Reconnaissance et classification de phonèmes

Julien RACHEDI
Mémoire pour le Master Sciences et Technologie de l' UPMC
Spécialité SAR
Parcours ATIAM

Laboratoire d'accueil : IRCAM, Paris

Equipe : Applications Temps Réel
Responsables : Norbert Schnell et Diemo Schwarz

Mars / Août 2005

Table des matières

Table des figures

Liste des tableaux

Remerciements

Je remercie M. le Directeur Bernard Stiegler, de m'avoir permis d'effectuer mon stage de Master II, ainsi que ce Master II, au sein de l'institut IRCAM.

Je remercie Norbert Schnell de m'avoir acceuilli au sein de son équipe, l'équipe Applications Temps Réel (ATR). Les réunions auxquelles nous avons participé m'ont donné des directions de travail claires et stimulantes.

Je remercie vivement Diemo Schwarz, chargé de recherche et développement au sein de l'équipe ATR, qui m'a encadré tout au long du stage. Sa disponibilité, son aide et ses idées, et surtout sa capacité à répondre à toutes mes questions m'ont énormément apporté.

Je remercie également l'ensemble de l'équipe ATR, mes collègues stagiaires, mais aussi Laurent Ghys, Patrice Tisserand et Remy Muller pour leur passion du partage des connaissances. Je remercie toutes celles et ceux qui m'ont prêté leur voix lors des enregistrements de la base de données.

Enfin, je remercie les Ouiches Lorènes pour m'avoir permis de relâcher la pression en musique durant les week-ends, et dédie ce travail à Christelle.

Introduction

Le traitement de la parole est un domaine de recherche actif, au croisement du traitement du signal numérique et du traitement symbolique du langage. Depuis les années 60, il bénéficie d' efforts de recherche très importants, liés au développement des moyens et techniques de télécommunications. Reconnaître la parole, vecteur d'information privilégié de notre société, constitue donc un défi à relever pour permettre le passage à la prochaine génération d'interfaces homme-machine.

Dans le domaine artistique, outre les systèmes purement "signal" comme les voice-coders largement répandus dans la musique populaire, certaines applications intéressent particulierement les compositeurs, metteurs en scène et artistes de toutes disciplines. Les possibilités d'interactions en temps réel entre spectateur, acteur, chanteur, musicien et machine leur confèrent une inspiration nouvelle et inédite. Des demandes plus pragmatiques émanent également des ingénieurs du son ou régisseurs lumière des milieux du spectacle vivant, qui souhaiteraient une assistance pour synchroniser la parole au déclenchement d'événements sonores, visuels ou autres.

Après avoir développé des systèmes de suivi en temps réel de parole ou de partition de musique basés sur des Modèles de Markov Cachés (HMM), Norbert Schnell, responsable de l'équipe Applications Temps Réel de l'Ircam, et Diemo Schwarz, chargé de recherche, souhaitaient préciser temporellement leur système de reconnaissance à l'échelle d'une fenêtre d'analyse d'une trentaine de millisecondes de signal. C'est dans cette direction que se sont orienté mes recherches, dans le cadre d'un stage de Master ATIAM de l'Université Paris VI.

Dans le but d'implémenter un système de reconnaissance de phonèmes et temps réel, l'idée de ce stage est de pouvoir, pour chaque fenêtre d'analyse du signal, prendre une décision de classification et retourner le phonème prononcé. Cette idée se base sur l'hypothèse que les phonèmes sont des états stationnaires, spectralement reconnaissables et dissociables. Il faut donc trouver les paramètres décrivant de la manière la plus pertinente le signal, de façon non contextuelle. Ainsi, tous les phonèmes perceptivement reconnaissables par leur déroulement temporel, typiquement les plosives ou occlusives, ne peuvent rentrer dans ce cadre de travail, dans la mesure ou ils contiennent eux même une succession d'états stationnaires. Le travail de recherche à effectuer dans ce stage consistait donc à trouver les limites de la reconnaissance de phonèmes sans emploi de modèles temporels.

Ce document résume les différentes études réalisées. Après un bref aperçu du contexte dans lequel se place le sujet de ce stage, le concept du dispositif est décrit en détail dans les chapitres 2 et 3. Ce concept est basé sur l'emploi de techniques statistiques de classification sur une base de données de paramètres acoustiques de phonèmes. La principale difficulté de cette étude a été de

trouver des paramètres permettant une description stable des phonèmes, tâche délicate face à la grande variabilité de la parole. Une solution trouvée a été de constituer une base de données robuste donnant une représentation séparable des differentes classes de phonèmes, décrite au chapitre 4.

Chapitre 1

Etat de l'art

Ce chapitre donne un bref aperçu des théories et recherches antérieures sur lesquelles s'appuient les travaux effectués pendant ce stage. Avant de commencer mes recherches, une étude bibliographique a été réalisée afin de bien cerner et maîtriser les differents sujets à aborder. Nous introduirons d'abord la notion de phonèmes. Par la suite sera décrit le fonctionnement des reconnaisseurs de parole standard, basés sur le suivi temporel d'événements stationnaires à l'aide de Modèles de Markov Cachés.

1.1 Reconnaissance

L'apellation "reconnaissance de la parole" (ASR pour Automatic Speech Recognition en anglais) se réfère à plusieurs types de systèmes dont la mission est de décoder l'information portée par le signal vocal. Selon l'information à extraire, on distingue deux types de reconnaissance :

- La reconnaissance du locuteur, dont le but est de reconnaître la personne qui parle parmis une population de locuteurs (identificateur) ou de vérifier son identité (vérificateur). On sépare les cas de reconnaissance dépendante du texte, avec texte dicté ou indépendante du texte.
- La reconnaissance de parole, dont le but est de transcrire l'information symbolique exprimée par le locuteur. On distingue les cas de reconnaisseur monolocuteur, multilocuteur ou indépendant du locuteur. Une distinction est également faite entre reconnaisseur de mots isolés, de mots connectés et de parole continue. Nous souhaitons dans ce stage nous placer dans le cas d'un reconnaisseur de parole continue indépendant du locuteur.

Ces tâches demandent la contribution d'outils techniques aussi divers que puissants : traitement du signal, modèles mathématiques statistiques, algorithmique,... De plus, les cas d'applications sont toujours plus complexes que la théorie, dans la mesure ou ils introduisent des bruits exterieurs, la contribution du materiel utilisé, les differences de locutions...

1.2 Le phonème

Deux sciences étudient les sons du langage, la phonétique et la phonologie. Il s'agit de deux approches differentes du même phénomène, deux types de descriptions. La phonétique s'interesse à leur production, transmission, perception, alors que la phonologie étudie la manière dont ces sons participent au fonctionnement du langage et en assurent le codage [CY96].

Le concept de phonème est issu de l'hypothèse phonologique selon laquelle les sons du langage forment un code de l'information linguistique. Ils doivent être différentiables entre eux. Par définition, le phonème est "la plus petite unité phonique fonctionelle, i.e. distinctive"[BBD+99]. Lorsque deux modes de production différents entraînent un changement de sens du mot prononcé on peut leur donner le caractère de phonèmes ; par exemple si on remplace le [p] de *pain* par un [b]. En revanche, si la prononciation d'un [r] roulé contre un [r] grasseyé ne change pas le codage associé, on puis dire qu'ils sont phonétiquement différents mais phonologiquement semblables.[CAL89] Il en découle également des notions comme le diphone ou le triphone utilisés pour la synthese de parole par concaténation.

Le codage des phonèmes utilisés dans ce travail est la liste "XSAMPA" (pour Extended Speech Assessment Methods Phonetic Alphabet), qui permet une transcription en caractères ASCII [Wel95]. Ceux du français [Wel03] sont présentés dans le tableau ci-après.

Les phonèmes sont regroupés par classes selon leurs modes de production ; le rapport bruit de friction, voisement, l'utilisation de la cavité nasale ou l'occlusion permettant l'éclat des plosives sont autant de descripteurs articulatoires donnant cette classification :

Les consonnes

- "p b t d k g" sont les plosives, qui peuvent être elles-même séparées en 2 sous-classes : les plosives voisées (b, d et g) et les plosives non-voisées (p, t et k). Ces dernières se caractérisent par un silence puis un bruit. Si cette classe est aisément reconnaissable à l'aide d'un modèle avec suivi temporel, elle n'est pas reductible à une enveloppe spectrale stable.
- "f v s z S Z j" sont les six fricatives. "v z Z" sont les fricatives voisées, les autres sont non-voisées,
- "m n J N" sont les nasales. Bien qu'elles nécessitent un voisement, les nasales [m] et [n] présentent une plosion. Pour les deux dernières, il est très délicat d'en repérer le début et la fin, et elles sont phonétiquement assimilables à des concaténations serrées de deux consonnes,
- "l R w H" sont les liquides. De même, pour les deux dernières il est très délicat d'en repérer le début et la fin, et elles sont phonétiquement assimilables à des concaténations serrées de deux voyelles,

Les voyelles.

- "i e E a A O o u y 2 9 @" sont les douzes voyelles "orales".
- "e∼ a∼ o∼ 9∼" sont les voyelles nasales.

Notation	Exemple	Transcription XSAMPA
p	pont	po∼
b	bon	bo∼
t	temps	ta∼
d	dans	da∼
k	quand	ka∼
g	gant	ga∼
f	femme	fam
v	vent	va∼
s	sans	sa∼
z	zone	zon
S	champ	Sa∼
Z	gens	Za∼
j	ion	jo∼
m	mont	mo∼
n	nom	no∼
J	oignon	oJo∼
N	camping	ka∼piN
l	long	lo∼
R	rond	Ro∼
w	coin	kwe∼
H	juin	ZHe∼
i	si	si
e	ses	se
E	seize	sEz
a	patte	pat
A	pâte	pAt
O	comme	kOm
o	gros	gRo
u	doux	du
y	du	dy
2	deux	d2
9	neuf	n9f
@	justement	Zyst@ma∼
e∼	vin	ve∼
a∼	cent	va∼
o∼	bon	bo∼
9∼	brun	bR9∼

TAB. 1.1 – Liste XSAMPA des phonèmes du français

1.3 Modèle de production

Comme pour tous les instruments à vent les systemes de description de la voix se basent sur le modèle de production acoustique source-filtre.

