ÉCOLE DE TECHNOLOGIE SUPÉRIEURE
UNIVERSITÉ DU QUÉBEC

MÉMOIRE PRÉSENTÉ À
L'ÉCOLE DE TECHNOLOGIE SUPÉRIEURE

COMME EXIGENCE PARTIELLE
À L'OBTENTION DE LA

MAÎTRISE EN TECHNOLOGIE DES SYSTÈMES
M.Ing.

PAR
AZZOUZ BENLAHOUAR

NOUVELLES TECHNIQUES DE SEGMENTATION POUR CARACTÉRISER LE TIMBRE VOCAL D'UN LOCUTEUR EN VUE DE LA VÉRIFICATION AUTOMATIQUE DE L'IDENTITÉ.

MONTRÉAL, LE 27 JUIN 2003

CE MÉMOIRE A ÉTÉ ÉVALUÉ
PAR UN JURY COMPOSÉ DE :

M. Chakib Tadj, professeur et directeur de mémoire.
Département de génie électrique à
l'École de technologie supérieure.

M. Christian Gargour, professeur et président du jury.
Département de génie électrique à
l'École de technologie supérieure.

M. Rita Noumeir, professeure.
Département de génie de production automatisée à
l'École de technologie supérieure.

IL A FAIT L'OBJET D'UNE SOUTENANCE DEVANT JURY ET PUBLIC

LE 30 MAI 2003

À L'ÉCOLE DE TECHNOLOGIE SUPÉRIEURE

NOUVELLES TECHNIQUES DE SEGMENTATION POUR CARACTÉRISER LE TIMBRE VOCAL D'UN LOCUTEUR EN VUE DE LA VÉRIFICATION AUTOMATIQUE DE L'IDENTITÉ.

Azzouz Benlahouar

SOMMAIRE

Dans ce travail de recherche nous avons développé de nouvelles techniques de segmentation fréquentielle caractérisant l'empreinte vocale en vu de l'authentification de l'identité du locuteur. Nous avons proposé un nouvel algorithme MSAAB (Meilleure Structure d'Arbre ABstrait) impliquant une analyse en ondelettes et une analyse en composante principale. À partir d'un signal vocal d'un locuteur donné, nous pouvons déterminer les paramètres acoustiques appropriés qui lui permettent d'être discriminant. En utilisant l'algorithme MSAAB, nous avons effectué une série d'expériences pour la vérification de l'identité par la voix en mode texte-dépendent et en mode texte-indépendant. Durant nos expériences, nous avons utilisé deux types de corpus : Yoho, une base de données propre et Spidre, une base de données téléphonique bruitée. Les paramètres extraits sont utilisés comme une entrée du système de vérification d'identité. Celui-ci utilise une modélisation Markovienne. Les résultats obtenus ont été comparés avec d'autres types de paramètres, Fourier notamment. La robustesse des algorithmes proposés a pu être vérifiée et confirmée.

REMERCIEMENTS

Je tiens à remercier mon directeur de recherche, monsieur Chakib Tadj, professeur à l'école de technologie supérieure, pour le soutien technique et moral qu'il m'a accordé tout au long de cette recherche. Je le remercie aussi pour sa disponibilité, ses conseils judicieux, et son aide lors des travaux expérimentaux.

Enfin, j'aimerais remercier toute ma famille, tous mes amis de près et de loin.

TABLE DES MATIÈRES

LISTE DES TABLES

LISTE DES FIGURES

LISTES DES ABRÉVIATIONS ET SIGLES

a	Facteur de changement d'échelle.
ACP	Analyse en Composante Principale.
AL	Arbre par Locuteur.
ALN	Arbre par Locuteur relative à un Niveau.
ALNk	Arbre par Locuteur relative au Niveau k
ALN_t	Arbre ALN obtenu pour un taux de variance expliqué égal à t
AP	Arbre par phrase
BBS	Best Basis Select
CWT	Transformée en ondelettes continue.
DCT	Discret Cosine Transform
DWT	Transformée en ondelettes discrète.
FA	Pourcentage des fausses acceptations
FR	Pourcentage des faux rejets
HMM	Hidden Markov Model
MFCC	Mel Frequency Cepstal Coefficients
MFDWC	Mel Frequency Discret Wavelet Coefficients
MSAAB	Meilleure Arbre ABstrait.
SC	Pourcentage des scores
TVE	Taux de Variance Effectif
TVR	Taux de Variance Réel
VITD	Vérification de l'Identité en mode Texte-Dépendant
VITI	Vérification de l'Identité en mode Texte-Indépendant
c_n^j	Coefficients d'approximation à l'échelle j
d_n^j	Coefficients du détail à l'échelle j
C_j^p	Nœud de l'arbre admissible à l'échelle j et à la position p.
τ	Facteur de translation.

V_j	Espace d'approximation à l'échelle j
W_j	Espace de détail à l'échelle j
φ	Fonction d'échelle
ψ	Fonction d'ondelette

INTRODUCTION

Un système de reconnaissance de locuteur est l'authentification d'une empreinte vocale, c'est un processus de décision utilisant des caractéristiques du signal de parole pour déterminer des éléments d'information sur l'identité du locuteur à partir d'un signal prononcé. Il existe deux types de systèmes : le système de la vérification de l'identité en mode texte-dépendant (VITD), et le système de la vérification de l'identité en mode texte-indépendant (VITI). Le locuteur peut prononcer le même texte pour l'entraînement et pour la vérification s'il s'agit de VITD, alors que le système VITI ne tient pas compte du texte prononcé. Généralement les systèmes VITI sont plus naturels et moins performants que les systèmes VITD.

Les méthodes d'analyse acoustiques de la parole peuvent être classées en deux catégories: soit de type temporel, soit de type fréquentiel. Les systèmes actuels sont essentiellement fondés sur un pré-traitement par analyse de Fourier. Dans le cadre de ce travail nous étudierons 4 types d'analyses pour caractériser les traits acoustiques d'un signal audio. La première dite MFCC (Mel Frequency Cepstal Coefficients) [12], basée sur l'analyse de Fourier, et utilisant un banc de filtre à l'échelle de Mel. Ce banc de filtre est obtenus par modélisation de la perception auditive de l'oreille humaine. La deuxièmes méthode, appelée MFDWC (Mel Frequency Discret Wavelet Coefficients) [11]. C'est une méthode basée sur l'analyse en ondelettes, et utilisant un banc de filtre à l'échelle de Farooq. Cette échelle est une approximation linéaire de l'échelle de Mel [12]. La troisième méthode dite BBS (Best Basis Select) [1] est aussi basée sur l'analyse en ondelettes, mais utilisant un arbre admissible pour chaque locuteur. Cet arbre est obtenu par sélection de la meilleure base d'ondelettes selon le critère de l'entropie minimal [1,7]. L'application de ces 3 méthodes ne permet pas d'obtenir des caractéristiques appropriées et discriminantes pour un locuteur donné. La quatrième méthode représente la contribution de ce travail. Bien qu'elle soit similaire à la troisième, la principale différence réside dans la technique de sélection des nœuds. Un

nœud de l'arbre admissible est sélectionné selon son taux d'information qu'il porte. L'ensemble de ses nœuds est obtenu par application de l'Analyse en Composante Principale (ACP).

L'ACP est un outil puissant dont on fait appel dès qu'il s'agit d'une analyse statistique multivariable à dimension très élevée. Elle a été largement utilisée dans le domaine de la reconnaissance des formes, particulièrement pour l'apprentissage est l'analyse des images [5]. Son application dans le domaine de la parole est relativement rare. La plupart des applications de l'ACP dans le domaine de recherche de la parole ont pour objectif la modélisation des variabilités de la voix d'un locuteur causée par exemple par l'age, l'accent ou l'environnement. Kuhn [14] l'a appliqué au niveau de la modélisation. À partir d'un ensemble de paramètres acoustiques, chaque locuteur lui correspond un espace propre. Cette espace est obtenus par application de l'ACP sur un ensemble de données d'entraînement. C'est une représentation des 'voix propres' (eigenvoices) du locuteur obtenue par analogie aux images propres (eigenfaces). Cette dernière est utilisée pour la vérification et l'identification.

Dans ce travail nous allons utiliser l'ACP pour analyser chaque signal de parole dans ses bandes de fréquences où l'information est la plus pertinentes. Durant cette étude, nous allons utiliser d'une manière exclusive un pré-traitement impliquant la transformée en ondelettes et l'analyse en composante principale. Le signal vocal sera alors caractérisé par des paramètres explicites. Ces derniers seront utilisés comme paramètres d'entrée au système de reconnaissance basé sur une modélisation Markovienne.

Ce travail comporte une nouveauté dans le domaine de traitement du signal. Cette nouveauté réside dans la construction d'arbre admissible relatif à chaque locuteur. Ceci permet d'extraire d'un signal vocal d'un locuteur donné, les paramètres acoustiques lui permettant d'être discriminé de tous les autres locuteurs. Nous aurons ainsi extrait le timbre vocal caractéristique et propre de chaque individu. La structure de cet arbre est

différente de celles existant dans la littérature. Il est de type 'abstrait' du fait qu'il peut contenir un nœud et ceux qu'il peut engendrer.

Les grandes lignes de ce projet de recherche sont :
Chapitre 1 : donne une étude sur la Modélisation Markovienne que nous avons utilisé dans la partie entraînement et vérification.

Chapitre 2 : survole les principaux types d'analyse existant à savoir analyse de Fourier et analyse en ondelettes. Nous présenterons également un survol de l'analyse en composante principale, outil mathématique utilisé dans nos différents développements.

Chapitre 3 : présente notre contribution dans le domaine du traitement du signal, l'étude détaillée d'une nouvelle technique de segmentation fréquentielle. Une étude de fusion de l'information et également présentée.

Chapitre 4 : une série de simulations est menée pour tester les méthodes déjà existantes et celles que nous avons développées. Deux types de corpus sont utilisés : Yoho, une base de données propre enregistrée en studio et Spidre, une base de données téléphonique bruitée.

CHAPITRE 1

PARAMÉTRISATION DES SIGNAUX

Dans ce chapitre nous étudierons deux types de paramétrisation. La section 1.1 traite de la paramétrisation utilisant l'analyse en ondelette. La section 1.2 traite de celle utilisant l'analyse de Fourier. Comme notre contribution nécessite une analyse en composante principale (ACP), celle-ci est présentée à la fin de ce chapitre.

1.1 Analyse en ondelettes

1.1.1 Transformation continue en ondelette

La transformée en ondelette repose sur le fait que toute fonction s intégrable peut être exprimée sous la forme d'une somme de fonctions, les fonctions ondelettes, toutes issues d'une seule et d'une même fonction, l'ondelette mère ψ, de manière à ce que [20]:

$$\forall \tau \in R \text{ et } \forall a>0,\ \forall t \in R\ \ \psi_{a,\tau}(t)=\left(\frac{1}{\sqrt{a}}\right)\psi\left(\frac{t-\tau}{a}\right) \tag{1.1}$$

où :

τ est le facteur de translation, a est le facteur d'échelle, la fonction $\psi(t)$ appelée ondelette mère et doit vérifier les propriétés suivantes :

a) La condition d'admissibilité $$\int_{-\infty}^{+\infty}\frac{\Psi(\omega)}{|\omega|}d\omega=0, \tag{1.2}$$

b) La régularité et l'atténuation des n premiers moments de l'ondelette

$$\int_{-\infty}^{+\infty} t^n|\psi(t)|^2\,dt=0 \tag{1.3}$$

La transformation continue des ondelettes (continuous Wavelet transform CWT) d'une fonction $s \in L^2(R)$ est définie par les coefficients suivant :

$$CWT(a,\tau)=\langle s,\psi_{a,\tau}\rangle \tag{1.4}$$

$$CWTs(t)_{[a,\tau]} = C(a,\tau) = \frac{1}{\sqrt{a}} \int s(t)\psi(\frac{t-\tau}{a})dt \tag{1.5}$$

La transformée en ondelette de s(t) est la projection du signal s(t) sur la fonction ondelette $\psi_{a,\tau}(t)$. La Figure 1 est une illustration de cette opération.

