celui-ci retrouve son allure normale. Cet artefact est dû au recouvrement entre la trame courante et la trame précédente. Il est très perceptible sur le plan auditif, et même gênant. Il n'y aucun moyen direct de prévenir ce comportement sans avoir recours à un mode transitoire.

Transition musique/parole

L'initialisation du codeur ACELP pour la transition musique/parole est plus complexe que pour le codeur G.722.1. On n'utilise pas de recouvrement avec la trame précédente, en revanche certains

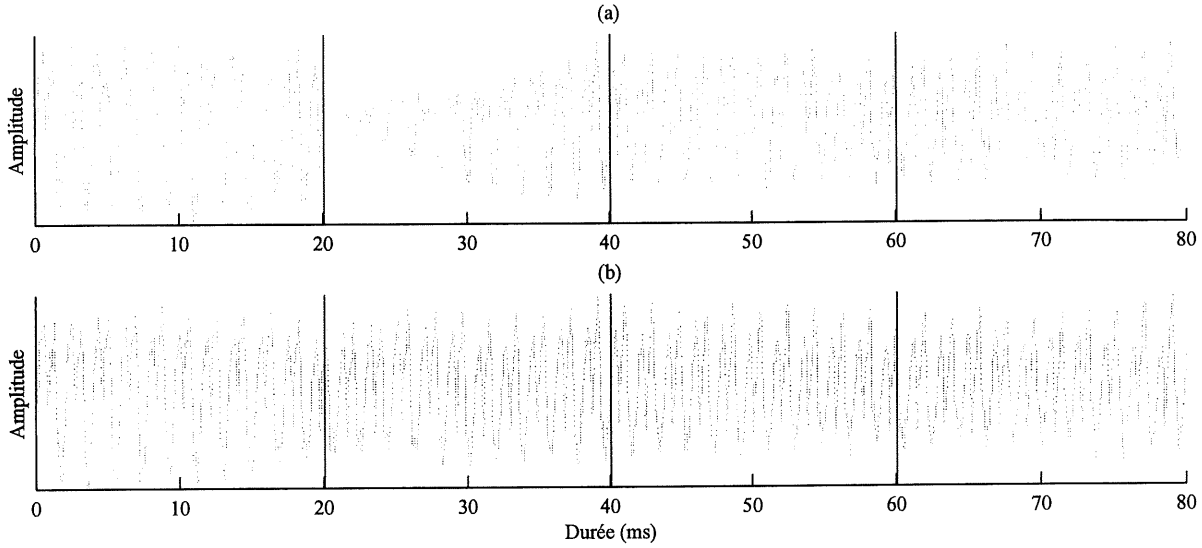

Figure 4.9 – Illustration du comportement du codeur G.772.1 pendant une transition. (a)Séquence audio de 80ms avec transition ACELP-G.772.1. (b)Même séquence codée entièrement avec le G.772.1.

paramètres transmis sont calculés par quantification vectorielle algébrique prédictive (paramètres représentant la prédiction linéaire). Ces paramètres sont réinitialisés en supposant une enveloppe fréquentielle plate. De même, l'excitation précédente, ainsi que les mémoires de filtres sont mis à zéro. La figure 4.10 représente les deux mêmes séquences audio. La première (a) a été obtenue en codant la première trame par le G.722.1, et les suivantes par l'ACELP en initialisant ce dernier. La seconde (b) a été obtenue en codant toute la séquence par le codeur ACELP. Cette fois ci encore, les résultats de l'initialisations sont très mauvais pour le signal codé. L'effet est moins long puisque certains paramètres sont estimés sur des sous-trames, en revanche, l'initialisation des mémoires de filtre amplifie le signal au moment de la transition. Leur réponse est similaire à celle d'un filtre devant un signal échelon. En effet, le signal passe d'une énergie nulle (mémoire des filtres mise à zéro) à un signal d'énergie non-nulle. Les conséquences auditives sont encore plus mauvaises que pour le codeur G.722.1 puisque dans ce cas on entend des craquements très gênants. Cette fois encore, il n'y a pas de solution directe à ce problème sinon d'utiliser un mode transitoire.