Le signal de parole est provoqué par des mécanismes complexes issus de plusieurs sources. Lorsque les cordes vocales entrent en vibration périodique harmonique, le son est voisé et une hauteur est déterminée. Du bruit peut être provoqué par le souffle de friction. D'autres sources, impulsives, sont également en cause dans le cas des plosives, comme des claquements de langue ou l'ouverture rapide des lèvres après accumulation d'air dans la bouche, etc...

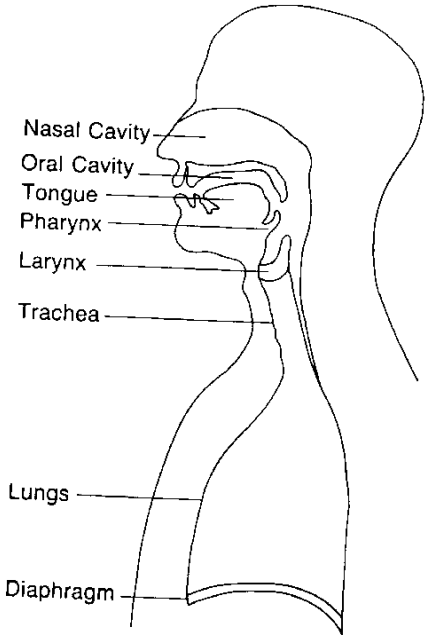

FIG. 1.1 – Anatomie de l'appareil phonatoire

Le conduit vocal, long de 17 cm en moyenne chez l'homme adulte, se compose de plusieurs cavités : pharynx, cavité orale, fosses nasales forment autant de résonances (formants) et d'anti résonances (anti-formants). Si ce corps n'agit pas sur la fréquence fondamentale il est en revanche extrêmement déformable, rallongeable, ou divisible. Agissant comme un filtre acoustique, il donne ainsi au son les différentes couleurs qui distinguent les voyelles ou les fricatives entre elles.[DDR95]

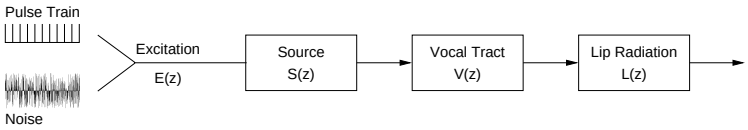

FIG. 1.2 – Modele de production de la parole

La source donne donc au son hauteur ou bruit et puissance, tandis que le conduit vocal agit comme un filtre acoustique. L'information phonétique est contenue dans le type de source utilisée et dans l'enveloppe spectrale du filtre. Il faut donc séparer les contributions de chacun dans le signal pour la décrire efficacement.

1.4 Les Modèles de Markov cachés

Les Modèles de Markov Discrets sont des automates décrivant des processus stochastiques temporels. Si l'état courant q_t change à chaque instant, le modèle permet de calculer la probabilité de transition vers un état S_j à l'instant $t+1$ sachant l'état S_i dans lequel le modèle se trouvait à l'instant t. Considérant que cette probabilité est indépendante de l'instant t, on définit la matrice de transitions d'un Modele de Markov Discret à N états du premier ordre :

$$a_{ij} = P[q_t = S_j | q_{t-1} = S_i], 1 \leq i, j \leq N \tag{1.1}$$

avec

$$\sum_{j-1}^{N} a_{ij} = 1 \tag{1.2}$$

On peut ainsi connaître la probabilité d'observations d'une suite d'événements. Si notre modèle représente par exemple le temps qu'il fait et change chaque jour à heure fixe, passant dans les 3 états pluvieux, ensoleillé et nuageux, on peut calculer la probabilité que le temps passe par la suite d'états ensoleillé, nuageux, pluie, ensoleillé, nuageux, pluie, ou bien encore la probabilité qu'il pleuve pendant un an.

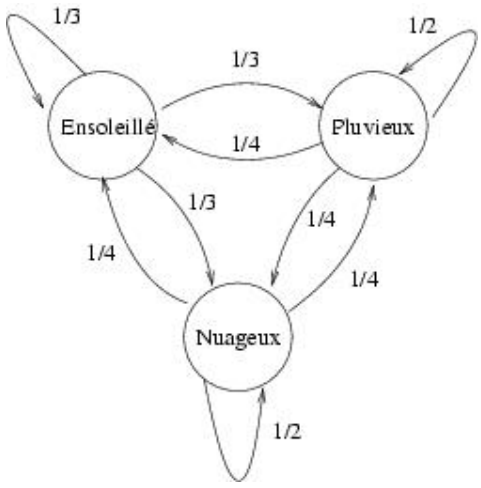

FIG. 1.3 – Modèle de Markov Discret à 3 états

Les Modèles de Markov Cachés ou HMM diffèrent des modèles discrets dans la mesure ou les états sont caractérisés par des distributions de probabilité sur l'espace des observations possibles, ces états ne sont alors plus observés directement. Dans notre exemple, on peut imaginer que

l'observateur, qui travaille et vit au sous sol d'un laboratoire, voit ses collègues arriver le matin avec ou sans imperméable, sans savoir si il pleut ou si ses collègues se méfient des caprices du temps alors même qu'il fait beau. On a alors deux modèles stochastiques liés : les imperméables de ses collègues et le temps qu'il fait.

Un modèle HMM est défini par :
- La matrice des a_ij, qui définit les probabilités des transitions d'un état S_i vers un état S_j (ou lui même)
- La matrice qui définit les probabilités d'émission des M observations pour chaque état $b_i(x_m) = p(x_m|S_i)$ avec $\sum_{m=1}^{M} b_i(x_m) = 1$
- La matrice donnant la distribution de départ des états.

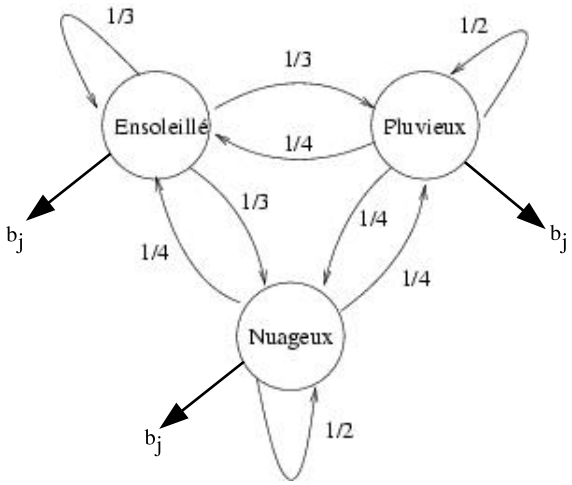

FIG. 1.4 – HMM à 3 états

Le modèle le plus répandu dans les techniques de reconnaissance de la parole continue est le modèle gauche droit d'ordre 1 dit de Bakis. Partant du principe que la parole est une suite d'évenements stationnaires, chaque état correspondant à un instant t_n admet une transition vers les états correspondant aux instants $t > t_n$. Il est parfaitement adapté à la parole dans son déroulement temporel puisqu'il permet de répéter les états dans le cas d'un déroulement plus lent, et de les sauter dans le cas d'un déroulement rapide.

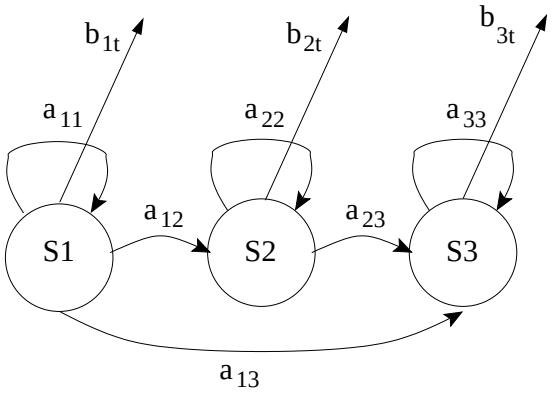

FIG. 1.5 – HMM à de Bakis d'ordre 1 à 3 états

Il est également possible de rajouter des états fantômes (ghost states) dont le but est de donner au systeme la souplesse de permettre au locuteur de tousser ou de begayer sans que la reconnaissance ne soit gravement affectée, ce qui est un risque lorsque la locution doit suivre trop fidèlement le modèle.

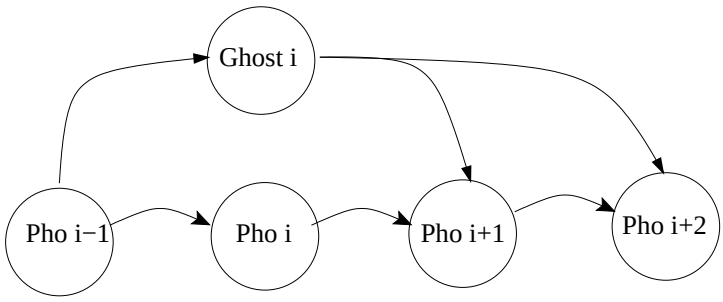

FIG. 1.6 – HMM avec ghost states

Le Modèle HMM est toujours couplé à une méthode de classification probabiliste comme une mixture de gaussiennes, un estimateur (maximum likehood), un classificateur à vaste marge (SVM) ou autre. Le lecteur pourra se référer à [Rab89] pour plus d'informations sur l'emploi des HMM en ASR.

Si les HMM représentent le modèle le plus adapté à la reconnaissance de la parole, le but de notre système est de réagir en temps réel et de pouvoir pour chaque frame donner une réponse. Nous n'utiliserons donc que très peu les HMM, en fin de chaîne. Elles ne sont pas implémentées dans le cadre de ce stage.

Chapitre 2

Modèles et paramètres utilisés

2.1 Le phonème par ses particularités spectrales

Comme vu précédemment, les phonèmes different par les caractéristiques spectrales mises en place pour les former. Ces caractéristiques, indépendantes pour la langue française du pitch (f0) ou de son mouvement (on pense ici par exemple aux differentes classes de phonèmes du vietnamien qui diffèrent selon les mouvements du pitch) résident dans l' enveloppe spectrale, c' est à dire la position des formants et anti-formants.

On sait qu' il est possible de caractériser les voyelles par la position des deux premiers formants. Un modèle de traitement du signal existe pour détecter les pôles d'une enveloppe spectrale. C'est le modèle auto-régressif ou par prédiction linéaire (LPC) [MG80, Opp78]. Il est utilisé pour le codage de la parole dans le cas du téléphone. Malheureusement, lors de la parole, les positions des formants se croisent, ils se séparent et leur nombre varie, ce qui rend leur détection difficilement automatisable. De plus, l'auto-régression est instable quand le nombre de formants demandé est différent du nombre réel, et deux configurations acoustiques semblables peuvent donner lieu à des coefficients LPC très différents. Enfin, il arrive souvent que les pôles LPC tombent sur un pic harmonique et non sur le centre d'un formant [Sch98, SR99].