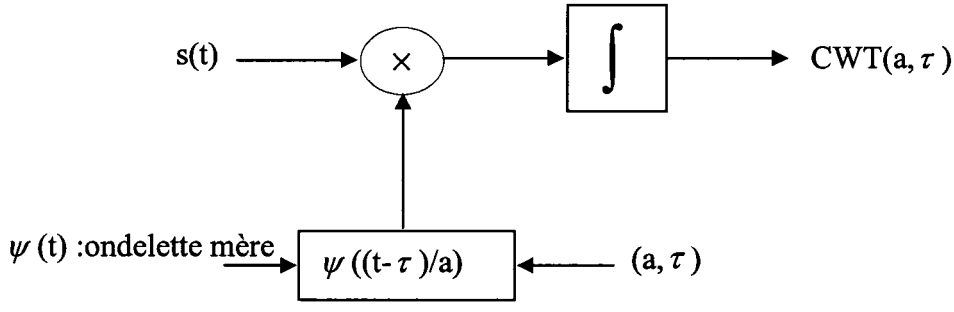


Figure 1 Transformée en ondelette d'un signal [20].

ce produit scalaire ($CWT(a,\tau)=\langle s,\psi_{a,\tau}\rangle$) exprime [20] :

- La corrélation du signal s(t) avec $\frac{\psi(\frac{t}{a})}{\sqrt{a}}$ lorsqu'il est décalé dans le temps d'un facteur égal à τ/a, et la similarité entre le signal s(t) et $\frac{\psi(\frac{t}{a})}{\sqrt{a}}$.
- Le filtrage de s(t) par un filtre de réponse impulsionnelle $\frac{\psi(-\frac{t}{a})}{\sqrt{a}}$ à l'instant τ/a.

Et avec le changement de variable $t'=\frac{t}{a}$ l'équation CWT devient :

$$CWTs(t)_{[a,\tau]} = C(a,\tau) = \sqrt{a}\int s(at)\psi(t'-\frac{\tau}{a})dt' \qquad (1.6)$$

Ce produit scalaire qui est lui aussi la corrélation du signal s(at) (s(at) est le signal s(t) lorsqu'il est dilaté dans le domaine temporel avec un facteur d'échelle égal à a) avec l'ondelette mère lorsqu'elle est décalée dans le temps de τ/a.

1.1.2 Transformée discrète en ondelettes

La transformation en ondelettes continues transforme une fonction à une dimension s(t) en une fonction à deux dimensions CWT(a, τ). Les variables a et τ sont continues. Pour des applications d'analyse du signal, on choisit de restreindre les valeurs des paramètres a et τ à une grille discrète. Dans ce cas on fixe un pas de dilatation $a_0 > 1$ et un pas de translation $\tau_0 \neq 0$ de la façon suivante [20]: $a=a_0^m$ *et* $\tau=n\tau_0 a_0^{-m}$, et la famille d'ondelettes est alors donnée par:

$$\psi_{m,n}(t)=a_0^{-\frac{m}{2}}\psi\left(a_0^{-m}t-n\tau_0\right) \qquad (1.7)$$

Pour réduire au maximum la redondance de cette représentation, on choisit des valeurs de a_0 et τ_0, typiquement : $a_0 = 2$ et $\tau_0=1$. Ainsi le facteur échelle varie de façon dyadique et les valeurs du couple (a, τ) sont représentées par une grille dyadique par la Figure 2. L'équation (1.7) devient

$$\psi_{m,n}(t)=2^{-\frac{m}{2}}\psi\left(2^{-m}t-n\right) \qquad (1.8)$$

Avec ce choix des valeurs du couple (a, τ) on parle de transformée en ondelette dyadique :

$$DWT(m,n) = \frac{1}{\left(\sqrt{2}\right)^{-m}}\int s(t)\psi(\frac{t}{2^m}-n)dt \qquad (1.9)$$

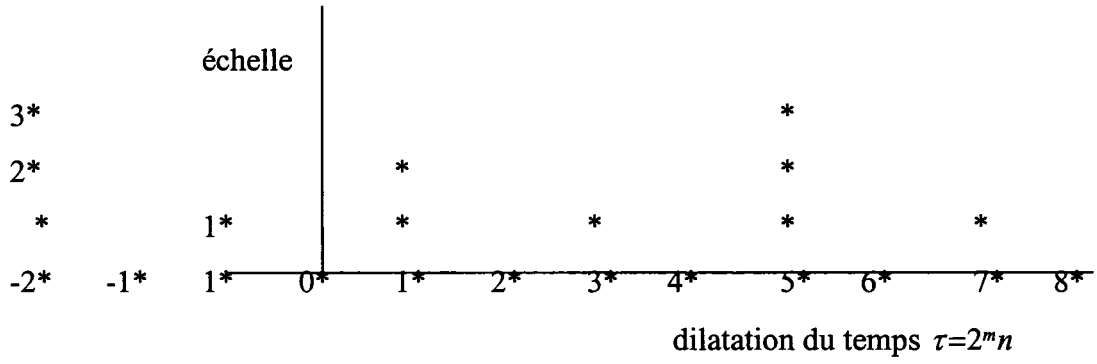


Figure 2 Grille d'échantillonnage dyadique .

1.1.3 Principe d'un banc de filtre à base d'ondelette

L'idée de banc de filtre est principalement utilisée pour décomposer un signal s(t) en plusieurs sous-bandes fréquentielles pour mieux le traiter et le transmettre. De façon générale, on peut décrire une procédure utilisant les ondelettes soit en parlant de la fonction ondelette et de la fonction d'échelle, soit en parlant des filtres h et g. Le motif de banc de filtre est illustré par la figure suivante [22].

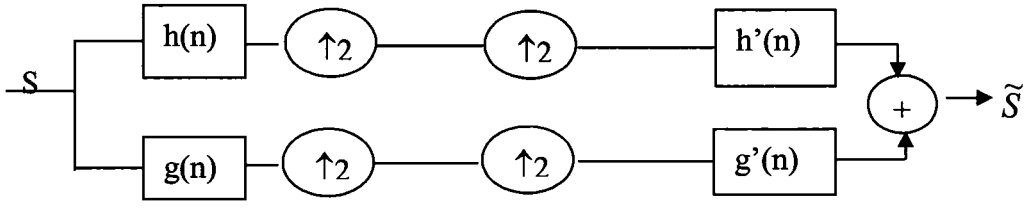


Figure 3 Banc de filtre à un seul étage.

Le signal s(n) est reconstruit avec un décalage de N échantillons, $\widetilde{S}(z)=S(z).z^{-N}$. Les filtres H et G sont des filtres miroir. Ils sont symétriques par rapport à $f_{\max}/2$ ($f_{\max}$ est la fréquence maximale du signal s(t)), comme le montre la figure suivante :

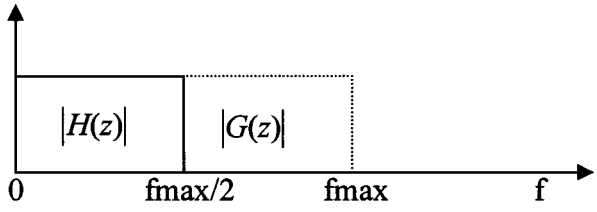


Figure 4 Deux filtres idéals sans chevauchement.

1.1.4 Analyse multirésolution

L'idée principale d'une analyse multirésolution est la décomposition d'un signal en sous-bandes . Deux principales notions pour expliquer les fondements de cette analyse sont l'échelle et la dilatation [17,20] . L'analyse multirésolution produit par filtrage et décimation successifs une série de signaux à une résolution de plus en plus faible. La Figure 5 donne un exemple d'analyse multirésolution, où le signal s(t) est divisé selon un ensemble de sous-bandes de fréquence.

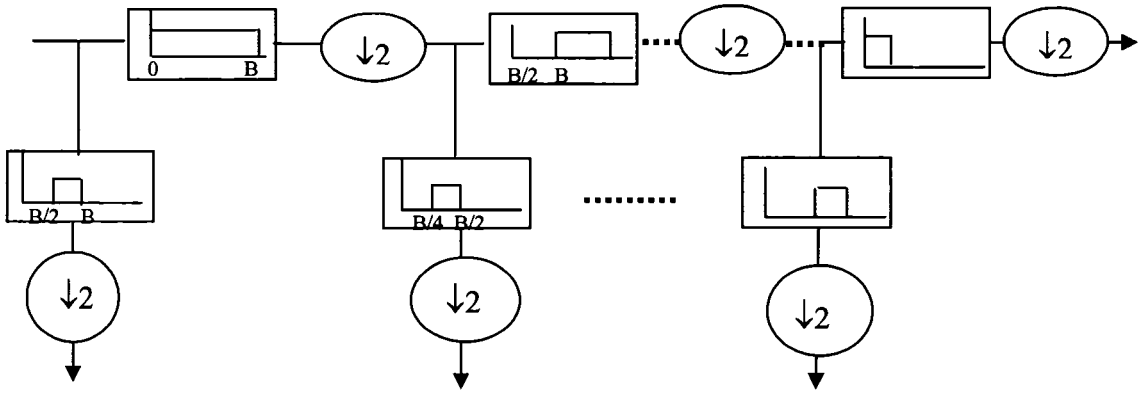


Figure 5 Décomposition de s(t) de spectre de fréquences compris entre 0 et B.

Le symbole ↓2 représente une décimation par 2.

Une définition plus théorique d'une analyse multirésolution est la suivante [8,17] :

Une suite de sous espaces fermés $\{V_i, i \in Z\}$ est une analyse multirésolution si les six propriétés suivantes sont vérifiées [8] :

1. $\forall (i,k)$, $s(t) \in V_i \quad \Leftrightarrow \quad s(t-2^i k) \in V_i$

2. $\forall i \in Z$, $V_{i+1} \subset V_i$

3. $\forall i \in Z$, $s(t) \in V_i \Leftrightarrow s(\frac{t}{2}) \in V_{i+1}$

4. $\lim_{i \to +\infty} V_i = \bigcap_{i=-\infty}^{+\infty} V_i = \{0\}$

5. $\lim_{i \to +\infty} V_i = Adherence\left(\bigcup_{i=-\infty}^{+\infty} V_i \right) = L^2(R)$

6. Il existe θ tel que $\{\theta(t-i)\}_{i \in Z}$ sera une base de Riesz de V_0

Rappel : Une base de Riesz [8] $\{\psi_i\}_i$ est une famille de fonctions dans un espace de Hilbert telle que :

$$\forall c_j \quad A\sum_j |c_i|^2 \le \left| \sum_j c_j \psi_j \right|^2 \le B\sum_j |c_j|^2 \text{ avec } 0 \le A \le B \le \infty \qquad (1.10)$$

Une base orthogonale est une base Riez, il suffit de choisir [8] A=B =1, en effet pour une famille de base orthogonale $\{\psi_i\}_i$, on a $\left| \sum c_i \psi_i \right|^2 = \sum_i |c_i|^2$.