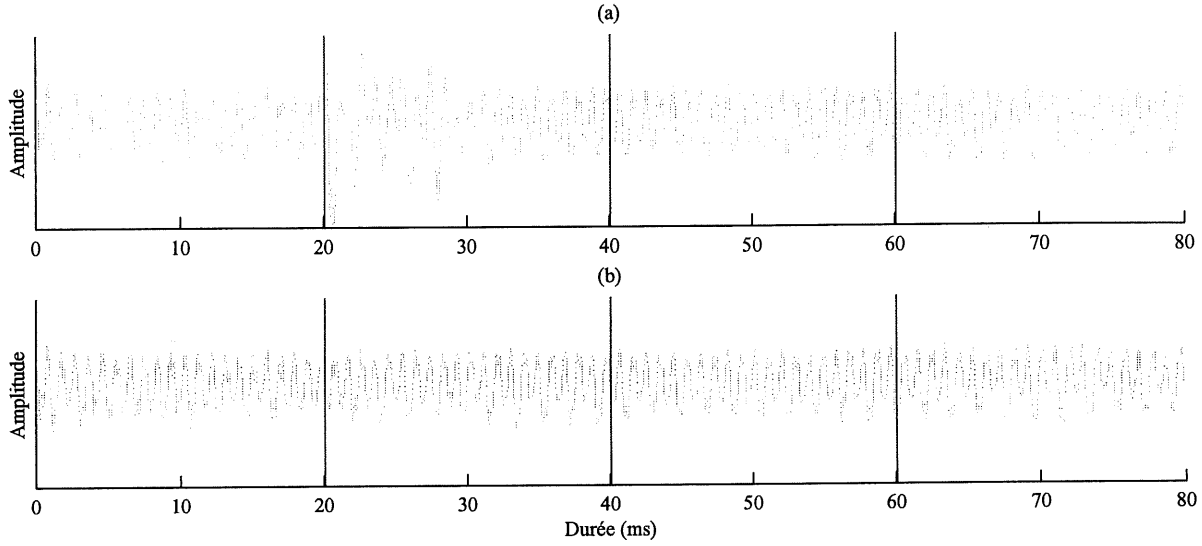

Figure 4.10 – Illustration du comportement du codeur G.772.1 pendant une transition. (a)Séquence audio de 80ms avec transition G.772.1-ACELP. (b)Même séquence codée entièrement avec le codeur ACELP.

Conclusion

Les problèmes soulevés par la transition entre les codeurs sont incompatibles avec les résultats obtenus par le discriminateur. Seuls des performances de 100% de classification permettraient d'occulter ces problèmes d'artefact. Cependant, même s'il est sans doute possible de gagner quelques dixièmes de pourcent sur les performances du discriminateur, il est inenvisageable d'obtenir une classification parfaite. Il faut donc trouver une parade qui permette de contourner ces artefacts de codage.

Les transitions des codeurs deviennent imperceptibles si elles ont lieu dans des segments de signal qui ont une faible énergie. C'est donc ce qui est exploité pour compléter le système. Les transitions entre codeurs ne sont autorisées que lorsque l'énergie du signal est suffisamment faible. Le critère choisi doit être indépendant du niveau du signal et du niveau de bruit.

4.3 Système final

L'introduction d'un critère énergétique pour autoriser les transitions entre codeurs modifie quelque peu le principe de la discrimination. Le schéma bloc du classificateur est décrit à la figure 4.11. L'utilisation de la valeur de l'enveloppe énergétique calculée pendant l'estimation des paramètres a été utilisée. Ce critère a l'avantage d'être normalisé, donc indépendant du niveau du signal, et adaptatif. Cependant, il ne donne pas d'information sur la nature du signal, son voisement, sa stationnarité. Les performances et les défauts inhérents à ce genre de décision sont décrits dans la section 4.3.2. Une autre solution possible est d'utiliser un détecteur d'activité vocale (*VAD*). Un