Certaines classes de phonèmes sont aussi caractérisées par la présence ou non de friction, et dans le cas de la voix non chuchotée, la présence de voisement. Ces particularités peuvent être renseignées par des calculs adaptés.

Nous utiliserons pour caractériser ces particularités spectrales des coefficients Cepstraux, les MFCC, ainsi que des coefficients renseignant le taux de voisement du signal, le taux de passage par zero, l'autocorrelation du premier ordre, et le taux d'aperiodicité donné par l'algorithme Yin.

2.2 Coefficients

Tous les coefficients sont calculés sur une fenêtre glissante. Le premier calcul exécuté est l'algorithme Yin, proposé par A. de Cheveigné [dCK02]. Cet estimateur de f0 nous permet d'adapter la

taille des fenêtres d'analyse à la période du signal, et ainsi d'en réaliser une analyse pitch-synchrone. Cette faveur, réservée aux sons voisés, permet de réduire de manière évidente la variabilité des coefficients. Une fenêtre d'analyse contient deux périodes, ce qui évite la plupart des effets de fenêtrage. En revanche, comme les pas d'incrémentation ne sont pas synchronisés sur les pulses glottaux, les pics d'énergie de la forme d'onde ne coincident donc pas dune fenêtre à l'autre. Mais ce problème de phase ne joue que de façon très peu signifiante dans l'instabilité des coefficients.

Déscripteurs cepstraux

Ils permettent une description approximée de l' enveloppe spectrale en un nombre réduit de coefficients.

Le spectre donné par la FFT contient des renseignements sur la source et le conduit vocal, mais leur intermodulation rend difficile la mesure de F0 et celle des formants qui caractérisent respectivement source et conduit. Le lissage cepstral ou cepstre est une méthode visant à séparer leur contribution dans le spectre par déconvolution. On fait alors l'hypothèse que le signal vocal s_n est produit par une source excitatrice g_n traversant un système linéaire passif de réponse impulsionnelle b_n.

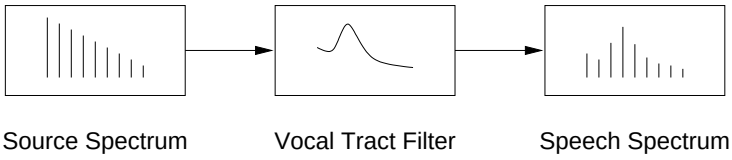

FIG. 2.1 – Modèle source filtre spectral

$$s = g * b \tag{2.1}$$

et leurs transformées en z respectives

$$S(z) = G(z)B(z) \tag{2.2}$$

Le principe de déconvolution est de se situer par homomorphisme dans un espace ou l'opérateur de convolution $(*)$ correspond à un opérateur d'addition $(+)$. En supposant que g_n est une séquence d'impulsions periodique pour les sons voisés ou de bruit pour les sons fricatés, il vient que sa contribution agit plutôt dans des variations "rapides" du spectre. En revanche, le conduit vocal (b_n) donne au spectre sa forme générale, ce qui concerne les variations "lentes" du spectre.

On définit cet homomorphisme par :

$$DM_*^+ = Z(.) \circ \log|.| \circ Z^{-1}(.) \tag{2.3}$$

Et son inverse :

$$DM_+^* = Z(.) \circ \exp|.| \circ Z^{-1}(.) \tag{2.4}$$

où Z est la transformée en Z et Z^{-1} la transformée inverse.

On a alors :

$$s_n = g_n * b_n \xrightarrow{DM_*^+} \tilde{s_n} = \tilde{g_n} + \tilde{b_n} \tag{2.5}$$

En posant l'hypothèse que g_n se réduit à une séquence d'impulsion séparés de n_0 échantillons (ou n_0 correspond à F0) et que $\tilde{b_n}$ décroît rapidement, on peut séparer les contributions de la source et du filtre en ne conservant que les $n < n_0$ premiers coefficients cepstraux. On appelle n_0 l'ordre du lissage cepstral. Ce filtrage par une fenêtre rectangulaire s'appelle *liftrage*.

Coefficients MFCC

Le principe de calcul des MFCC (Mel-scaled Frequency Cepstral Coefficients) est issu des recherches psychoacoustiques sur la tonie et la perception des différentes bandes de fréquences par l'oreille humaine. La FFT passe dans un banc de filtres à l'Echelle de Mel [Ste37]. Cette échelle, non linéaire, tient principalement compte du fait que la perception des intervalles change suivant la zone du spectre à laquelle les hauteurs qui les composent appartiennent. Le principal interêt de ces coefficients est d'extraire des informations pertinentes en nombre limité en s'appuyant à la fois sur la production (théorie Cepstale) et à la fois sur la perception de la parole (échelle des Mels).

Le calcul se déroule de la manière suivante :
- La FFT est calculée sur le fragment (frame).
- Cette dernière est filtrée par un banc de filtres triangulaires répartis le long de l'echelle de Mel.
- Le logarithme module de l'énergie de sortie du banc de filtres est calculé.
- Une Transformée en Cosinus Discrète inverse, (équivalente à la FFT inverse pour un signal réel) est appliquée.
- Seul les premiers coefficients sont conservés.

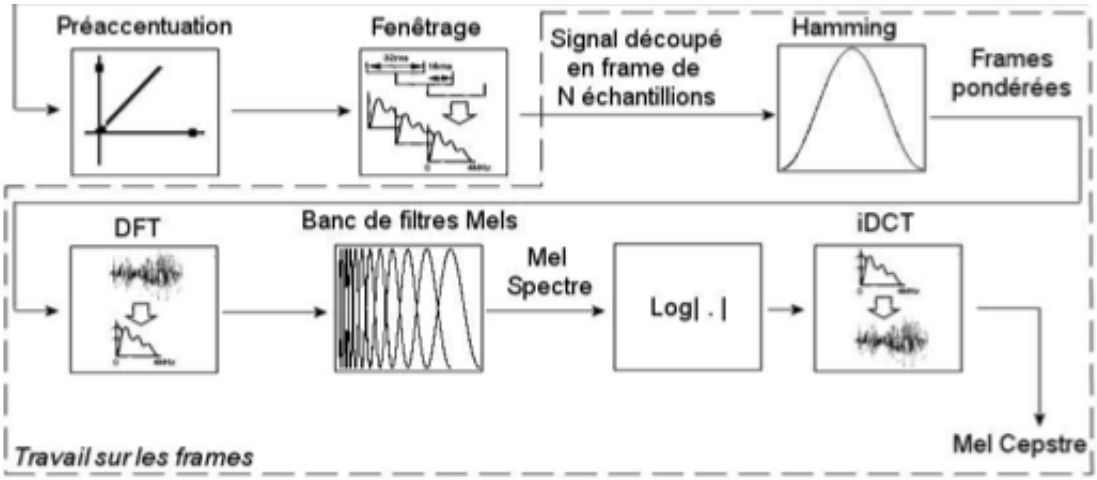

Fig. 2.2 – Schéma de calcul des MFCC

Taux de passage par zéro

Le taux de passage par zéro (Zéro Crossing Rate en anglais) est un indice très simple à calculer qui donne une indication sur le taux de bruit contenu dans le signal. Il représente le nombre de fois que le signal, dans sa représentation amplitude/temps, passe par la valeur centrale de l'amplitude (généralement zéro). Du fait de sa nature aléatoire, le bruit possède un taux de passage par zéro supérieur à celui de la parole voisée. Pour un signal x de longueur N, sa formule est donnée par :

$$zcr(x) = \frac{1}{N}\sum_{n=1}^{N-1}\begin{cases} 1\ si\ x_{n+1} - x_n < 0 \\ 0\ sinon \end{cases} \tag{2.6}$$

Il existe toutefois un paramètre plus efficace pour déterminer le taux de voisement d'un signal.

Premier coefficient d'autocorrélation

L'autocorrélation est la convolution du signal avec lui même. C'est une méthode couramment utilisée pour déterminer la fréquence fondamentale. Le résultat d'une autocorrélation est dans le cas d'un son voisé (possédant une f0) une suite de lobes espacés de n_0 échantillons. La fréquence fondamentale peut être déterminée en prenant l'inverse de la distance entre les deux premiers lobes de la deuxième moitié de la courbe. Le premier coefficient correspond à la corrélation de deux échantillons consécutifs du signal. Un échantillon d'écart correspond à une très haute fréquence. Or un bruit contient beaucoup plus d'énergie dans les très hautes fréquences qu'un son voisé. Deux échantillons consecutifs sont donc fortement décorellés dans le cas de bruit blanc. Ainsi, ce coefficient donne une information très fiable du rapport signal/bruit, pour de très faibles temps de calculs. Pour un signal x de longueur N, sa formule **normalisée en énergie** est donnée par :

$$fac(x) = \frac{\sum_{n=0}^{N-2} x_n \cdot x_{n+1}}{\sqrt{\sum_{n=0}^{N-2} x_n^2}\sqrt{\sum_{n=0}^{N-2} x_{n+1}^2}} \tag{2.7}$$

2.3 Partis pris de classification

L'idée principale du stage était d'extraire au niveau spectral les informations permettant la classification la plus discriminatoire possible. Il était délicat de mettre en place un classificateur par phonème, aussi nous avons décidé de procéder par étapes. Si au fur et à mesure la classification est de plus en plus précise, nous ne pouvons néanmoins pas repérer tous les phonèmes. Nous procédons donc à une classification et non à une reconnaissance.

Certains phonèmes ne peuvent être reconnus par un système basé uniquement sur ses particularités spectrales. Typiquement, les plosives nécessitent un suivi des différentes phases articulatoires, qui donnent lieu à différentes configurations spectrales, et ce pour un même phonème. Tout d'abord un silence (ou un son voisé dans le cas des plosives sonores), puis une barre d'explosion ("burst") provoquant une très courte perturbation acoustique, et enfin un court bruit de friction. Si chacun de ces phénomènes a ses particularités spectrales, seule une bonne combinaison des trois caractérise

un phonème. Cette classe ne peut être reconnue sans employer un suivi temporel par HMM ou DTW.