$\{V_i, i \in Z\}$ est une multirésolution engendrée par une fonction échelle φ. W_i est le sous-espace complémentaires (complémentaire orthogonal) de V_i dans V_{i+1} : $V_{i+1} = V_i \oplus W_i$

La figure suivante montre une subdivision : $V_{i+1} = V_{i-1} \oplus W_{i-1} \oplus W_i$

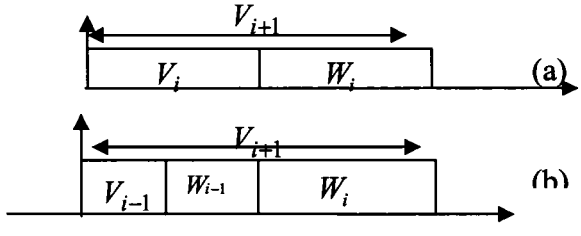


Figure 6 Subdivision de l'espace de Fourier en sous-espaces.

Le calcul des coefficients à différentes résolutions est obtenu en calculant les projections de la fonction signal s(n) sur les espaces d'approximation (V_i) et de détail (W_i) (voir paragraphe 1.1.5).

1.1.5 Coefficients de détail et coefficients d'approximation

Soit $\{V_i, i \in Z\}$ une multirésolution engendrée par une fonction échelle (V_i est l'espace engendré par la base $\left\{2^{\frac{i}{2}}\varphi(2^i n-k)\right\}, k \in Z$), et soit ψ une ondelette associé qui engendre les sous-espaces complémentaires. Les coefficients de détail (resp. coefficients d'approximations) d'un signal s(t) à un niveau d'échelle i (Figure 7) sont calculés de façon récursive [22] :

$$\begin{aligned} d_n^i &= \langle s, \psi_{i,n} \rangle \\ &= \sum_k g(2n-k) c_k^{i-1} \end{aligned} \qquad (1.11)$$

L'extraction du coefficient détail du signal au niveau de résolution j est obtenu par projection orthogonal du signal original s(t) sur l'espace complémentaire (W_i) de V_i sur V_{i+1} [22].

$$\begin{aligned} c_n^i &= \langle s, \varphi_{i,n} \rangle \\ &= \sum h(2n) c_k^{i-1} \end{aligned} \qquad (1.12)$$

Et la fonction échelle est définie telle que [22] :

$$\varphi(t) = \sum_{n \in Z} h(n) \varphi_{-1,n}(t) \qquad (1.13)$$

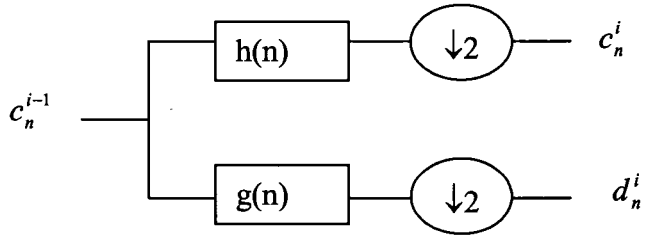


Figure 7 Calcul des coefficients détail et approximation à l'aide de l'algorithme de S. Mallat [22].

1.1.6 Décomposition en paquet d'ondelettes et recherche de meilleures structures d'arbre

L'algorithme de S.Mallat [22] permet d'obtenir une décomposition de type dyadique où seul les espaces d'approximation peuvent être redivisés. Ce type de décomposition permet une segmentation fréquentielle donnée par la figure 8.

Dans une décomposition en paquets d'ondelette (

Figure 9) aussi bien l'espace d'approximation que celui de détail est redivisé, et par cela on génère une multitude de bases. Cet ensemble de bases doit être organisé de la meilleure façon pour conduire à une série d'échantillons plus représentative de l'information portée par un signal donné.

Dans le cas général, et en terme de banc de filtre, chaque sous–bande (généralement appelé père) d'un niveau i engendre deux sous-bandes fils, et ses deux fils deviennent à leurs tour de nouveau père pour engendrer d'autre fils [15]. Le but de ces décompositions réside dans la possibilité d'un choix de la meilleure base.

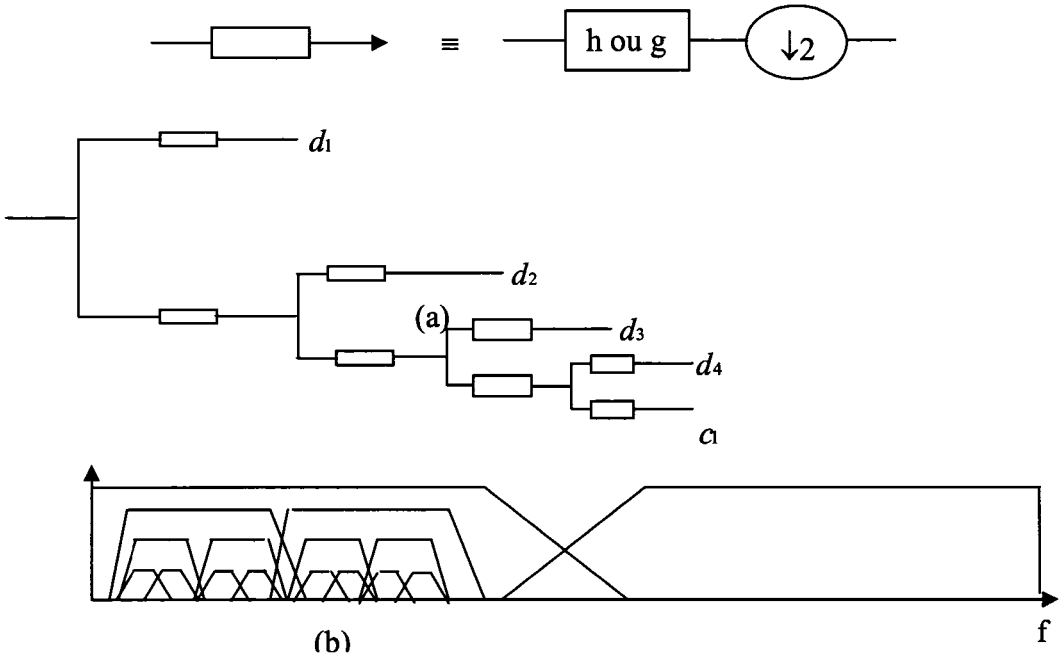


Figure 8 **a) Décomposition dyadique d'un signal, b) réponse en fréquence.**

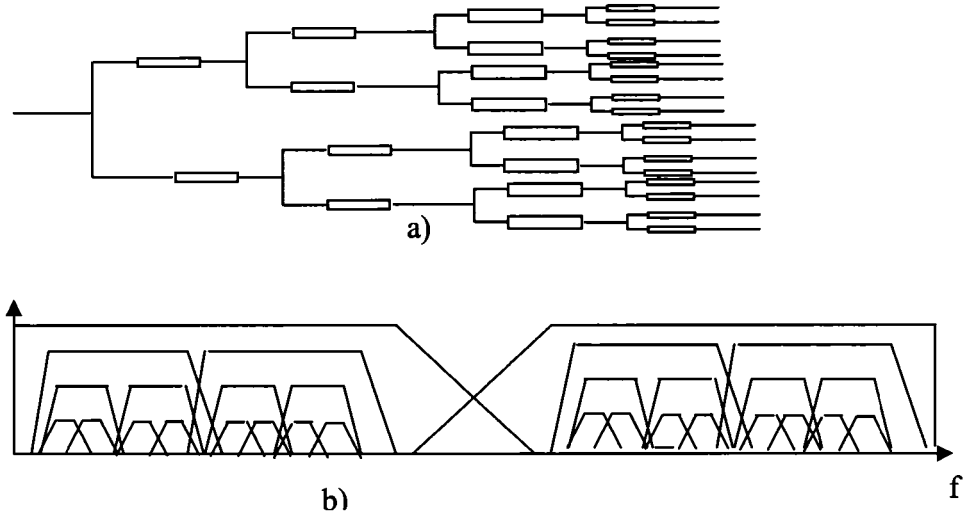


Figure 9 **a) Décomposition en paquet d'ondelettes de profondeur 4, b) segmentation fréquentielle correspondante.**

1.1.7 Algorithme proposé par Coifman pour une meilleure base

Un algorithme de la meilleure base d'ondelette proposé par Coifman repose sur une structure ou chaque sous-bande (père) est divisée en deux sous-bandes (deux fils). Une fonction coût est ajoutée pour le choix d'une base optimale. Le principe de cet algorithme peut être résumé en 4 étapes [7] :

a) Effectuer une décomposition jusqu'à une profondeur souhaitée.
b) Calculer la fonction coût pour chaque niveau de décomposition. Cette fonction utilise généralement le critère d'entropie, et en partant du bas (profondeur maximale) vers le haut.
c) Sélectionner des sous-bandes de la manière suivante : si la somme de l'entropie des deux fils d'un père est plus petite que celle de ce dernier, alors les deux fils sont retenus, et leur père est rejeté. Dans le cas contraire, c'est le père qui est retenu.
d) Augmenter la profondeur de l'arbre général et reprendre les tests b et c.

Les familles d'ondelettes sont abondement détaillées dans plusieurs livres. Le lecteur est invité à consulter le document [8] pour plus de détails.

1.2 Analyse de Fourier

Dans cette section, nous allons présenter la méthode MFCC [12] (Mel Frequency Cepstral Coefficients), de paramétrisation des signaux de parole. Cette méthode est fondée sur l'analyse de la transformée de Fourier. Alternativement au banc de filtre obtenu par analyse en ondelette, les coefficients MFCCs sont obtenus à partir d'un banc de filtres à l'échelle de Mel.

1.2.1 Banc de filtres à l'échelle de Mel

Un banc de filtre à l'échelle de Mel est un banc de filtre dont les fréquences sont reparties linéairement dans l'intervalle [0,1khz] et de façon logarithmique dans l'intervalle de fréquence [1,4khz]. La figure suivante illustre cette répartition utilisant un ensemble de filtres triangulaires. La fonction de cette échelle est [4] :

$$Mel(f) = 2595\log_{10}\left(1+\frac{f}{700}\right) \tag{1.14}$$

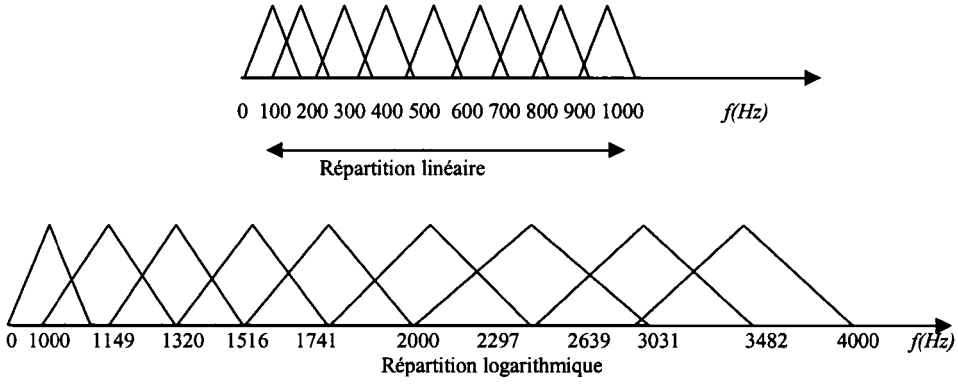


Figure 10 Type de banc filtre à échelle de Mel utilisant des filtres triangulaires.