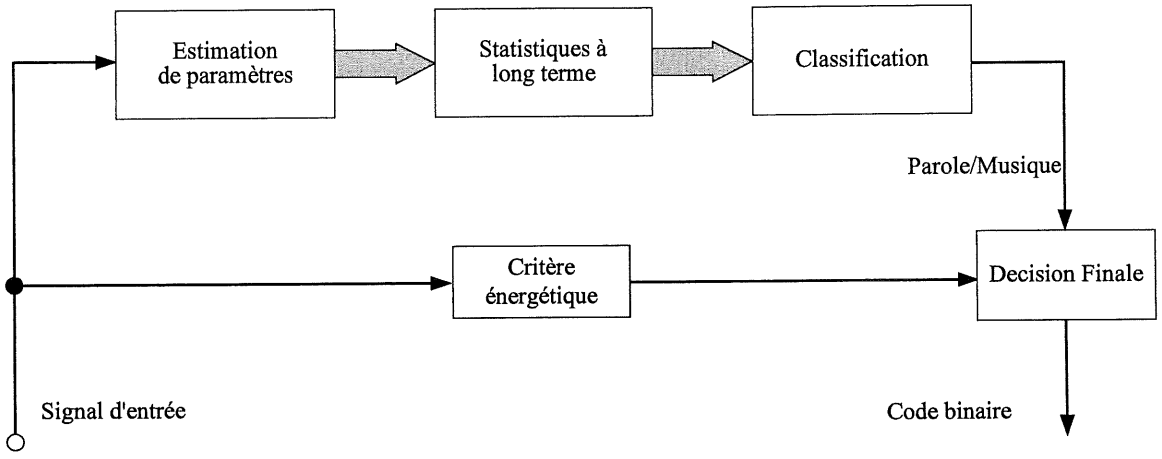

Figure 4.11 – Schéma-bloc du système de discrimination parole/musique

exemple de *VAD* est décrit dans la section suivante. Il a l'avantage de donner une décision basée à la fois sur l'énergie, le voisement et la stationnarité du signal. Cependant, l'utilisation d'un tel système comme critère de transition entrainerait une augmentation importante de la complexité du système.

4.3.1 Détection d'activité vocale

Le détecteur d'activité vocale (VAD) utilisé dans notre système est détaillé dans [JL00]. La description haut niveau du VAD est présentée à la figure 4.12. Des paramètres de voisement sont extraits pour faire une première décision locale qui sert seulement à controler la mise à jour de l'estimation du niveau de bruit. La décision finale de voisement est finalement basée sur le rapport signal sur bruit (SNR) entre le signal et l'estimée du niveau de bruit. Cette approche a l'avantage d'être plus robuste qu'une décision locale seule.

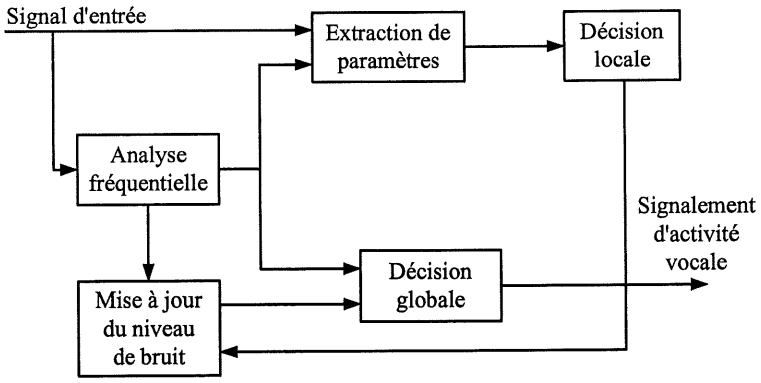

Figure 4.12 – Schéma-bloc du VAD

Dans un premier temps, une analyse fréquentielle en 9 sous-bandes est effectuée toutes les 20ms. Le degré de stationnarité, le SNR et le niveau de bruit sont estimés sur chacune d'elle. Pour ces deux derniers, un moyennage est effectué avec la trame précédente pour améliorer leur fiabilité.

L'adaptation du niveau de bruit repose sur les six paramètres suivants : la stabilité du pitch, la corrélation normalisée pour deux sous-trames de 10ms, une mesure de non-stationnarité, un paramètre basé sur l'ordre d'un modèle auto-régressif suffisant pour modéliser le bruit, et les rapports d'énergie dans les plus hautes et les plus basses fréquences. La décision locale est basée sur ces paramètres. Si elle indique l'absence de signal utile pour plusieurs trames consécutives, l'inactivité

vocale est signalée en sortie du VAD, et le niveau de bruit est mise à jour dans chaque bande fréquentielle pour la trame suivante.

4.3.2 Performances

Les problèmes d'artefacts sont résolus en utilisant un critère énergétique. Les performances sur le plan perceptuel sont alors meilleures que lorsqu'on utilise un seul codeur. Cependant, cette solution soulève deux nouveaux problèmes qui sont en quelque sorte la limite du système tel qu'il a été envisagé.