Les nasales et les liquides, présentant des déroulements temporels particuliers, et très sensibles à la coarticulation, sont abandonnées afin de réduire les difficultés auxquelles le système doit faire façe.

Aussi certaines classes, trop ressemblantes, ont été rassemblées pour éviter de surcharger le système de difficultés. C'est le cas de [@], noté comme [2], [O], noté comme [o], et [9∼], noté comme [e∼].

Les différents niveaux de classification peuvent être résumés dans des arbres binaires de décision. Les phonèmes n'étant pas présents dans la base de données ne figurent pas dans cette représentation.

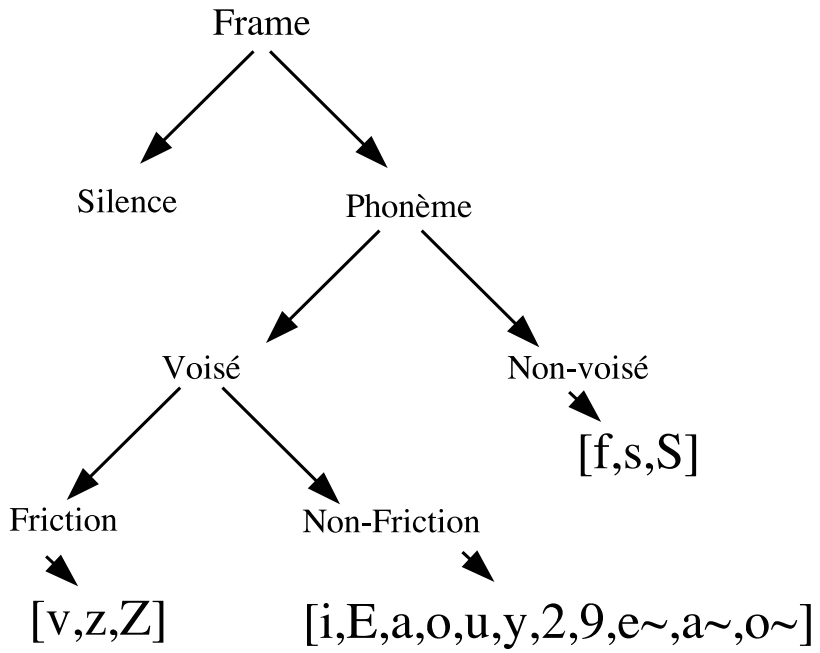

FIG. 2.3 – Arbre de classification de Type I

Dans une classification de Type I (figure 2.3), outre la séparation phonème-silence, la première classification est basée sur la présence ou non de voisement, information véhiculée principalement par les coefficients zero-cross et autocoréllation du 1er ordre. Dans un deuxième temps on cherche à regrouper les phonèmes dont la prononciation nécessite de la friction. Pour chacune des trois classes résultantes, on mettra en place un classificateur KNN (vois section 3.2) qui prendra une décision précise sur l'identité du phonème.

Dans une classification de Type II (figure 2.4), on cherchera à reconnaître moins de classes différentes. La première classification se fait sur la présence de friction, on cherche ensuite à séparer les différentes classes de fricatives d'un côté, et de voyelles de l'autre.

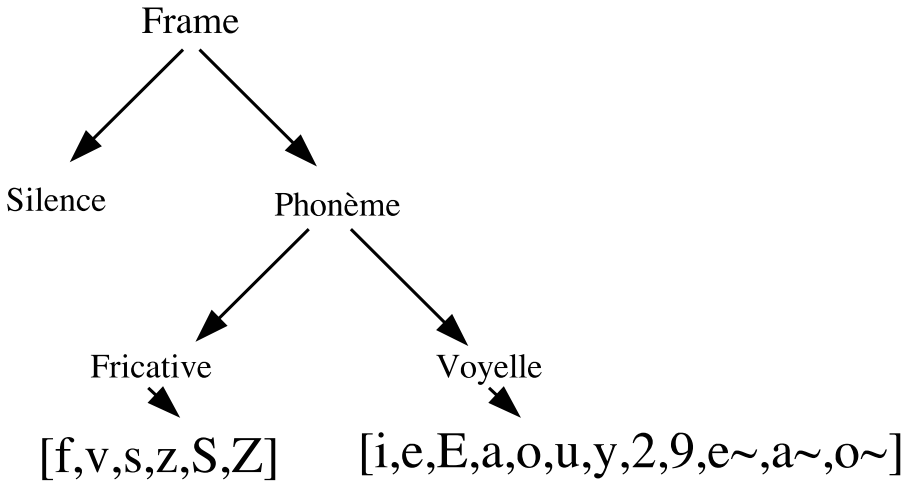

FIG. 2.4 – Arbre de classification de Type II

Chapitre 3

Machine Learning

3.1 Méthodes statistiques de traitement des données

Les methodes statistiques de traitement des données visent à préparer les données pour le classificateur.

Analyse en composantes principales (PCA) L' analyse en composante principales est une méthode véctorielle linéaire de réduction des dimensions de parametres non supervisée, choisissant les directions dont la variance intra-cluster est la plus grande. Les données sont alors plus facilement visualisables sur moins de dimensions.

La PCA se calcule à partir de la matrice de covariance des données. Celle ci est diagonalisée afin d' en extraire les valeurs et vecteurs propres. Les données sont projetées dans l' espace défini par les vecteurs propres. Les valeurs propres, classées dans l' ordre décroissant, correspondent dans l'espace d'arrivée au vecteur propre dont la direction maximise la variance.

Toutefois, la PCA n'est pas une methode de classification ; si les clusters ont individuellement des grandes variances sur une dimension PCA alors celle ci sera jugée Composante Principale. Elle ne permet pas la séparation des classes.

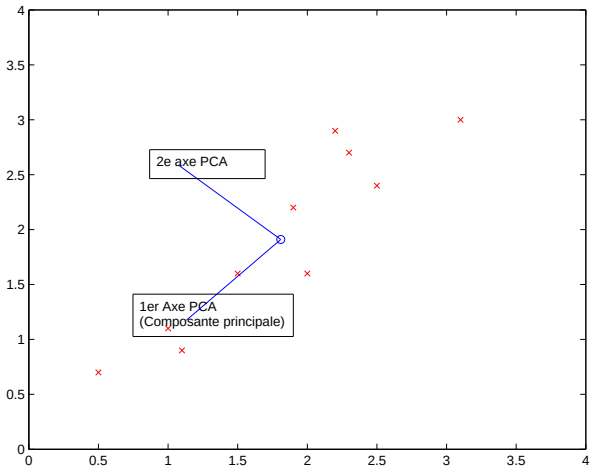

FIG. 3.1 – Composantes calculées par PCA

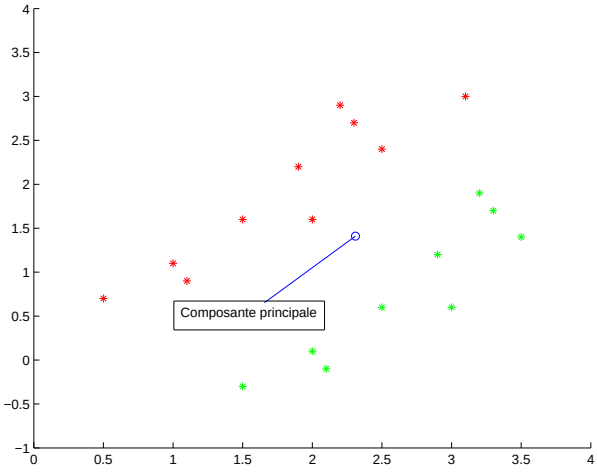

FIG. 3.2 – PCA ne permet pas la classification

LDA L' analyse discriminante linéaire est une méthode similaire mais cette fois ci supervisée. Elle est utilisée pour optimiser le rapport entre la dispersion "intraclusters" et la dispersion "inter-clusters" Les directions jugées principales sont celles dont la variance inter-clusters est la plus grande. Les exemples sont alors plus facilement séparables sur moins de dimensions.

La LDA attribue à chaque cluster une moyenne et une variance, et obtient ainsi une *scatter-matrix*, représentant les distances séparant les clusters les uns des autres autour de la moyenne de l'ensemble des points. Sur ces données est réalisée une PCA qui aura donc pour but de favoriser les variances inter-cluster.

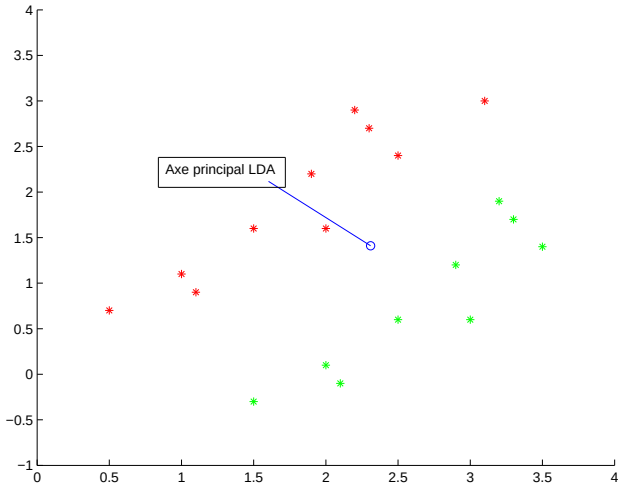

FIG. 3.3 – LDA favorise la séparation de classes

Si la LDA represente les données dans un espace permettant une classification rapide en sous classes, il ne faut pas négliger la contribution des directions secondaires qui portent tout de meme une information, et peuvent faire la difference pour une classification plus subtile.

Attention ! PCA et LDA ne sont pas des méthodes de classification. Elles décrivent la transformation linéaire permettant une meilleure lisibilité des données. L 'information importante (grande variance, séparabilité) est concentrée dans les premières dimensions du nouvel espace. Si les données n'étaient pas séparables avant le traitement statistique, elles ne le seront pas plus après.

3.2 Méthodes de classification

Une fois la base de données constituée, avec une indication de la classe associée à chaque set de coefficients, il existe diverses methodes pour les confronter à un vecteur de coeffs exterieur, afin de lui associer une classe. Elles sont issues des statistiques et de " l' intelligence artificielle ".