1.2.2 Filtre de pré-accentuation

Indépendamment du type de paramétrisation, il est nécessaire de procéder à certaines opérations sur les fichiers audio avant de faire toute analyse. Un filtre de pré-accentuation est recommandé pour ne pas favoriser la représentation des basses fréquences par rapport aux hautes fréquences. Généralement ce filtre est du premier ordre, non récursif et sa fonction de transfert est donnée par [4]:

$$H(z)=1-kz^{-1} \tag{1.15}$$

k est le coefficient de pré-accentuation (0.97 dans notre cas) généralement de valeur comprise dans l'intervalle[0.9,1].

1.2.3 Fenêtres de Hamming

Le fenêtrage de Hamming est utilisé dans les deux types d'analyses : analyse en ondelette et analyse par la transformée de Fourier. Cette technique qui consiste à diviser le signal a pour effet [4]:

- D'analyser le signal à partir de segments successifs et stationnaires.
- D'éliminer les effets de bord.

La fonction de Hamming est définie comme suit :

$$z(n) = \left\{0.54 - 0.46\cos\left(\frac{2\pi(n-1)}{N-1}\right)\right\}_n, \qquad 0 \le n \le N \tag{1.16}$$

Le signal échantillonné est multiplié par la fenêtre de Hamming de façon périodique (de duré Df) avec une duré de recouvrement (Dr). Dans notre cas : Df=25ms et Dr=15ms.

1.2.4 Coefficients MFCC

Avant tout traitement, le signal est échantillonné selon le théorème de Nyquist. Les paramètres MFCC sont calculés pour chaque période de la fenêtre de Hamming. Le bloc diagramme de la Figure 11 représente un système d'extraction de ces paramètres [12].
Pour chaque fenêtre d'analyse on calcule son énergie. Seules les énergies qui correspondent au banc de fréquences à l'échelle de Mel sont retenues.

Soit N_{mfcc} le nombre de coefficient MFCC (figure 11), N_{bf} le nombre de banc de filtre et $S_f(n)$ le signal obtenu après fenêtrage de Hamming. Le cofficient MFCC correspondant au filtre i du banc de filtre utilisé est donnée par la relation suivante [4]:

$$MFCC(i) = \sqrt{\frac{2}{N_{bf}}} \sum_{j=1}^{Nbf} mfb(j)\cos\left(i\left(j-\frac{1}{2}\right)\frac{\pi}{N_{bf}}\right), i = 1,\ldots,N_{mfcc} \tag{1.17}$$

mfb(i) est le vecteur des logarithmes des énergies des signaux qui correspondent aux sorties du banc de filtre i.

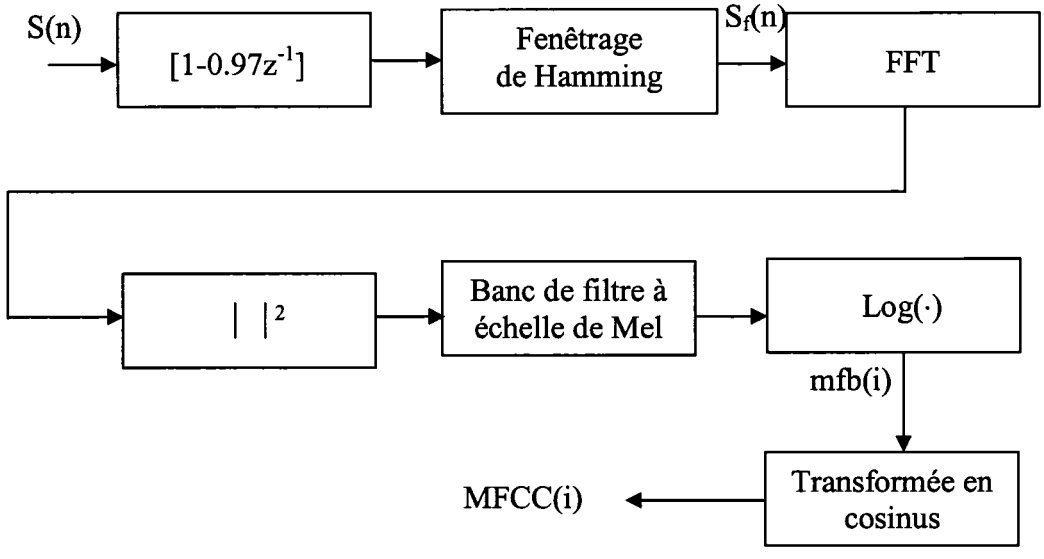


Figure 11 Bloc diagramme d'extraction des paramètres MFCC.

À partir d'une décomposition en paquets d'ondelettes, et Alternativement à l'algorithme proposé par Coifman [7], nous allons appliquer l'Analyse en Composante Principale (ACP) pour sélectionner les meilleurs nœuds de l'arbre admissible. La section 1.3 présente un brève aperçue sur la théorie de l'ACP, et le chapitre 3 présente la technique son application

1.3 Analyse en composante principale

L'Analyse en Composante Principale (ACP) est une méthode statistique classique. Elle a été largement répandue dans l'analyse et la compression de données [10]. L'analyse en composante principale est fondée sur la représentation statistique d'une variable aléatoire. Supposons que nous ayons un vecteur X qui représente une population de N variables aléatoires, $X=[x_1,x_2,...,x_N]$.

Cette population est de moyenne :

$$\mu_x=E\{(X)\} \tag{1.18}$$

et de covariance

$$C=E\{(X-\mu_x)\times(X-\mu_x)^T\} \tag{1.19}$$

Les composantes de C, notées c_{ij}, représentent les covariances entre les composantes des variables aléatoires x_i et x_j. Si ces deux composantes sont non corrélées, leur covariance est zéro. Cela signifie que la variable x_i porte de l'information absente dans celle de x_j. Par contre si ces deux variables corrèlent bien entre elles, elles contiennent pratiquement la même information.

1.3.1 Calcul des vecteurs et valeurs propres d'une matrice

Les valeurs propres (λ_i) d'une matrice C sont la solution de l'équation caractéristique suivante :

$$\det(C-\lambda_i I)=0 \tag{1.20}$$

I est la matrice identité de même dimension que C et det(.) est le déterminant d'une matrice.

Les vecteurs propres sont déterminés par la résolution de l'équation suivante

$$(C - \lambda_i I)Vp_i = 0 \tag{1.21}$$

Vp_i est le vecteur propre correspondant à la valeur propre λ_i de C.

$\{u_i\}$ sont les vecteurs propres Vp_i normalisés données par :

$$u_i = \frac{Vp_i}{\sqrt{Vp_i^T Vp_i}} \tag{1.22}$$

1.3.2 Sélection des variables

Par classification des vecteurs propres dans l'ordre des valeurs propres descendantes (première est la plus grande), on peut créer une base orthogonale. Le premier vecteur propre de cette base correspond à la première composante principale. Celle-ci contient les informations relatives à la variance maximale. Le deuxième vecteur propre contient les informations relatives à la variance suivante et ainsi de suite.

Les principales étapes de l'application de l'ACP se résument comme suit [6] :

- Calcul de la matrice de covariance et de corrélation.
- Calcul des vecteurs et valeurs propres de la matrice de corrélation.
- Définition d'un taux de variance expliquée.

Le taux de variance expliquée par les k premiers vecteurs propres sélectionnés est :

$$T_k = \frac{\sum_{i=1}^{k} \lambda_i}{\sum_{i=1}^{N} \lambda_i} \tag{1.23}$$

- Mise à l'échelle des valeurs propres tel que :

$$v_i = u_i \times \sqrt{\lambda_i} \tag{1.24}$$

- Sélection des variables à éliminer

La sélection des variables à éliminer est faite selon les critères suivants :

1- Une variable est associée à un élément des vecteurs v_i

2- Elle est choisie dans le vecteur colonne v_i ayant la plus petite valeur propre.

3- Elle est représentée par le plus grand élément du vecteur v_i choisi.

Le schème suivant résume les étapes de l'application de l'ACP.

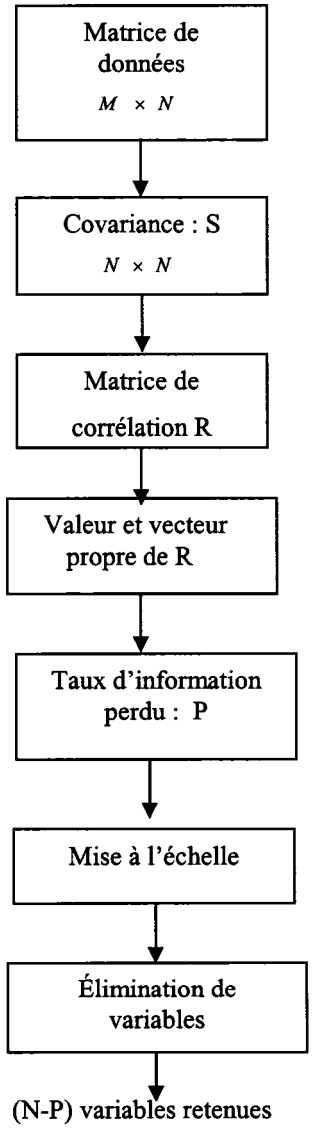


Figure 12 Principales étape de l'application de l'ACP.

1.4 Conclusion

L'analyse par ondelette est un outil mathématique qui nous permet de représenter un signal temporel dans un nouveau domaine 'domaine de la transformée en ondelette'. Cette représentation permet une analyse temps-échelle alternativement à l'analyse de Fourier qui permet elle aussi une analyse temps-fréquence. Dans ce travail nous allons exploiter les propriétés de dilatation et d'échelle en impliquant la transformé en paquet d'ondelettes et l'analyse en composante principale. La technique qui permet cela est détaillée dans le chapitre 3.

Les paramètres obtenus par une analyse basée sur les ondelettes et (ou) par Fourier vont nous servir de paramètres d'entrée du système de modélisation Markovienne décrite dans le chapitre suivant.

CHAPITRE 2

IDENTIFICATION ET VERIFICATION DE LOCUTEUR

2.1 Introduction

La problématique de la reconnaissance de locuteur se pose comme suit : quelle est la meilleure façon de modéliser une entité (l'entité dans notre cas est le phonème) représentative d'un signal vocal?

Dans ce travail nous utilisons des modèles statistiques, précisément les modèles de Markov cachés (Hidden Markov Model, HMM). Nous allons exploiter les modèles HMM pour concevoir deux systèmes principaux: un système pour vérifier l'identité d'un locuteur à partir de son message vocal, et un système d'identification pour connaître l'appartenance d'un message vocal. Ces deux systèmes peuvent commettre deux types d'erreur : des faux rejets et des fausses acceptations. Pour la vérification, un faux rejet se produit lorsque le système rejette un utilisateur qui tente d'accéder à ses propres ressources. Une fausse acceptation se produit lorsque le système accepte un imposteur qui tente d'accéder à des ressources qui ne lui appartiennent pas. En identification, un faux rejet se produit lorsque le système rejette un utilisateur membre d'un registre, et une fausse acceptation se produit lorsque le système accepte un non membre du registre.