Le premier problème se pose lorsque le discriminateur parole/musique se trompe au début d'un signal de musique. Cela peut arriver lorsque les caractéristiques de ce signal sont proches d'un signal de parole. Dans ce cas, même si le discriminateur rétablit son erreur, le mode utilisé va rester en parole, puisqu'on empêche toute transition dans des séquences à haut niveau d'énergie. La transition ne pourra avoir lieu que si le critère d'énergie passe à un moment donné sous le seuil qui a été défini. La solution à ce problème est d'utiliser un retard (*lookahead*) plus important sur le signal. Cela permettra d'avoir une meilleure connaissance de la nature du signal, et donc une meilleure anticipation sur les changements de celui-ci.

Le second problème est plus difficile à résoudre. Il arrive lorsque la nature du signal change, sans que l'on passe en dessous d'un certain seuil énergétique. C'est le cas lorsqu'on a un locuteur avec une musique en fond sonore, avec un fondu enchainé lorsque le locuteur se tait et la musique devient plus forte. Il n'existe pas vraiment de solutions, c'est une conséquence directe de la façon dont on a abordé le problème du codage par une discrimination.

Chapitre 5

Conclusion

La combinaison du codage de parole et audio par une discrimination parole/musique permet d'obtenir une qualité supérieure à celle d'un codeur individuel. La complexité résultante est inférieure à celle d'une décision en boucle fermée. Le délai nécessaire empêche d'appliquer le système à une communication bi-directionnelle, mais peut en revanche être utilisé dans une application de type diffusion telle que la radio-diffusion sur Internet.

L'analyse long-terme des signaux permet d'extraire des paramètres qui sont ensuite classifiés par une technique de reconnaissance de formes. La technique la plus efficace du point de vue complexité-performances est la classification par mélange de gaussiennes. Les résultats obtenus pourraient encore être améliorés en utilisant une base d'apprentissage plus importante.

La transition entre les codeurs est limitée par un critère d'énergie qui permet d'éviter des artefacts de codage inhérents au système tel qu'il a été pensé. Cette limitation ne garantit pas un codage optimal, c'est à dire que tous les signaux de musique codés par une codeur de musique, et tous les signaux de parole codé par un codeur de parole. Cela est dû au fait que la discrimination n'est pas parfaite, et surtout qu'il peut arriver que les transitions entre la parole et la musique ne soient pas nettes (fondue enchaînée). Ce phénomène est une limite de notre système par rapport à un modèle qui intègrerait complètement les deux modèles de codage [TRL00, TRRL00].

En revanche, il présente certains aspects pratiques notamment pour la diffusion sur Internet. Les contraintes que nous nous sommes imposées sur l'intégration des codeurs permet une souplesse sur l'utilisation de ceux-ci. Les codeurs que nous avons utilisé étaient prévus pour fournir un débit de 16 Kbit/s. Cependant, on pourrait imaginer d'intégrer une batterie de codeurs de parole, et une batterie de codeurs audio qui fonctionneraient à des débits différents, commandés par la capacité du canal ou la qualité exigée. C'est cette souplesse de codage qui fait que notre système est très intéressant pour la diffusion audio-numérique.

BIBLIOGRAPHIE

[1696] ITU Telecommunication Standardization Sector Study Group 16. Detailed description of the ptc (picturetel transform coder), October 1996.

[AMDM87] J-P. Adoul, P. Mabilleau, M. Delprat, and S. Morissette. Fast celp coding based on algebraic codes. *Proceedings of the International Conference on Acoustics Speech and Signal Processing*, pages 1957–1960, April 1987.

[aPMN75] K. Fukunaga anf P. M. Narendra. A branch and bound algorithm for computing k-nearest neighbors. *IEEE Transactions on Computers*, 24 :750–753, 1975.

[Bis95] C. M. Bishop. *Neural Networks for Pattern Recognition.* Clarendon Press, 1995.

[BSLL99] B. Bessette, R. Salami, C. Laflamme, and R. Lefebvre. A wideband speech and audio codec at 16/24/32 kbit/s using hybrid ACELP/TCX techniques. *IEEE Workshop on Speech Coding, Porvoo (Finland)*, June 1999.