K - plus proches voisins (KNN) La méthode des kppv offre l'avantage d'être très simple et néanmoins efficace. Son principe consiste à calculer une distance entre l'individu à classer et les individus connus, puis à attribuer le premier à la classe présentant le plus grand effectif parmi ses k plus proches voisins. Outre sa simplicité, cette méthode est couramment employée en reconnaissance des formes, parce qu'elle s'y prête à de nombreux titres : tout d'abord, elle ne nécessite pas de connaître la distribution de probabilité des classes de la population, ce qui est rarement le cas. Ensuite, comme on fixe le nombre k de voisins et pas le volume, la méthode ne dépend pas de la densité de probabilité. L'expérience montre de plus que cette méthode présente souvent un bon pouvoir prédictif.

Distance de Mahalanobis Afin d'éviter les effets dus à l'inégalité du nombre d'occurrences de chaque classes, le statisticien P.C. Mahalanobis a mis au point un calcul des distances basé sur la variance des clusters. Pour chaque cluster, la moyenne et la matrice de covariance est calculée. Ces données définissent la transition vers l'espace centré sur la moyenne du cluster, et dont la base est normée selon les variances des directions du cluster. Ainsi on peut calculer pour tout point test sa distance au cluster en calculant dans l'espace d'arrivée la distance entre son origine (correspondant à la moyenne du cluster dans l'espace des paramètres ou *feature space*) et le point test.

C'est cette distance que l'algorithme LDA tente de maximiser entre les clusters. Ce calcul est bien plus rapide que l'algorithme KNN puisque le nombre de distances à déterminer est égal au nombre de clusters et non au nombre d'examples. Or en reconnaissance de la parole on est contraint à travailler avec les bases de données les plus grandes possibles, pour lesquelles le temps de calcul de KNN n'est plus adapté à une application en temps réel.

3.3 Résumé du dispositif

Notre dispositif de classification se présente donc de la manière suivante :
- Fenêtrage du signal par une fenêtre de Hamming
- Calcul du set de coefficients pour la fenêtre (MFCC, ZCR, FAC, (+YIN))
- En partant de la racine de l'arbre de classification :
  - Calculer KNN pour les coefficients **dans l'espace LDA** correspondant au noeud parcouru.
  - Selon le résultat de KNN, passer au fils correspondant ou conclure.

Chapitre 4

Constitution d'une base de données

Le bon fonctionnement des classificateurs depend en grande partie de la robustesse de la base de données d'apprentissage. Si les données ne sont pas séparables, il sera impossible au classificateur de trancher en faveur d'une ou d'une autre classe.

Le but du système était de s'approcher le plus possible d'un reconnaisseur multilocuteur, fonctionnel dans tous types d'environnement, avec tous types de matériel. La base de données d'entraînement devait donc être fidèle au données qui y seraient confrontées lors de l'utilisation du système.

4.1 Variabilité de la parole

Le majeur problème rencontré provient de la variabilité de la parole ; aucun locuteur n' est capable de respecter à la seconde près la prononciation exacte (on dira plutôt optimale) d'une phrase.

Hélas la reconnaissance de la parole par l'homme est un processus cognitif qui ne nécessite pas une compréhension exacte de chaque unité acoustique. Elle s'effectue surtout au niveau des mots du langage, par une sorte de suivi. C'est pourquoi la plupart des systèmes ASR se basent sur un modèle de suivi par HMM ou DTW (voir section 4.4). Lors de la prononciation d' un mot, le cerveau l'identifiera de la même façon or certaines syllabes sont obligatoirement "fausses" (un phonème est prononcé à la place d' un autre).

Mes premiers travaux ont porté sur des enregistrements divers de voix parlée ; voix d'acteurs, d'amis, ma propre voix... Après une longue phase de segmentation (à la main), et les coefficients calculés, il était impossible de définir une méthode pour séparer efficacement ces clusters. Il était également possible à l'écoute de se rendre compte que la concaténation des fragments audio d'une classe comportait énormément de fragments qui appartenaient à d'autres classes ou qui étaient inclassables. Ce pour deux principales raisons :

– Lorsque l'on parle à vitesse normale, le temps de parole stable (rester sur un phonème) est au mieux aussi long que le temps de *cross-over*, temps de transition pendant lequel le son

prononcé est un hybride des deux phonèmes. Dans la plupart des extraits sur lesquels je travaillais, sur toute la durée d'un phonème (environ 15 périodes) seulement six ou sept périodes de signal représentaient le phonème "pur" et permettaient de l'identifier hors contexte.
- Coté locuteur, on ne prononce jamais exactement les bonnes unités phonétiques, on a parfois même tendance à prononcer un phonème à la place d'un autre. Ce phénomène est visible chez tous les locuteurs, ayant leur diction propre, et non uniquement chez les personnes ayant un accent étranger, et donc associant différemment les unités symboliques aux unités phonétiques. Le cerveau s'efforcera d'entendre le bon phonème, alors que hors contexte il associerait le fragment de signal à une autre classe.

4.2 Continuité de l'espace articulatoire

On sait pourtant que si l'espace articulatoire des voyelles ou des fricatives est continu, c'est à dire qu'il est physiquement possible de passer d'un phonème à l'autre sans discontinuité articulatoire, il existe une limite perceptive permettant de distinguer les différentes classes de phonèmes hors contexte sémantique[NWGD05]. Cette limite apparaît chez tous les sujets sondés. Il est donc possible de déterminer cette limite permettant une classification "aveugle" (hors contexte) des phonèmes, ou au moins des unités phonétiques.

4.3 Enregistrements à l'Ircam

Nous avons donc décidé de créer une base de données robuste. Cette base de données devait répondre à plusieurs critères :
- Les nombres d'occurrences des différents phonèmes doivent être équilibrés, afin que les classificateurs ne favorisent pas une classe qui serait plus représentée qu'une autre.
- Les phonèmes doivent rester stables sur une certaine durée, pour que les clusters soient le plus possibles concentrés autour d'une moyenne et ainsi réduire la variance interne au cluster, et éviter les effets de recouvrement entre les différents clusters.
- Enfin, les clusters doivent ne comporter que des données associées à des phonèmes appartenant à la classe représentée, afin d'assurer la cohérence de la base.

Pour réunir ces trois conditions, nous avons demandé à des volontaires (du personnel de l'Ircam) de lire un set de 10 phrases phonétiquement équilibrés. Ce set contient pour chaque phonème un nombre proportionnel à sa présence dans la langue française [Com81]. La lecture de ce texte à été faite par 28 personnes (9 femmes, 19 hommes) dans d'excellentes conditions de prise de son, une cabine du bureau Design Sonore et un microphone statique Neumann d'excellentes qualités. La solution pour améliorer la stabilité des phonèmes a été de demander aux locuteurs de faire de ces phrases, en plus d'une diction normale, une diction lente, en faisant durer chaque phonème entre une et deux secondes. Cet astuce a pour conséquence, outre le fait que chaque phonème est prononcé avec une durée de stabilité importante, de provoquer un phénomène de rétro action sur le locuteur. Ce dernier entend alors bien le phonème qu'il prononce et rectifie sa diction.

Finalement, seules 18 personnes sur les 28 ont réussi à prononcer les dix phrases de la façon espérée. Malgré mes efforts pour leur expliquer 28 fois ce que j'attendais d'eux, il semble qu'inconsciemment 10 d'entre eux n'arrivaient pas à réussir cet exercice, probablement par habitude ou

par peur du ridicule. Ceci nous donne donc une indication sur les problèmes à résoudre pour la prochaine création d'une base de données de ce type.

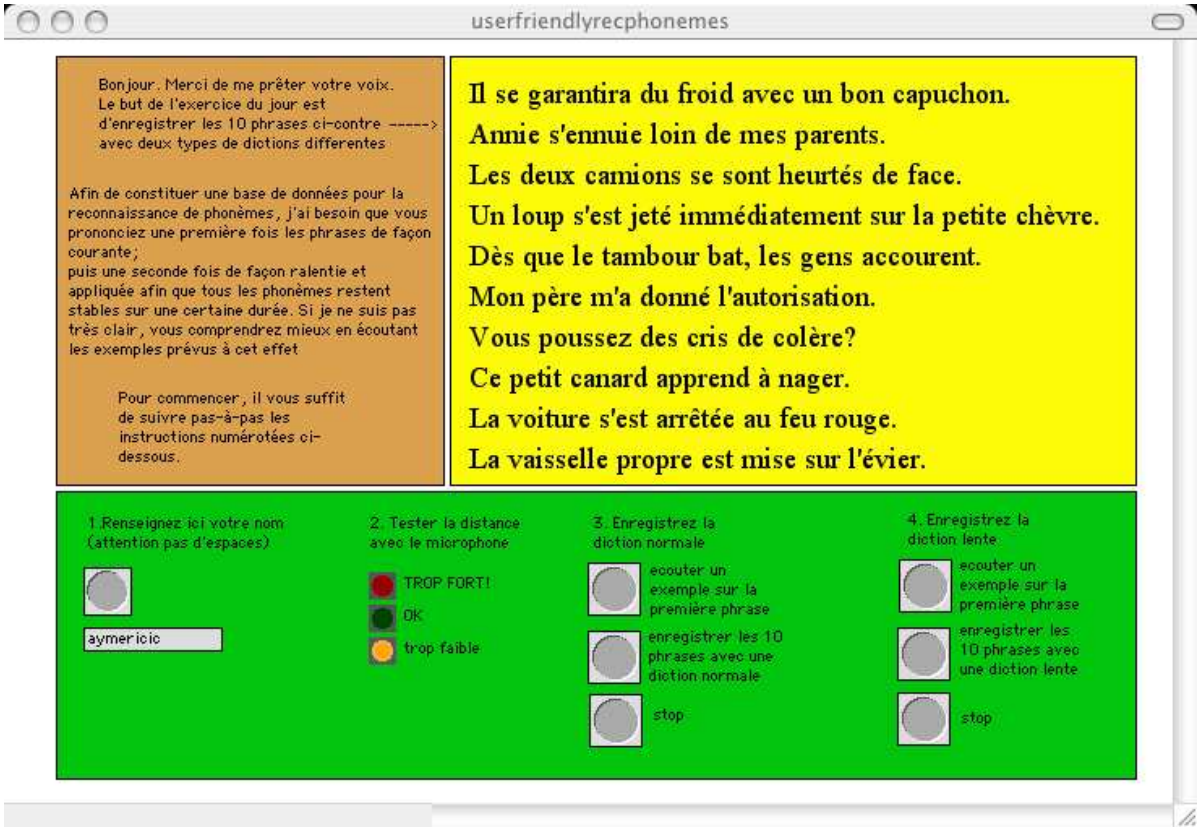

FIG. 4.1 – Interface utilisateur pour la prise des sons constituant la base de données

4.4 Alignements

Pour éviter de segmenter à la main une base de données suffisamment grande pour assurer le bon fonctionnement des classificateurs, nous avons pris pour principe qu'un maximum de dictions d' une même phrase pouvaient être alignées à l'aide de l' algorithme DTW. Ainsi, une segmentation d'un seul exemple peut être reporté sur les autres, favorisant également le caractère multi locuteur de notre système.