La décision d'accepter ou de rejeter un locuteur est basée sur la similitude entre les paramètres d'un mot de test prononcé par celui-ci et le modèle d'un locuteur connu par le système. Cette décision dépend de ce qu'on appelle le seuil.

Dans ce chapitre nous donnerons un bref aperçu des modèles de Markov, ensuite nous présenterons les deux critères suivants : Critère ML (Maximum Likelihood), et le Critère MAP (maximum a posteriori) qui sont utilisés dans l'algorithme de Baum-Walech pour l'estimation des paramètres d'apprentissage du modèle HMM. Nous présenterons par la suite quelques techniques de seuillage.

2.2 Modèles de Markov cachés (HMM)

Un modèle HMM est un processus doublement stochastique au sens où il est constitué d'un processus stochastique sous-jacent qui n'est pas observable directement (il est caché), il ne peut être observé qu'à travers un autre ensemble de processus stochastiques, qui produisent la séquence de symboles observés. Un modèle HMM est gouverné par deux types de probabilités [19].

- Une probabilité de changement d'état $\{a_{ij}\}$, i et j sont deux indices qui représentent les états i et j.
- Une probabilité d'émission de symboles $\{b_j(o_i)\}$: distribution de la probabilité de l'observation o_i quand le système est dans l'état j.

La Figure13 montre un exemple de modèle HMM à six états qui génère la séquence o_1, o_2, o_3, o_4,o_5 o_6.

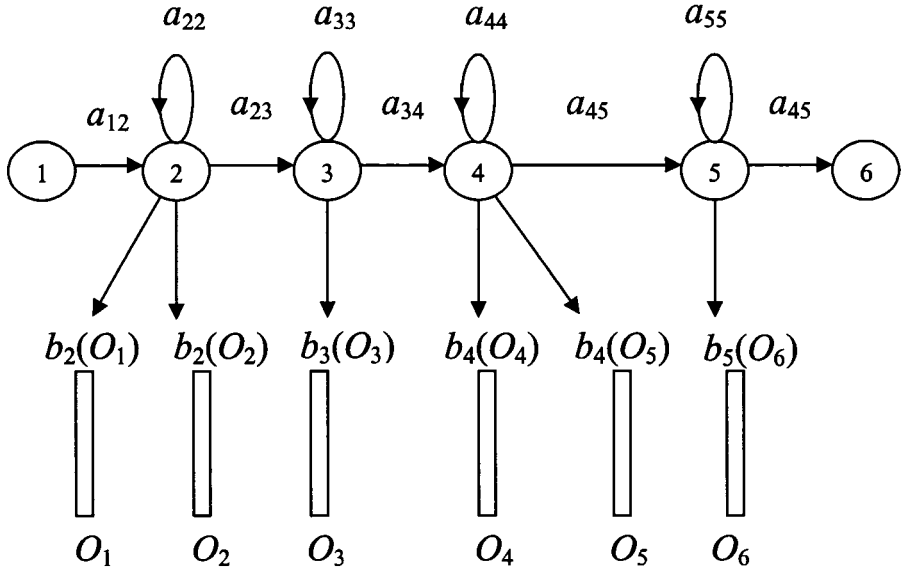


Figure 13 Modèle de Markov caché à 6 états [23].

2.3 Paramètres du modèle HMM

Pour identifier un modèle HMM, on utilise la notation compacte suivante : $\lambda = (A, B, \Pi)$ où A, B, Π sont les composantes essentielles correspondant à un modèle à N états qui sont définis par Rabiner et Juang comme suit [19] :

1. N est le nombre d'états estimé pour le modèle, dans notre cas nous allons utiliser un modèle à 5 états pour modéliser un phonème. Parmi ces 5 états, deux sont non émetteurs, celui de l'entrée, et celui de la sortie.
2. M est le nombre de symboles observables dans chaque état. L'ensemble des observations est noté V= $\{v_1, v_2, \ldots, v_M\}$. Un élément O_t de V désigne un symbole observé à l'instant t.
3. A est la matrice de probabilités de transition entre les états A=$\{a_{ij}\}$.
4. B est la matrice de probabilité d'observation des symboles dans chacun des états du modèle : $b_j(k)$ représente la probabilité que l'on observe le symbole v_k alors que le modèle se trouve dans l'état j, soit :

$$b_j(k) = P(O_t = v_k / q_t = j) \quad 1 \le j \le N \,, 1 \le k \le M,$$
$$b_j(k) \ge 0 \quad \forall j,k \; et \sum_{k=1}^{M} b_j(k) = 1; \tag{2.1}$$

5 $\Pi = \{\pi_i\}$, l'ensemble des probabilités initiales.

$$\pi_i = P(q_1 = s_i) \quad 1 \le i \le N \quad (\, \pi_i \ge 0 \; \forall i \quad , \text{et} \sum_{i=1}^{N} \pi_i = 1) \tag{2.2}$$

2.4 Estimation du modèle HMM

Soit λ un modèle de Markov caché qui représente un locuteur, et $O = \{o_1, o_2, ..., o_T\}$ une séquence d'observation. Notre but est de déterminer les paramètres du modèle HMM ($\lambda = (A, B, \pi)$)) qui maximise la probabilité P(O/ λ).

L'idée de l'apprentissage est d'utiliser des procédures de re-estimation qui affinent le modèle petit à petit en suivant les étapes suivantes [3,19] :

- Choisir un modèle initial λ_0 ;
- Calculer λ_1 à partir de λ_0 ;
- Répéter ce processus jusqu'à ce qu' un critère d'arrêt soit satisfait.

λ_n est le modèle optimal, il doit vérifier :

$$\prod_r P(O_r / \lambda_{n-1}) \leq \prod_r P(O_r / \lambda_n) \tag{2.3}$$

λ_n doit améliorer la probabilité d'émission, ce qui revient à définir une fonction F telle que λ_n=F(λ_{n-1}).

Pour l'estimation d'un modèle (entraînement des paramètres $\lambda=\{A,B,\Pi\}$), il est nécessaire de choisir un des deux critères suivants :

- Critère ML (Maximum Likelihood) : maximum de vraisemblance.
- Critère MAP (Maximum a posteriori) : maximum *a* posteriori.

Ces critères ont été abondement étudiés dans la littérature (voir [18,19]).

2.5 Spécification des probabilités de sorties

Les paramètres utilisés pour générer les modèles HMM sont des données multiples qu'on appelle les trames de données et qui sont supposés être statistiquement indépendants. Nous utiliserons également des composantes appelées mixtures que nous pouvant qualifier comme un sous-processus généré par chacun des états avec un poids égal à sa probabilité de transition.

La figure suivante montre un exemple d'un état à M mixture :

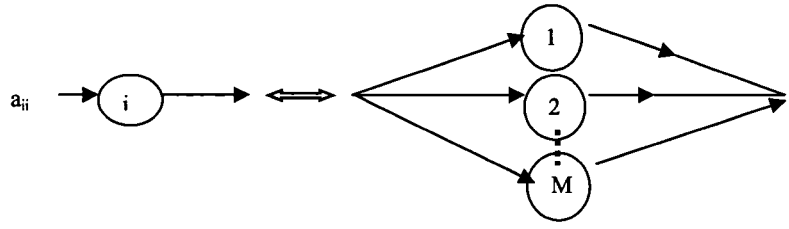


Figure 14 Mixture Gaussienne à M composantes [23].

Une distribution de probabilité commune pour la plupart des modèles HMM est la GMD (Gaussien Mixture Density) donnée par [2] :

$$b_j(o_t) = \prod_{s=1}^{S} \left[\sum_{m=1}^{M_S} c_{jsm} N(o_{st}; \mu_{jsm}, \Sigma_{jsm} \right]^{\gamma_S} \tag{2.4}$$

M_s, est le nombre de mixtures utilisées dans S (stream). C_{jsm} est le poids de la $m^{ème}$ mixture. $N(o;\mu,\Sigma)$ est une distribution gaussien avec un vecteur moyenne égale à μ et une matrice de covariance Σ définie par :

$$N(o,\mu,\Sigma) = \frac{1}{\sqrt{(2\pi)^n |\Sigma|}} e^{-0.5(o-\mu)^T \Sigma^{-1}(o-\mu)} \tag{2.5}$$

où n est la dimension du vecteur o, et γ_s est le poids du vecteur paramètre S.

2.6 La méthode de Baum –Welch pour la re-estimation des paramètres b_{ij}

Dans ce paragraphe nous allons utiliser la technique de Baum-Welch [2], aussi appelée algorithme EM (Expectation-Maximisation) pour résoudre le problème d'estimation des paramètres de façon itérative. Le problème essentiel est d'estimer la valeur de la moyenne (μ) et de la variance (Σ) de la distribution de probabilité de sortie $b_j(o_t)$ de chaque observation (o_t) générée par le modèle HMM.

Dans un modèle HMM à un seul état, cette estimation est facilement calculée. En effet l'estimation des paramètres μ_j et Σ_j de l'équation (2.5) est obtenue par une simple moyenne :

$$\hat{\mu}_j = \frac{1}{T}\sum_{t=1}^{T} o_t \tag{2.6}$$

et

$$\hat{\Sigma}_j = \frac{1}{T}\sum (o_t - \mu_j).(o_t - \mu_j)' \tag{2.7}$$

Les vecteurs observation d'entraînement sont divisés d'une façon égale à travers les états du modèle pour obtenir les valeurs initiales de la moyenne et de la covariance qui correspondent à chaque état. Par la suite, l'algorithme de Viterbi est utilisé pour trouver la séquence d'états qui maximise la probabilité de vraisemblance, et un autre ré-assignement que le précédent est utilisé pour avoir d'autres valeurs initiales meilleures. Ce processus est répété jusqu'à l'obtention de valeurs stables [23].

Vu que la vraisemblance totale pour chaque séquence est basée sur la somme de toutes les séquences d'états possibles, chaque observation o_t contribue dans le calcul des valeurs des paramètres de chaque état correspondant au maximum de vraisemblance, pour cela au lieu de l'approximation précédente, chaque observation est assignée à chaque état avec un poids égal à la probabilité d'émission de cet état par le modèle. Par conséquent, si $L_j(t)$ est la probabilité d'être à l'état q_j à l'instant t, l'équation de la moyenne et celle de la variance sont définies par Baum-Welch comme suit :

$$\hat{\Sigma}_j = \frac{\sum_{t=1}^{T} L_j(t)(o_t - \mu_j).(o_t - \mu_j)'}{\sum_{t=1}^{T} L_j(t)} \tag{2.8}$$

$$\hat{\mu}_j = \frac{\sum_{t=1}^{T} L_j(t) o_t}{\sum_{t=1}^{T} L_j(t)} \tag{2.9}$$

Les valeurs de $L_j(t)$ sont calculées par l'algorithme avant-arrière (Forward-Backward). Cet algorithme est une approche qui repose sur le calcul de la fonction forward

$\alpha(t,q_j)$ qui représente la probabilité d'observer les t premières observations et d'être à l'instant t à l'état q_j, et la fonction backward $\beta(t,q_j)$ qui représente la probabilité d'observer les $(T-t)$ dernières observations sachant qu'on était à l'instant t à l'état q_j. Ces deux fonctions se calculent par récurrence sur le temps de la manière suivante :

Calcul de α :

On a:

$$\alpha(t,q_j) = \alpha_t(j) = P(O_1 O_2 ... O_t, (q_t = s_j / \lambda)) \tag{2.10}$$

État initial

$\alpha_1(1) = 1$

Itération j

$$\alpha j(1) = a_{1j} b_j(o_1) \quad 1 \le j \le N \tag{2.11}$$

Terminaison

$$\alpha_N(T) = \sum_{i=2}^{N-1} \alpha_i(T) a_{iN} \tag{2.12}$$

Backward probabilité β :

État initial.

$$\beta_i(t) = a_{iN} \quad 1 \le i \le N \tag{2.13}$$

Condition finale

$$\beta_1(1) = \sum_{j=2}^{N-1} a_{1j} b_j(o_1) \beta_j(1) \tag{2.14}$$

Induction

$$\beta_i(t) = \sum_{j=2}^{N-1} a_{ij} b_j(o_{t+1}) \beta_j(t+1) \tag{2.15}$$

Par définition de la probabilité avant-arrière :

$$P(O, q_t = j/\lambda) = \alpha_j(t)\beta_j(t)$$

Par conséquent,

$$\begin{aligned} L_j(t) &= P(q_t = j/O, \lambda) \\ &= \frac{P(O, q_t = j/\lambda)}{P(O/\lambda)} \\ &= \frac{1}{P(O/\lambda)}\alpha_j(t)\beta_j(t) \end{aligned} \tag{2.16}$$

Les étapes de l'algorithme de E.M sont utilisées pour mieux ré-estimer les paramètres d'un modèle HMM,et peuvent être résumés comme suit :

1 Stockage des valeurs du numérateur et du dénominateur des deux équations définissant la valeur moyenne et la covariance pour chaque observation o_t. Le stockage est fait par un accumulateur.