[Cal89] Calliope. *La parole et son traitement automatique.* Collection technique et scientifique des télécommunications. Masson, 1989.

[CH67] T. M. Cover and P. E. Hart. Nearest neighbor pattern classification. *IEEE Transactions on Information Theory*, 13(1) :21–27, 1967.

[CPLT99] M. J. Carey, E. S. Parris, and H. Lloyd-Thomas. A comparison of features for speech, music discrimination. *Proceedings of the International Conference on Acoustics Speech and Signal Processing*, pages 149–152, 1999.

[Fuk72] K. Fukunaga. *Introduction to Statistical Pattern Recognition.* Electrical Science series. Academic Press, 1972.

[GG92] A. Gersho and R. M. Gray. *Vector Quantization and signal compression.* Kluwer Academic Publishers, 1992.

[Hay94] S. S. Haykin. *Neural Networks : A Comprehensive Foundation.* Maxwell Macmillan International, 1994.

[Hes83] W. Hess. *Pitch Determination of Speech Signals - Algorithms and Devices*, volume 3 of *Information Sciences.* Springer-Verlag, 1983.

[JL00] M. Jelinek and F. Labonté. Robust signal/noise discrimination for wideband speech and audio coding. *IEEE Workshop on Speech Coding, Delevan, Wisconsin, U.S.A.*, 2000.

[Ked86] B. Kedem. Spectral analysis and discrimination by zero-crossings. *Proceedings of the Institute of Electrical and Electronics Engineers*, 74(11) :1477–1493, November 1986.

[Koh88] T. Kohonen. *Self-Organisation Maps.* Series in Information Sciences. Springer-Verlag, 1988.

[KP95] W.B. Kleijn and K.K. Paliwal. *Speech coding and synthesis.* Elsevier, 1995.

[LAMM90] C. Laflamme, J-P. Adoul, S. Morissette, and P. Mabilleau. 16 kbps wideband speech coding technique based on algebraic celp. *Proceedings of the International Conference on Acoustics Speech and Signal Processing*, pages 177–180, April 1990.

[Low99] D. Lowe. *Statistics and Neural Networks.* Traité des nouvelles technologies. Oxford, 1999.

[Moo96] Todd K. Moon. The expectation-maximation algorithm. *IEEE Signal Processing Magazine*, pages 47–70, 1996.

[Mor95] N. Moreau. *Techniques de compression des signaux.* Collection technique et scientifique des télécommunications. Masson, 1995.

[Ols52] H. F. Olson. *Musical Engineering*, chapter Properties of music. McGraw-Hill, 1952.

[Pai92] Bruno Paillard. *Codage perceptuel des signaux audio de haute qualité.* PhD thesis, Université de Sherbrooke, 1992.

[Pat72] E. A. Patrick. *Fundamentals of Pattern Recognition.* Electrical Engineering series. Prentice-Hall, 1972.

[Ram99] S. Ramprashad. A multimode transform predictive coder (MPTC) for speech and audio. *IEEE Workshop on Speech Coding, Porvoo (Finland)*, June 1999.

[SA85] M.R. Schroeder and B.S. Atal. Code-excited linear prediction (celp) : high quality speech at very low bit rates. *Proceedings of the International Conference on Acoustics Speech and Signal Processing*, March 1985.

[Sau96] J. Saunders. Real time discrimination of broadcast speech/music. *Proceedings of the International Conference on Acoustics Speech and Signal Processing*, pages 993–996, 1996.

[SS97] E. Scheirer and M. Slaney. Construction and evaluation of a robust multifeature speech/music discriminator. *Proceedings of the International Conference on Acoustics Speech and Signal Processing*, pages 1331–1334, 1997.

[SZ96] M. S. Spina and V. W. Zue. Automatic transcription of general audio data : Preliminary analyses. *Proceedings International Conference on Spoken Language Processing*, pages 594–597, 1996.

[TRL00] L. Tancerel, S. Ragot, and R. Lefebvre. Speech/music discrimination for universal audio coding. *Proceedings of the* 20^{th} *Biennal Symposium on Communications*, May 2000.

[TRRL00] L. Tancerel, S. Ragot, V. T. Ruoppila, and R. Lefebvre. Combined speech and audio coding by discrimination. *IEEE Workshop on Speech Coding*, September 2000.