Dynamic Time Warping (DTW) L'une des méthodes les plus employées pour l'alignement est Dynamic Time Warping [RJ93]. Cet algorithme se base sur un algorithme de recherche de chemin de Viterbi qui minimise les distances locales accumulées le long d' un chemin d' alignement. C'est une instance de la classe des algorithmes en $O(n^2)$ appelée *programmation dynamique*.

L'algorithme DTW cherche le meilleur chemin d'alignement entre deux réalisations d'un même phénomène, la réalisation source et la réalisation cible. Les distances locales sont calculées puis stockées dans la matrice des distances locales $mdl(m, n)$. Cette matrice donne pour chaque fragment m de la source et chaque fragment n de la cible la distance Euclidienne qui les sépare.

Ensuite, un chemin est calculé pour minimiser la distance ajoutée entre les points $(1, 1)$ et (M, N) avec M et N respectivement nombre de fragments de la source et de la cible. Il est calculé de façon itérative, en mettant à jour la matrice des distances ajoutées $mda(m, n)$, qui donne la distance du meilleur chemin reliant $(1, 1)$ de (m, n). Une matrice, $\psi(m, n)$ conserve les coordonnées du point précédant dans le chemin. Il existe différents types de méthodes DTW, mais toutes ne diffèrent que dans le calcul de $mda(m, n)$. Dans une implémentation de Type I, la matrice $mda(m, n)$ sera calculée à partir de la matrice $mdl(m, n)$ sous les contraintes suivantes :

$$mda(m,n) = \min \left\{ \begin{array}{l} mda(m-1,n-1) + w_d * mdl(m,n) \\ mda(m-1,n) + w_v * mdl(m,n) \\ mda(m,n-1) + w_h * mdl(m,n) \end{array} \right\} \tag{4.1}$$

Avec w_d, w_v et w_h les poids attribués à chaque direction (diagonale, droite, haut), on peut jouer sur la direction globale de l'alignement (favoriser l'alignement, le retard ou l'avance de la cible sur la source). Le chemin optimal, selon la fidélité de la cible à la source, doit se rapprocher de la diagonale.

L'algorithme DTW est donné dans l'algorithme 1.

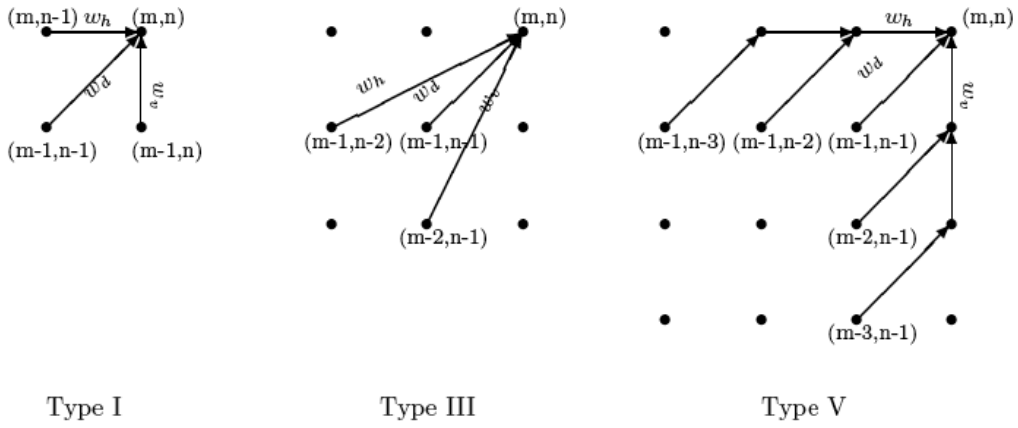

FIG. 4.2 – Voisinage du point (m, n) pour les Types I, III et V

Algorithm 1 Dynamic Time Warping

Initialisation
$mda(1,1) = mdl(1,1)$
$\psi(1,1) = (0,0)$
for $2 \leq n \leq N$: **do**
  $mda(1,n) = mda(1,n-1) + w_h * mdl(1,n)$
  $\psi(1,n) = (1,n-1)$
end for

Déroulement frame par frame
for $2 \leq m \leq M$: **do**
  $mda(m,1) = mda(m-1,1) + w_d * mdl(m,1)$
  $\psi(m,1) = (m-1,1)$
  for $2 \leq n \leq N$: **do**
    $$mda(m,n) = \min \left\{ \begin{array}{l} mda(m-1,n-1) + w_d * mdl(m,n) \\ mda(m-1,n) + w_v * mdl(m,n) \\ mda(m,n-1) + w_h * mdl(m,n) \end{array} \right\}$$
    Donner à $\psi(m,n)$ les coordonnées du point minimisant la distance
  end for
end for

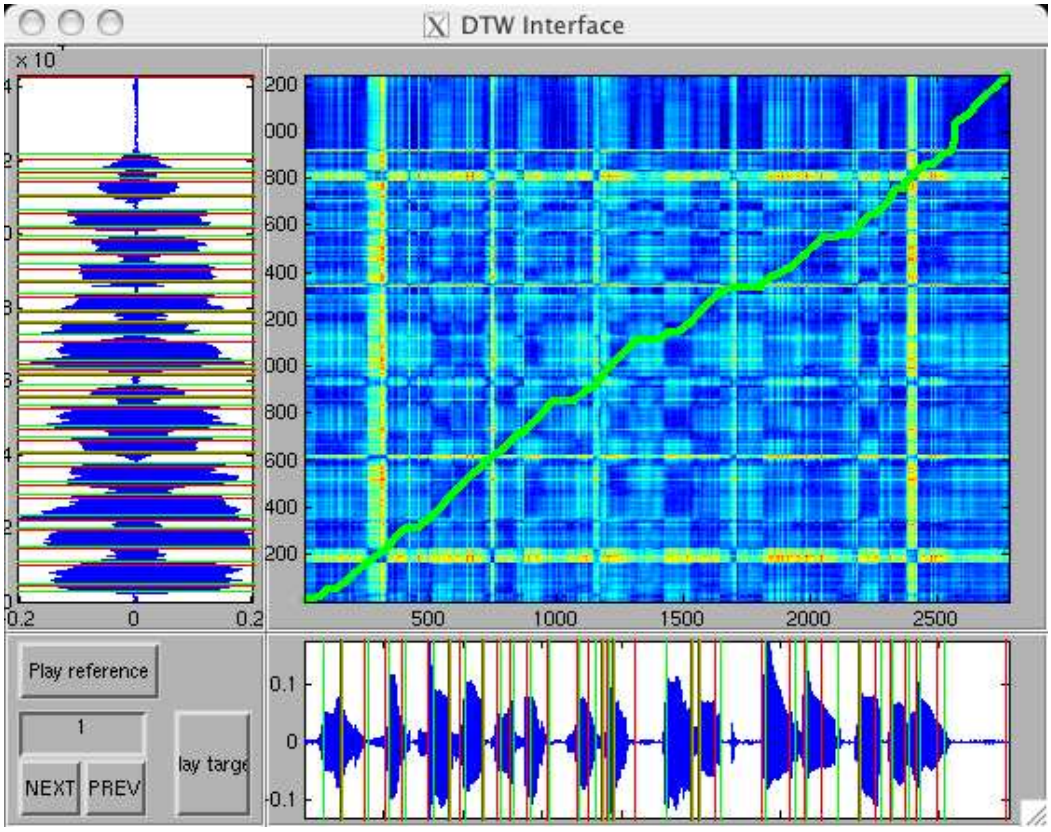

FIG. 4.3 – Alignement avec DTW

4.5 Optimisation

Si les données sont encore trop éparpillées, on peut de plusieurs manières retirer les points aberrants c'est à dire ceux qui ne sont pas représentatifs de leur classe. La méthode que nous utilisons consiste à ne conserver qu'un certain pourcentage de points, les plus proches du cluster selon la distance de Mahalanobis, soit les plus proches de la moyenne du cluster avec pour base les vecteurs variance du cluster.

Chapitre 5

Tests et résultats

Toutes les méthodes ont été testées en Matlab. Les automatisations de calculs et de tests sur les bases de données, son architecture, les algorithmes DTW et KNN ont été implémentés par mes soins, tandis que les calculs de MFCC, PCA et LDA sont issus de Toolboxes pour Matlab.

Les tests ont été réalisés sur les enregistrements réalisés à l'Ircam, mais cette fois ci sur la diction normale des dix phrases phonétiquement équilibrées. Une fois segmentées, les coefficients des parties labellisées sont passées dans le classificateur. On a ainsi un moyen de contrôler le taux d'erreurs du système.

5.1 Comparaison des bases de données

Les premiers tests ont été effectués sur des bases de données de parole normale, telles que le corpus d'oeuvres de Jean Cocteau, lues par Xavier Rodet dans le cadre du projet Talkapillar [BSHR05], ou les dictions de cinq sets de dix phrases phonétiquement équilibrées, effectuées par quelques proches ayant l'accent parisien. Le dispositif était totalement défectueux sur ces bases de données, si bien qu'il atteignait le score de 0% de reconnaissance correcte ou au moins cohérente.

La nouvelle base de données nous a permis d'obtenir quelques résultats. En comparant la prononciation du locuteur testé et la transcription phonétique du texte à lire, il est évident que le taux de classification correcte ne peut atteindre les 100%. Hélas les résultats obtenus n'ont pas été très brillants puisqu'aucune des configurations testées à ce jour n'a dépassé les 50%.

5.2 Jeu de Paramètres 1

Les résultats présentés ci-après correspondent à un certain jeu de paramètres. On emploie un classificateur de Type II utilisant l'algorithme des K-nearest neighbours.