2 Calculer $\alpha_j(t)$ et $\beta_j(t)$ pour tous les états j aux temps t.

3 Pour chaque état q_j à l'instant t, utiliser les probabilités Lj(t) et le vecteur d'observation courant o_t pour initialiser les accumulateurs pour cet état.

4 Utiliser les valeurs finales de l'accumulateur pour calculer les nouvelles valeurs des paramètres (μ et Σ).

5 Si la valeur de P (O/λ) qui correspond à la nouvelle itération est supérieure à celle de l'ancienne itération, alors on répète à nouveau le processus. Sinon le processus s'arrête.

6 Répéter les étapes 2 et 3 pour toutes les séquences d'entraînement qui sont disponible, jusqu'à l'obtention des valeurs stables de la moyenne et de la variance.

La figure suivante illustre les étapes de re-estimation des paramètres d'un modèle HMM.

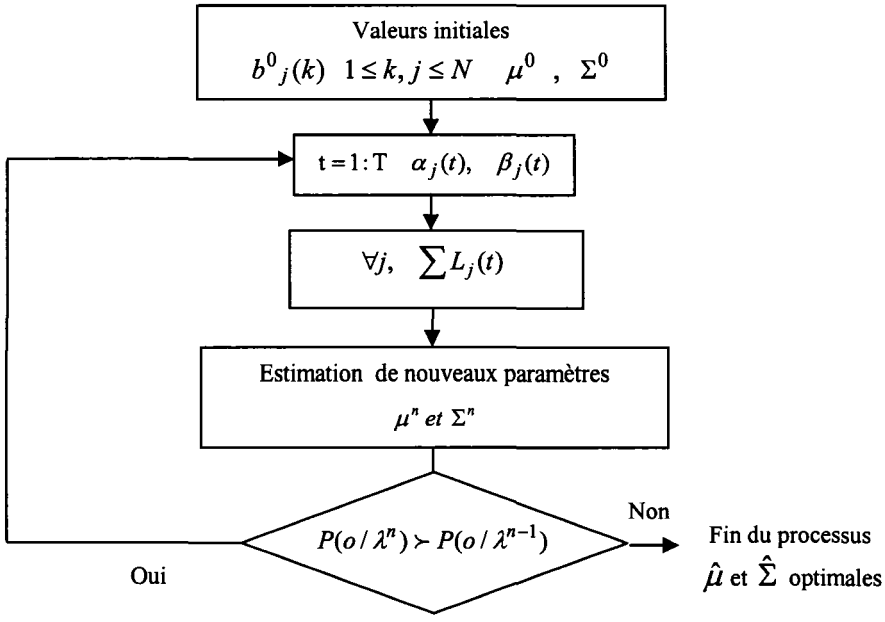


Figure 15 Algorithme E.M. pour la ré-estimation des paramètres de la probabilité de sortie.

2.7 Identification de locuteur utilisant le critère ML

Pour une séquence O ={ o_1, o_2,...o_T} donnée, correspondant à un locuteur appartenant à un registre, on devra identifier le vrai locuteur qui en est à l'origine. En terme stochastique, on doit trouver le modèle λ_l qui maximise P(λ_l/O). Cette probabilité peut être calculée en appliquant le théorème de Bayes :

$$P(\lambda_l \backslash O) = \frac{P(O/\lambda_l).P(\lambda_l)}{P(O)} \tag{2.17}$$

Le modèle optimal correspondant au vrai locuteur est :

$$\hat{\lambda} = \arg \max_{\lambda_l} \frac{P(O/\lambda_l).P(\lambda_l)}{P(O)} \tag{2.18}$$

La fonction arg est une fonction qui accumule les indices correspondant à la valeur maximale de la probabilité P (λ_i/O). En se servant de l'hypothèse suivante : la distribution de la probabilité *a priori* P(λ_i) est supposée être inconnue mais constante (locuteurs équiprobables), et la probabilité P(O) est indépendante du modèle HMM (λ), l'estimation de la valeur optimale $\hat{\lambda}$ est obtenue seulement par le calcul des probabilités de vraisemblance) : $\hat{\lambda} = \arg\max_{\lambda_i} P(O/\lambda_i)$ .Par la suite, le modèle optimal $\hat{\lambda}$ est celui qui correspond à la valeur maximale de la fonction de vraisemblance P(O/λ_i). C'est ce qu'on appelle la règle de décision du maximum de vraisemblance (voir la figure suivante). En pratique, on utilise le log P(O/λ_i) au lieu de P(O/λ_i).

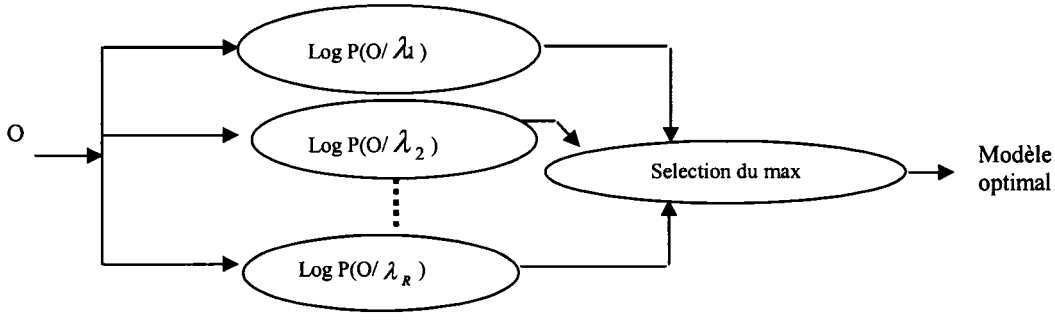


Figure 16 Diagramme bloc de l'identification de locuteur utilisant le maximum de vraisemblance.

2.8 Vérification du locuteur

La vérification du locuteur consiste à s'assurer que le locuteur qui tente d'accéder à ses ressources n'est pas un imposteur. La vérification de son identité est accomplie en établissant la similitude entre sa parole (phrase de test) et le modèle du locuteur désiré (modèle déjà connu). Si cette similitude est suffisamment grande, le locuteur est accepté. Sinon il est rejeté. Cette règle de décision est exprimée par [19] :

$$D = \begin{cases} A \ si \quad P(O_c / \lambda_d) \ / P(\lambda) \ \geq S \\ \\ R \qquad siP(O_c / \lambda_d) \ / P(\lambda) < \ S \end{cases} \tag{2.19}$$

λ_d est le modèle développé pour le locuteur propriétaire des ressources auxquelles l'utilisateur désire accéder. A et R désignent respectivement l'acceptation et le rejet, et S est le seuil ou la valeur minimale de [$P(O_c/\lambda_d)/P(\lambda)$] nécessaire pour que le requérant puisse être accepté. $P(\lambda)$ est la probabilité de normalisation. Généralement elle est égale à celle du modèle globale. Ce dernier est décrit dans le paragraphe suivant.
Dans les sections suivantes nous allons discuter de quelques techniques de seuillage appelées aussi : techniques de normalisation.

2.9 Techniques de seuillage

2.9.1 Normalisation par le modèle global

Cette technique consiste à comparer la probabilité de reconnaître un locuteur à celle de la probabilité de reconnaître le modèle globale étant donné un seuil S. Généralement ce seuil est égal à 0. Le modèle global peut être obtenu de deux façons :

a) Le modèle global est généré à partir d'un nombre de locuteurs p qui sont choisis parmi l'ensemble total de N locuteurs.
b) Le modèle global est généré à partir de tous les locuteurs.

La règle décision sera comme suit :
Si $P(O_t / \lambda_d) - P(O_c / \lambda_g) \succ 0$, le locuteur est accepté, si non il est refusé. Le système est indifférent si $P(O_t/\lambda_d)=P(O_c/\lambda_g)$

2.9.2 Normalisation par fraction de vraisemblance

L'établissement d'un seuil unique et stable n'est pas possible à cause des variations intra-locuteur. Higgins et al [12] propose une méthode de score appelée : score à fraction de vraisemblance. Pour l'obtenir, le signal d'entrée correspondant à un locuteur est comparé avec le modèle des autres locuteurs (c'est comme si cette phrase est supposée être produite par un imposteur). Ce ratio de vraisemblance est exprimé par :

$$Log(L(O_c)) \;=\; Log\frac{P(O_c / \lambda = \lambda_c)}{P(O_c / \lambda \neq \lambda_c)}) \qquad (2.20)$$

$Log\,P(O_c/\lambda_c{\neq}\lambda)$ est le terme de normalisation qui représente la vraisemblance de l'observation d'entrée et un imposteur. Généralement, le locuteur est accepté si la valeur du Log(L(Oc)) est positive.

2.9.3 Normalisation par cohorte d'un locuteur

On suppose que nous disposons de références représentatives de tous les locuteurs. Une décision statistique optimale peut être exprimée comme suit :

$$Log(L(O_c)) \;=\; LogP(O_c / \lambda = \lambda_c) - \max_{\lambda \in\, ref} Log(P(O_c / \lambda \neq \lambda_c)) \qquad (2.21)$$

Le nombre des locuteurs membres de la cohorte devient simplement égal au nombre de locuteurs les plus représentatifs (locuteurs référence (ref) pour le locuteur requérant). Pour une cohorte de dimension K, le score de normalisation est le suivant :

$$Log(O_c / \lambda \neq \lambda_c) = Log(\frac{1}{K}\sum_{S=1}^{K} P(O_c / \lambda_S) \qquad (2.22)$$

S est un membre de la cohorte du locuteur en question. Pour un test de vérification de locuteur, nous devons calculer tous les scores de vraisemblance pour tous les modèles des locuteurs membres de la cohorte. Le score de normalisation désiré sera leur moyenne.

Avec la méthode de cohorte basée uniquement sur le score de vraisemblance, on attribue à chaque locuteur du registre un ensemble de locuteurs global distinct. Mais le fait de supposer que tous les imposteurs sont des imitateurs, ou produisent du son comme celui du locuteur en question, la sélection par similarité devient inefficace et le système devient vulnérable aux imposteurs [18].