- Les extraits sonores, enregistrés à une fréquence d'échantillonnage de 44100 Hz, sont décimés à 11025 Hz, soit un rapport de 1/4, afin de ne conserver que les zones spectrales utiles pour la parole (30 Hz - 5000 Hz).
- Les coefficients sont calculés tous les 64 échantillons, pour des fenêtres représentant deux périodes pour les signaux voisés, et 128 échantillons pour les sons non voisés.

– Les coefficients MFCC sont au nombre de 12, plus les coefficients zero-crossing-rate et first-auto-co, ainsi que l'indice de périodicité donné par l'algorithme YIN. En tout, un *feature vector* possède 15 composantes.
– Le nombre de K nearest neighbours calculés est de 75. Parmi ces 75, on prend la population la plus représentée.
– La proportion de *feature vectors* conservée est de 50% par cluster, regroupés autour de leur moyenne.

5.3 Résultats

Le dispositif n'est testé que sur des occurrences des phonèmes qu'il est censé savoir reconnaître. Ces dernières sont issues d'une segmentation de parole continue.

%	f	v	s	z	S	Z	i	e	E	a	o	u	y	2	9	e∼	a∼	o∼	_
f	0	0	0	0	0	0	0	**78**	0	05	0	0	0	0	11	0	02	0	02
v	0	0	05	0	01	0	19	17	01	10	0	0	02	05	0	0	04	01	**30**
s	0	0	0	0	0	0	08	24	0	0	0	**66**	0	0	0	0	0	0	<1
z	0	0	04	0	0	0	**28**	**28**	0	04	0	0	0	16	0	0	0	0	2
S	0	0	03	0	18	0	03	25	0	0	0	**40**	0	0	09	0	0	0	0
Z	0	05	03	0	03	0	**22**	18	06	08	0	0	03	10	05	0	0	0	12
i	0	<1	04	0	0	0	27	03	02	**28**	0	<1	01	09	0	14	0	0	06
e	0	02	01	0	0	0	03	04	03	32	0	0	10	**35**	<1	02	0	<1	04
E	0	0	18	0	0	0	0	02	09	03	0	0	0	**47**	0	0	0	0	19
a	0	0	05	0	0	01	01	12	<1	**43**	02	0	0	21	0	<1	02	<1	09
o	0	0	0	0	0	04	02	**27**	0	08	0	02	0	16	0	0	06	12	20
u	0	<1	02	0	0	<1	02	10	0	05	0	0	<1	03	0	0	17	21	**36**
y	0	08	02	0	0	0	04	06	02	**34**	0	0	3	04	08	0	0	0	02
2	0	02	01	0	0	0	04	06	<1	08	0	0	04	**64**	0	0	<1	02	04
9	0	0	0	0	0	0	0	16	0	16	0	0	0	**58**	0	0	0	0	08
e∼	0	0	04	0	0	0	0	0	0	**53**	02	0	0	04	0	0	02	02	24
a∼	0	0	01	0	0	0	02	06	0	1	0	0	0	02	0	0	**71**	03	03
o∼	0	0	0	0	0	0	0	09	0	06	0	0	0	02	0	0	03	27	**51**
_	0	0	0	0	0	0	<1	05	01	03	0	<1	0	<1	<1	0	03	0	**83**

TAB. 5.1 – Répartition des resultats

Le tableau 5.1 résume les répartitions de la classification obtenue à l'aide de ces paramètres. Pour chaque phonème attendu, les différentes rangées donnent en pourcentage la proportion de réponses correspondant à chaque phonème. En gras est donné le score du phonème ayant le plus été reconnu pour chaque phonème attendu. Pour un dispositif optimal, les scores en gras devraient tous se trouver dans la diagonale. Ici on en est loin, aussi la proportion cumulée de bonnes réponses est de 29%, ce qui est un très mauvais score. Certaines classes bénéficient toutefois d'un classement raisonnable, comme le [a∼] avec 71% de prédiction correcte, ou à la rigueur le [2] avec 64% de prédiction correcte.

Il est également impressionnant de constater l'inefficacité de notre classificateur "bas-niveau" faisant la distinction entre fricatives et voyelles : aucune fricative n'a été reconnue une seule fois à par pour la classe [S]. Pire, elles semblent avoir été classées comme des voyelles. Cette défaillance constitue un problème critique à résoudre d'urgence. De plus, on ne peut évaluer notre classificateur de fricatives vu qu'aucune d'entre elles n'a été testée.

Il est possible que la fréquence d'échantillonnage utilisée ne soit pas adaptée à la reconnaissance des fricatives, dont les résonances caractéristiques se situent dans les hautes fréquences. Il faut donc recalculer la base de données de *feature vectors* à une fréquence d'échantillonnage plus élevée, tout en veillant à ce que les MFCC conservent l'information sur l'enveloppe spectrale pour les basses fréquences (des sons voisés).

Une autre amélioration est envisageable pour faciliter la classification son voisé, son non voisé. On peut imaginer que si les coefficients zero-crossing rate, first auto-co et YIN aperiodicity décrivent correctement cet attribut du signal, en revanche le calcul de LDA peut être bruité par les MFCC et leur grande variance pour les fricatives. Si on se contente d'effectuer cette classification à l'aide seulement de ces coefficients non-cepstraux, il se peut qu'elle gagne en robustesse, à condition de séparer les cas de fricatives voisées, fricatives non voisées et voyelles.

5.4 Jeu de Paramètres 2

Les résultats présentés ci-après correspondent à un certain jeu de paramètres. On emploie un classificateur de Type I utilisant l'algorithme des K-nearest neighbours.

- Les extraits sonores, enregistrés à une fréquence d'échantillonnage de 44100 Hz, sont maintenant décimés à 22050 Hz, soit un rapport de 1/2, afin de conserver les zones spectrales utiles pour la differenciation des fricatives.
- Le reste des coefficients demeure inchangé.

5.5 Résultats

Si cette classification permet une meilleure distinction entre les sons voisés et les non voisés, en revanche la distinction entre fricatives voisées et voyelles est bien plus délicate. Il apparaît également que les différentes classes de voyelles sont bien moins bien classifiées que dans l'expérience précédente.

La classification son voisé-son non voisé utilisée au bas niveau de ce dispositif donne de bons résultats. Cela est dû à la fiabilité des coefficients zero-crossing rate, first auto-co et aperiodicity pour déterminer la présence ou non d'une fréquence fondamentale. En revanche, les fricatives voisées possèdent du bruit de friction ainsi qu'une fréquence fondamentale. Leur situation entre deux classes ne favorise pas leur classification. De plus certaines prononciation de voyelles introduisent du bruit de friction, comme lorsque le locuteur chuchote une syllabe. La classification des fricatives voisées en tant que classe fonctionne donc très mal.

%	f	v	s	z	S	Z	i	e	E	a	o	u	y	2	9	e∼	a∼	o∼	_
f	0	0	0	0	10	0	0	10	7	2	0	0	0	0	2	0	0	0	**69**
v	0	0	0	0	0	0	4	21	0	22	0	0	3	19	0	0	0	1	**30**
s	30	0	**34**	0	17	0	0	1	0	0	0	0	0	0	5	0	0	0	12
z	0	0	0	0	0	0	4	**44**	0	12	0	0	0	16	0	0	4	4	16
S	3	0	6	0	**72**	0	0	0	0	0	0	0	0	0	0	0	0	0	19
Z	0	0	0	0	0	0	26	7	0	9	0	10	0	10	0	0	3	0	**34**
i	0	0	0	0	0	0	13	**47**	0	8	0	0	0	7	0	0	7	5	14
e	0	0	0	0	0	0	9	**47**	0	15	0	0	0	5	0	0	13	8	2
E	0	0	0	0	0	0	0	20	0	**68**	2	0	0	0	0	0	7	2	1
a	0	0	0	0	0	0	0	21	6	**52**	1	0	0	5	0	0	6	2	7
o	0	0	0	0	0	0	4	4	2	**77**	0	0	0	6	0	0	4	2	0
u	0	0	0	0	0	0	3	1	4	**50**	0	0	0	8	0	1	5	4	25
y	0	0	0	0	0	0	12	16	0	22	0	0	0	14	0	2	2	**24**	8
2	0	0	0	0	0	0	9	**33**	0	25	2	3	0	7	0	0	9	5	8
9	0	0	0	0	0	0	0	8	0	**67**	17	0	0	8	0	0	0	0	0
e∼	0	0	0	0	0	0	0	7	12	**39**	2	0	0	22	0	0	2	0	15
a∼	0	0	0	0	0	0	1	10	1	**69**	0	0	0	14	0	0	2	0	2
o∼	0	0	0	0	0	0	2	7	0	**61**	0	1	0	7	0	0	11	2	7
_	0	0	0	0	0	0	1	3	3	3	0	0	0	2	0	0	1	0	**84**

TAB. 5.2 – Répartition des resultats

On peut également remarquer sur le tableau 5.2 que les voyelles sont généralement très mal classifiées. Cela est probablement dû au changement de la fréquence d'échantillonnage. Les coefficients cepstraux se répartissent alors sur un spectre deux fois plus large, et représentent les caractéristiques acoustiques du signal avec moins de précision.

Il est clair que de tels résultats sont insuffisants pour assurer la robustesse d'un reconnaisseur de parole. Il reste énormément d'améliorations à trouver pour prétendre à exécuter une telle tâche.

A l'heure ou ce rapport est rédigé, des calculs sont encore en cours afin de déterminer les paramètres donnant les meilleures performances. Par la suite, de nouveaux tableaux de résultats avec leur analyse seront présentés.

Chapitre 6

Améliorations et travaux à venir

Durant ces quatre mois de stage, un certain temps a été nécessaire pour définir une vraie problématique et une méthodologie de travail. Ainsi l'enregistrement de la base de données d'entraînement a été un vrai moteur à un moment où mes recherches n'avançaient plus. Les quelques résultats présentés ont été calculés pour un ensemble de paramètres fixés (nombre de coefficients cepstraux, nombre de plus proches voisins pour KNN, pourcentage de points centraux à conserver pour un cluster, nombre de dimensions LDA à conserver, etc...). Aussi aucun test n'a été fait quant à l'influence de ces paramètres sur la qualité de la reconnaissance. Il reste donc un vaste travail de recherche à réaliser sur ces paramètres afin de trouver la combinaison donnant la meilleure reconnaissance.

6.1 Amélioration des bases de données

Afin de mieux comprendre où se situent les défaillances du système et celles des bases de données d'apprentissage et de validation, il faudrait pouvoir limiter au maximum celles des bases de données. Or les locuteurs ne sont pas forcement responsables du phénomène de haute variabilité de la parole. Il faudrait donc faire en sorte que chaque *feature vector* corresponde bien à la classe à laquelle il a été assigné, plutôt que d'attendre que le locuteur prononce le phonème qu'il est censé prononcer. Une solution pour constituer cette base de données de validation serait, pour chaque *feature vector*, de récupérer le grain de signal correspondant et, à l'aide d'une synthèse granulaire, de l'écouter seul sur une durée suffisamment longue pour lui attribuer la classe à laquelle il semble appartenir. Si on en croit les théories sur les frontières perceptives entre les phonèmes [NWGD05], les clusters ainsi obtenus ne se recouvriraient pas. En revanche, le travail de classification "à l'oreille" serait long et exhaustif.

6.2 Amélioration des descripteurs

Il semble également que certaines améliorations concernant les descripteurs eux mêmes peuvent être apportées au système. Il a été prouvé que l'utilisation des coefficients cepstraux multi-résolution par sous-bandes donnaient de meilleurs résultats pour la reconnaissance à l'aide de HMMs et de mixtures de Gaussiennes sur la base de données TIMIT [McC98]. Le principe est de diviser le spectre en sous-bandes [BD97], et d'effectuer le calcul des coefficients cepstraux non seulement sur ces sous bandes, considérées comme des spectres à part entière, mais également sur le spectre entier

et sur les re-division de ces sous-bandes etc... Le fait d'utiliser les différents sets de coefficients cepstraux dans le même *feature vector* confère à cette méthode le caractère *multi-résolution*. Les résultats escomptés de telle méthode ne sont pas miraculeux, mais ils peuvent aider à mieux décrire les phonèmes.

6.3 Limites

Toutefois, le développement d'un système de reconnaissance de phonèmes en temps réel ne peut se contenter de la classification hors contexte présentée dans ce document. Même si ce cette dernière était robuste et atteignait des taux de reconnaissance élevée, elle resterait bornée à reconnaître certaines classes de phonèmes, pourvus qu'ils soient bien prononcés. Il est impossible de prévoir ce que le système retournerait si on le confrontait à un phonème inconnu, comme une plosive. Ce défaut est une contrainte majeure, dans la mesure ou on souhaite pouvoir laisser fonctionner le système en continu, et non y introduire uniquement les phonèmes qu'il est susceptible de reconnaître.

Dans l'optique de développer un système applicable au spectacle vivant, il est donc essentiel d'étendre le système à la reconnaissance des autres classes de phonèmes. Or ces autres classes nécessitent un modèle avec suivi temporel. Le problème qui se pose alors est de savoir de quelle manière introduire les résultats du système actuel dans un modèle avec suivi temporel. Si notre système est capable pour chaque frame de signal de formuler ne réponse, comment évaluer la robustesse de cette réponse par rapport à celle donnée par les modèles temporels ? La solution peut être d'inclure dans les arbres de classification et pour les classificateurs de chaque noeud, la possibilité de dire qu'un phonème n'est pas reconnu par les classificateurs de ses feuilles. Le système mettrait alors en route ses modèles temporels afin de tenter de les reconnaître. On sait que certaines plosives sont composées de silences puis de bruits de friction, or chacun de ces éléments sont reconnaissables par le classificateur : un silence, puis une fricative...

Finalement, il semble qu'il est essentiel d'utiliser un modèle temporel afin de reconnaître efficacement tous les phonèmes. Le plus gros du travail à venir consiste dès lors à mettre en place un réseau d'autant de HMM qu'il existe de phonèmes, et de le rendre suffisamment peu coûteux en temps de calcul pour pouvoir le faire fonctionner en temps réel.

Conclusion

Au cours de ce stage, mes recherches se sont concentrées sur la reconnaissance de phonèmes en temps réel, instantanée et indépendamment du contexte. Un tel principe est principalement destiné au spectacle vivant, car les autres applications de la reconnaissance de la parole ne sont pas soumises à de telles contraintes. Ce stage a été l'occasion pour moi de mettre en pratique et d'enrichir des connaissances touchant à des disciplines diverses comme le traitement du signal, les statistiques, la reconnaissance de la parole, la phonétique... Les étapes de mon travail ont été difficiles à franchir, et beaucoup de retours en arrière on été nécessaires ; aussi certaines directions de recherches ont été abandonnées, comme les classificateurs SVM.

Durant une grande première moitié de ce stage, beaucoup de méthodes de classification ont été essayé (SVM, GMM, KNN...) mais sur des bases de données trop petites et comportant des erreurs de *labelling*. Des améliorations ont au fur et à mesure été ajoutées au système, comme le traitement pitch-synchrone ou l'utilisation du premier coefficient d'autocorrélation. Mais à l'écoute, les fragments composant les clusters d'apprentissage n'étaient jamais représentatifs du phonème à apprendre. L'enregistrement de la base de données à l'Ircam, s'il est arrivé tard, a au moins permis de présenter ces quelques résultats en temps voulu.

Aussi, il reste beaucoup de types de paramètres, de classificateurs à tester, mais surtout une bonne intégration de ces données dans un modèle temporel afin de trouver le système optimal maintenant que nous possédons une base de données fiable.

En revanche, la grande variabilité de la parole rend la reconnaissance de ses unités atomiques très difficile. Il semble que chercher à reconnaître les phonèmes indépendamment du contexte requiert une diction du locuteur extrêmement articulée ; sans quoi les phonèmes sont tellement fondus les uns dans les autres qu'il est impossible de les reconnaître même pour une oreille humaine. La reconnaissance de la parole courante par l'oreille humaine est un phénomène contextuel indépendant des phonèmes tels qu'on nous les apprend dans notre plus jeune âge.

Annexe

A Phrases phonétiquement équilibrées

Il se garantira du froid avec un bon capuchon.
[i l s 2 g a R a~ t i R a d y f R w a a v E k e~ b o~ k a p y S o~]

Annie s'ennuie loin de mes parents.
[a n i s a~ n H i l w e~ d 2 m e p a r a~]

Les deux camions se sont heurtés de face.
[l e d 2 k a m J o~ s 2 s o~ 9 R t e d 2 f a s]

Un loup s'est jeté immédiatement sur la petite chèvre.
[e~ l u s E j 2 t e i m m e d J a t 2 m a~ s y R l a p 2 t i t S E v R]

Dès que le tambour bat, les gens accourent.
[d e k 2 l 2 t a~ b u R b a l e j a~ a k u R]

Mon père m'a donné l'autorisation.
[m o~ p E R m a d o n e l o t o r i z a s J o~]

Vous poussez des cris de colère ?
[v u p u s e d e k R i d 2 k o l E R]

Ce petit canard apprend à nager.
[s 2 p 2 t i k a n a R a p R a~ a n a j e]

La voiture s'est arrêtée au feu rouge.
[l a v w a t y R s e t a R e t e o f 2 r u j]

La vaisselle propre est mise sur l'évier.
[l a v e s E l p R o p R E m i z 2 s y R l e v J e]

Bibliographie

[BBD+99] R Boite, H Bourlard, T Dutoit, J Hancq, and H Leich. Traitement de la parole. *Presses polytechniques et Universitaires Romandes, collection électricité*, 1999.

[BD97] H Bourlard and S Dupont. Sub-band based speech recognition. *Conference on Acoustics, Speech and Signal Processing*, 1997.

[BSHR05] Grégory Beller, Diemo Schwarz, Thomas Hueber, and Xavier Rodet. A hybrid concatenative synthesis system on the intersection of music and speech. In *Journées d'Informatique Musicale (JIM)*, MSH Paris Nord, St. Denis, France, June 2005.

[CAL89] CALLIOPE. *La Parole et son traitement automatique*. CNET-ENST, 1989.

[Com81] P Combescure. 20 listes de dix phrases phonétiquement équilibrées. *Revue d'acoustique*, 56, 1981.

[CY96] John E. Clark and Colin Yallop. *An Introduction to Phonetics and Phonology*. Blackwell, Oxford, 1996.

[dCK02] A de Cheveigné and H. Kawahara. Yin, a fundamental frequency estimator fo speech and music. *Acoustical Society of America*, 2002.

[DDR95] C D'Alessandro, P Depalle, and X Rodet. Machines à chanter. *Résonance*, 8, 1995.

[McC98] P McCourt. Multi-resolution cepstral features for phonème recognition across speech sub-bands. *Proceedings of ICASSP*, 1998.

[MG80] J.D. Markel and A.H. Gray. *Linear Prediction of Speech*. Springer, 1980.

[NWGD05] Noël Nguyen, S Wauquier-Gravelines, and J Durand. *Phonologie et phonétique : Forme et substance*, chapter La perception de la parole. 2005.

[Opp78] Alan V. Oppenheim, editor. *Applications of Digital Signal Processing*, chapter Digital Processing of Speech, pages 117–168. Prentice–Hall, 1978.

[Rab89] Lawrence R. Rabiner. A tutoriall on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77, 1989.

[RJ93] Lawrence R. Rabiner and Biing-Hwang Juang. *Fundamentals of Speech Recognition*. Prentice Hall, Englewood Cliffs, NJ, 1993.

[Sch98] Diemo Schwarz. Spectral Envelopes in Sound Analysis and Synthesis. Diplomarbeit Nr. 1622, Universität Stuttgart, Fakultät Informatik, Stuttgart, Germany, June 1998.

[SR99] Diemo Schwarz and Xavier Rodet. Spectral Envelope Estimation and Representation for Sound Analysis-Synthesis. Beijing, China, October 1999.

[Ste37] S.S Stevens. On hearing by electrical stimulation. *Journal of the Acoustical Society of America*, 8, 1937.

[Wel95] John C. Wells. Computer-coding the IPA : a proposed extension of SAMPA. Department of Phonetics and Linguistics, University College London, urlhttp ://www.phon.ucl.ac.uk/home/sampa/x-sampa.htm, April 1995.

[Wel03] John C. Wells. SAMPA for French. Web page, March 2003. Department of Phonetics and Linguistics, University College London, urlhttp ://www.phon.ucl.ac.uk/home/sampa/french.htm.