2.10 Conclusion

Ce chapitre nous a permis de nous familiariser avec les techniques utilisées dans les applications de reconnaissance et de vérification de l'identité par la voix.

Le chapitre suivant présente les principales contributions de notre recherche. De nouvelles méthodes sont proposées pour pré-traiter le signal vocal, et rendre ainsi les systèmes de reconnaissance du locuteur plus robustes.

CHAPITRE 3

PRINCIPALES CONTRIBUTIONS

3.1 Introduction

Ce chapitre comporte deux parties. La première traite l'utilisation de l'ACP principalement pour deux importantes applications : a) Réduire le nombre de coefficients DCT. b) Trouver une meilleure structure d'arbre adaptée à chaque locuteur. La deuxième partie traite la fusion de l'information en utilisant différentes combinaisons de paramètres acoustiques. Ces derniers sont obtenus soit par une analyse utilisant la transformée de Fourier soit par une analyse utilisant la transformée en ondelettes. Les paramètres utilisant les deux méthodes d'analyse sont ensuite combinées selon la technique décrite dans ce chapitre.

3.2 Utilisation de l'ACP comme post-traitement

Par application de l'ACP nous allons réduire le nombre de coefficients DCT statique (12 dans notre cas). La Figure 17 représente un exemple de traitement en considérant que les paramètres DCT sont obtenus à partir d'une analyse en ondelettes.

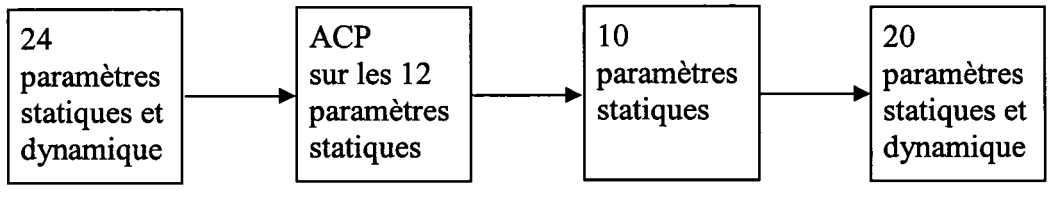


Figure 17 Schème d'exemple de traitement pour réduire le nombre de paramètres DCT.

3.3 Méthode proposé pour une meilleure structure d'arbre

Dans cette partie nous allons utiliser l'ACP pour trouver une meilleure structure de base d'ondelettes. L'idée fondamentale de cette méthode est l'utilisation d'un taux de variance adéquat qui permet d'obtenir une base optimale sur la quel l'information est plus concentrée. À partir d'une décomposition complète en paquet d'ondelettes d'un signal nous devons chercher les nœuds qui permettent une meilleure structure d'arbre. En effet l'arbre obtenu dépend des corpus utilisés et du taux de variance.

3.3.1 Les différents corpus utilisés

À partir d'un ensemble de phrases prononcées par un locuteur nous allons définir trois corpus.

Corpus AP :

C'est un ensemble de données obtenu uniquement pour une seule phrase donnée. Dans ce cas nous allons obtenir un arbre Abstrait par Phrase (AP).

Corpus AL :

C'est un ensemble de données obtenu à partir de l'ensemble de toutes les phrases d'entraînement disponibles pour un locuteur donné. Par application de l'ACP nous allons obtenir un arbre Abstrait par Locuteur (AL).

Corpus ALN :

C'est un ensemble de données obtenu à partir de l'ensemble de toutes les phrases d'entraînement d'un locuteur. À chaque niveau de décomposition en paquet d'ondelettes nous allons générer un corpus qui lui correspond. Par application de l'ACP, l'arbre obtenu dans ce cas est un arbre Abstrait

par Locuteur généré à partir des différents Niveaux de décompositions (ALN).

Dans la section suivante nous introduisons un nouvel algorithme de meilleure structure d'arbre abstrait qui permet d'obtenir les trois types d'arbre cités plus haut.

3.3.2 Algorithme de la Meilleure structure d'Arbre ABstrait obtenu par l'utilisation de l'ACP (MSAAB)

Dans le cadre de notre travail de recherche, nous avons proposé un algorithme qui permet de construire la meilleure structure d'arbre abstrait (MSAAB) selon un critère donné. Cet algorithme dépend du corpus utilisé et permettra d'extraire les caractéristiques de la phrase ou du locuteur selon les cas. L'algorithme MSAAB fonctionne de la façon suivante.

1- Pré-traitement du signal audio :
   a. Échantillonnage du signal selon le théorème de Nyquits.
   b. Pré-accentuation du signal.
   c. Fenêtrage.
2- Choisir l'un ou l'autre des deux critères : le critère d'entropie ou le critère d'énergie.
3- Choisir un type d'ondelette.
4- Faire une décomposition complète en paquet d'ondelettes jusqu'à un niveau voulu.
5- Pour tous les nœuds de l'arbre, calculer les valeurs numériques des paramètres utilisant le critère choisi.
6- Concaténer toutes ces valeurs numériques pour obtenir un corpus propre à chaque locuteur selon l'un ou l'autre des trois cas suivants :

Cas AP : corpus relatif à une phrase.

Cas AL : corpus relatif à un locuteur.

Cas ALN : corpus relatif aussi à chaque locuteur mais en introduisant les niveaux de la structure d'arbre.

7- Appliquer l'ACP pour un taux de variance donné.

Le schème général qui relie tous les blocs utilisés dans l'algorithme pour obtenir une structure d'arbre abstrait est illustré par la figure 18 :

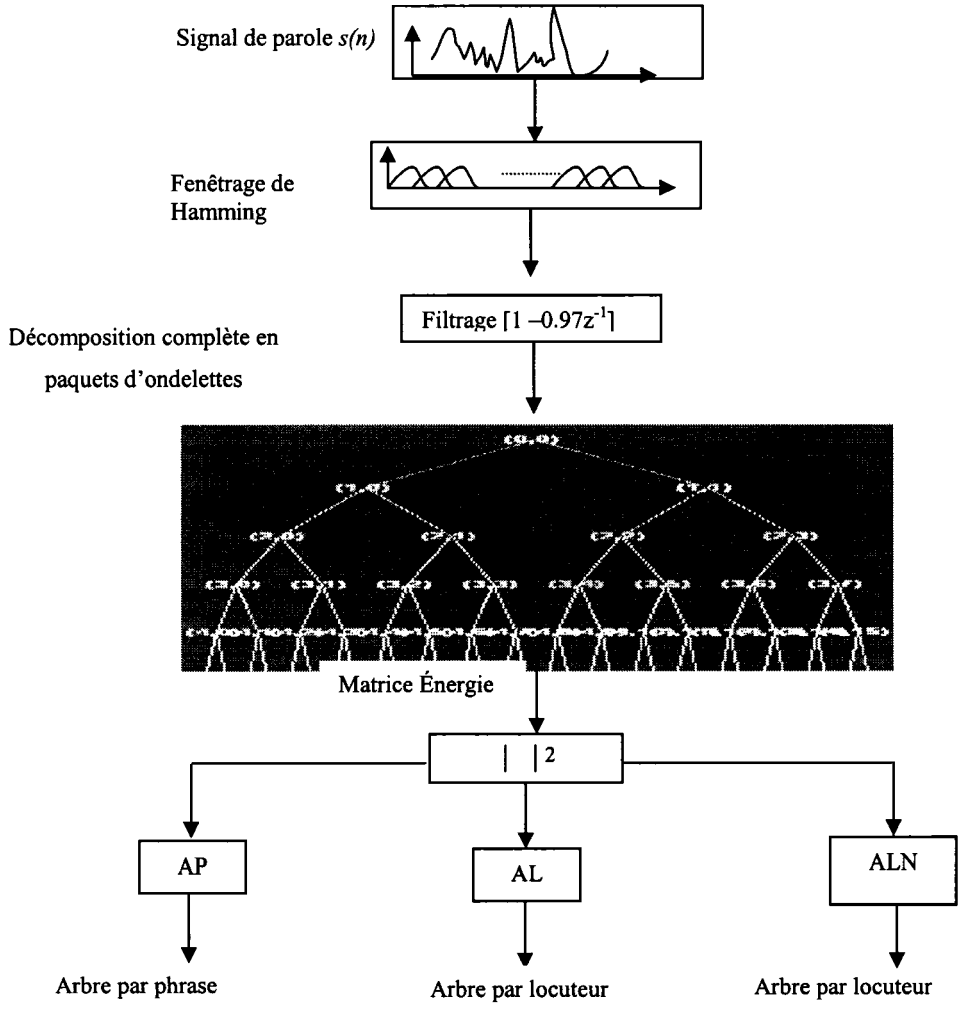


Figure 18 Schème général pour trouver la meilleure structure d'arbre utilisant le critère d'énergie.

3.3.3 Explication du fonctionnement de l'algorithme MSAAB

Les étapes 4 à 7 nécessitent des explications dont le détail est donné ci-dessous :

Étape 4 :

Pour une phrase correspondant à un locuteur donné nous allons utiliser l'ondelette de Daubechie. Comme il a était fait dans les travaux antérieurs, l'ordre optimal de ces ondelettes pour obtenir un meilleur taux de reconnaissance a été fixé à 8 [1]. Par la suite nous allons faire la décomposition complète en paquet d'ondelettes à 5 niveaux d'échelle. À la fin de cette étape nous obtenons une structure générale de l'arbre donnée par la Figure 19. Chaque nœud (père) C_n^k (nœud k du niveaux n, n=1,2,3,4,5) est divisé en deux nœuds fils C_{n+1}^{2k+1} et C_{n+1}^{2k+2}. Les coefficients du nœud C_0^0 sont ceux du signal obtenu à l'étape 3.

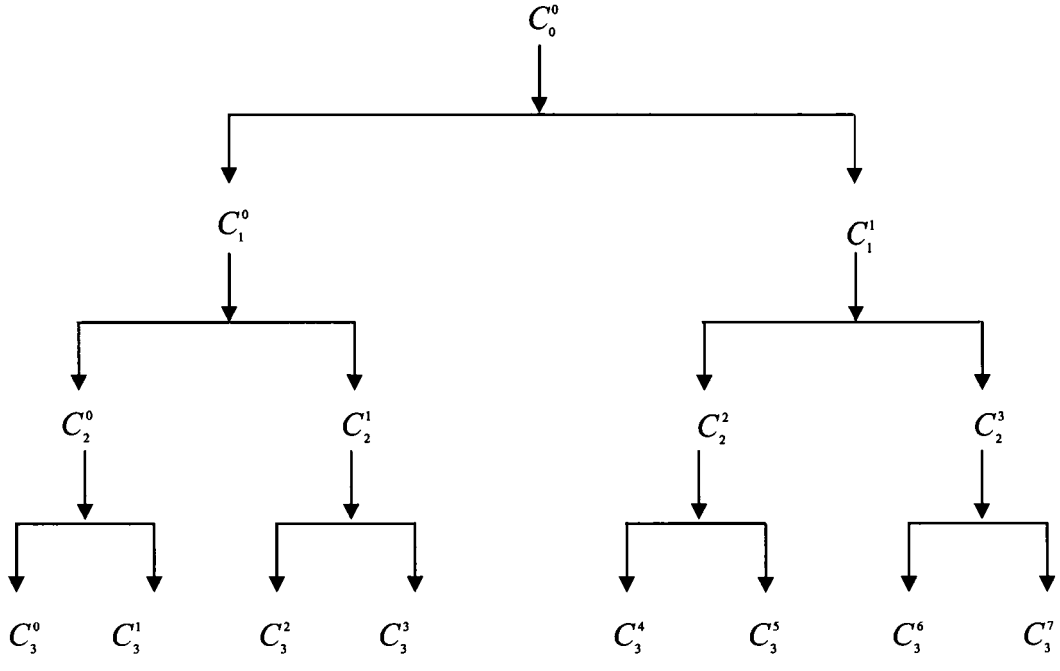


Figure 19 Structure générale d'une décomposition en paquet d'ondelettes à niveau 3 d'échelle.

Étape 5.

À partir des coefficients du signal correspondant au nœuds C_n^k nous allons calculer les valeurs numériques liées au critère choisi à l'étape 4 :
Si nous avions choisi le critère de l'entropie de Shannon non normalisée, la valeur numérique correspondante du nœud C_n^k est donnée par l'équation suivante :

$$E_n^k = -\sum_i |C_n^k(n)|^2 \log\left(|C_n^k(i)|^2\right) \qquad (3.1)$$

E_n^k est l'entropie de Shannon correspondant au nœud k du niveau d'échelle n.
Si nous avions choisi le critère de l'énergie la valeur numérique correspondante du nœud C_n^k est égale à son énergie donnée par l'équation suivante :

$$E_n^k = \sum_i |C_n^k(i)|^2 \qquad (3.2)$$

Étape 6 :

Dans cette étape nous devons générer l'un ou l'autre des trois corpus cités dans la section 3.3.1 de la façon suivante :

1. Corpus AP :

Le corpus appelé A_p est obtenu uniquement à partir d'une seule phrase. L'arbre obtenu après l'étape 9 est un arbre relatif à celle-ci. Pour un même locuteur nous obtiendrons autant d'arbres admissibles que de phrases disponibles. La matrice qui représente cet ensemble est donnée par :

$A_p = [\{A_f\} f = 1,2...,N_{f,p}]$, $N_{f,p}$ est le nombre de fenêtre obtenu pour la phrase p prononcée par un locuteur donné. L'élément du vecteur A_f est défini par :

$A_f = \{E_i^k\}_{i=1,2....,n}^{k=1,2,...k_i}$, avec k_i est le nombre de nœuds du niveau i. A_f

est un vecteur de dimension : $(2^n+1)\times 1$.

2. Corpus AL :

Ce corpus appelé Ar est obtenu à partir de l'ensemble des phrases d'entraînement d'un même locuteur.

$Ar=[\{As\}_{s=1,2,...Ns}]$, Ns est le nombre de sessions disponible.

$As=[\{Ap\}_{p=1,2....,Np}]$, Np est le nombre de phrases dans une session.

$$Ap=[\{Af\}_{f=1,2,...,N_{f,p}}] \tag{3.3}$$

Af est défini de la même manière que dans le premier cas. Ar est une matrice de dimension dn donnée par :

$$dn=(2^n+1)\times Nf\times Np\times Ns \tag{3.4}$$

Nf est le nombre total des fenêtres obtenu pour toutes les phrases prononcées

$$Nf=\sum_p N_{f,p} \tag{3.5}$$

L'arbre obtenu après l'étape 7 est un arbre relatif à un locuteur donné.

3. Corpus ALN :

À partir d'une décomposition en paquet d'ondelettes de profondeur n, nous allons construire n ensembles $\{Ae\}_{e=1,2,...,n}$. Chaque ensemble Ae est un sous-ensemble de Ar. Un ensemble Ae est obtenu de la même façon que Ar. La différence réside dans la définition de Af

$$Af=[\{Ee^k\}_e^{k=1,2,...ke}] \tag{3.6}$$

ke est le nombre de nœuds du niveau e. Chaque matrice Ae est une matrice relative à un niveau e et de dimension :

$$de=(2^e+1)\times Nf\times Np\times Ns \tag{3.7}$$

Étape 7 :

1. **Corpus AP** :

   Pour un taux de variance donné, l'ACP est appliquée directement sur la matrice A_p. L'arbre obtenu correspond à une phrase d'un locuteur donné.

2. **Corpus AL** :

   L'ACP est appliquée sur la matrice A_r avec un taux de variance bien déterminé, et la structure de l'arbre abstrait obtenu est propre à chaque locuteur.

3. **Corpus ALN** :

   Pour chaque matrice A_e on applique l'ACP comme dans les deux cas précédents. Les nœuds qui forment la structure de l'arbre dans ce cas sont les nœuds de chaque arbre obtenu pour toutes les valeurs de e correspondant au niveau d'échelle de la décomposition en paquets d'ondelettes e=1,2…,n (dans notre cas n=5). La structure de l'arbre obtenu est une structure propre à chaque locuteur.

Exemple :

- **Génération du corpus AP**

Pour une phrase ayant 5 fenêtres d'analyse et un arbre de profondeur 3, le nombre de noeuds obtenus par une décomposition en paques d'ondelettes est égal à 15 (figure 20). Dans cette section, nous allons designer un nœud par un nombre naturel j. Le nœud j correspond au nœud C_n^k avec k et n sont définis tel que : $2^n-1 \leq j \leq 2^{n+1}-2$, et $k=j-1-2^n$.

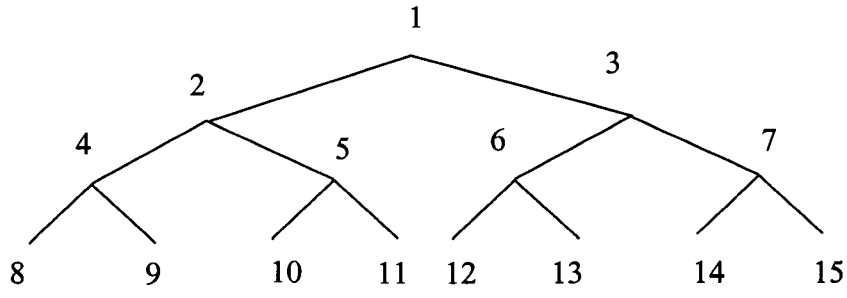
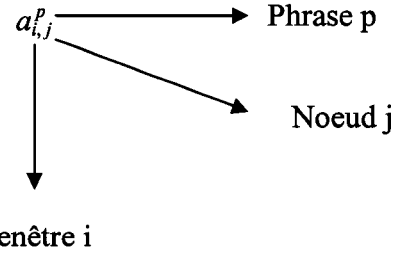


Figure 20 Nœuds correspondant à un arbre de profondeur 3.

La matrice A_p est la suivante :

$$A_{p=1} = \begin{bmatrix} a^1_{1,1} a^1_{1,2} .. a^1_{1,14} a^1_{1,15} \\ a^1_{2,1} a^1_{2,2} .. a^1_{2,14} a^1_{2,15} \\ . \quad . \quad . \quad . \quad . \\ a^1_{5,1} a^1_{5,2} .. a^1_{5,14} a^1_{5,15} \end{bmatrix}$$

la valeur de a^p_{ij} obtenue selon le critère choisi, représente l'information suivante :



La dimension de la matrice Ap est : $\dim(A_p) = 5 \times 15$

- **Génération du corpus AL**

Pour un locuteur ayant 2 phrases. Chacune des 2 phrases contient 5 fenêtres d'analyses. La matrice A_T est une concaténation des deux matrices $A_{p=1}$ et $A_{p=2}$

$$A_T = \begin{bmatrix} A_{p=1} \\ A_{p=2} \end{bmatrix}$$

où $A_{p=1}$ et $A_{p=2}$ sont les matrices du corpus A_P pour les phrases 1 et 2 respectivement. La dimension de A_T est dim(A_T)=(5+5)×15

$$A_T = \left[\begin{array}{ccc|ccc|ccc} a^1_{1,1} & a^1_{1,2} & a^1_{1,3} & \overbrace{a^1_{1,4} & a^1_{1,5} & a^1_{1,6} & a^1_{1,7}}^{A_{e=2}} & \overbrace{a^1_{1,8} & \cdots & a^1_{1,15}}^{A_{e=3}} \\ \vdots & & & \vdots & & & & \vdots & & \vdots \\ a^1_{5,1} & a^1_{5,2} & a^1_{5,3} & a^1_{5,4} & a^1_{5,5} & a^1_{5,5} & a^1_{5,7} & a^1_{5,8} & & a^1_{5,15} \\ a^2_{1,1} & a^2_{1,2} & a^2_{1,3} & a^2_{1,4} & a^2_{1,5} & a^2_{1,6} & a^2_{1,7} & a^2_{1,8} & & a^2_{1,15} \\ \vdots & & & \vdots & & & & \vdots & & \vdots \\ a^2_{5,1} & a^2_{5,2} & a^2_{5,3} & a^2_{5,4} & a^2_{5,5} & a^2_{5,6} & a^2_{5,7} & a^2_{5,8} & & a^2_{5,15} \end{array}\right]$$

- **Génération du corpus ALN**

Pour le même locuteur (cas AL), nous allons extraire deux matrices ($A_{e=2}$ et $A_{e=3}$) de la matrice A_T. Nous obtiendrons deux corpus ALN, le premier est relatif au niveau 2 et le deuxième au niveau 3.

$$A_{e=2}=\begin{bmatrix} a^1_{1,4}a^1_{1,5}a^1_{1,6}a^1_{1,7} \\ a^1_{2,4}a^1_{2,5}a^1_{2,6}a^1_{2,7} \\ . \\ . \\ . \\ a^2_{1,4}a^2_{1,5}a^1_{1,6}a^1_{1,7} \\ . \quad . \quad . \quad . \\ \\ a^2_{5,4}a^2_{5,5}a^2_{5,6}a^2_{5,7} \end{bmatrix} \qquad A_{e=3}=\begin{bmatrix} a^1_{1,8}a^1_{1,9}..a^1_{1,15} \\ a^1_{2,8}a^1_{2,9}..a^1_{2,15} \\ . \\ . \\ . \\ a^2_{1,8}a^2_{1,9}..a^1_{1,15} \\ . \quad . \quad . \quad . \\ \\ a^2_{5,8}a^2_{5,9}..a^2_{5,15} \end{bmatrix}$$

Nous allons définir deux nouveaux taux de variance afin d'appliquer l'ACP à l'étape 7 le Taux de Variance Effectif (TVE) et le Taux de Variance Réel (TVR).

3.3.4 Taux de Variance Effectif (TVE), et Taux de Variance Réel (TVR)

Notre objectif dans l'application de l'ACP est d'éliminer un ensemble de variables tout en gardant l'essentiel de l'information. Le taux cumulé par l'ensemble des Composantes Principale (CP) retenu ne peut être égal au taux de variance fixé. Pour cela nous allons définir deux taux de variance le TVR et le TVE.

- Le taux de variance TVE est le pourcentage de l'information qu'on désire retenir de l'information initiale.
- Le Taux de variance réel est le pourcentage de l'information réellement retenu.

Soit i un nombre de CP de valeur cumulative TVE_i. Il existe un et un seul nœud à partir duquel le TVE_{i+1} devient supérieur à TVE. Nous définissons les deux paramètres suivants : $T_- = TVE_i - TVE$ et $T_+ = TVE_{i+1} - TVE$

Ces deux paramètres nous sont utiles dans les prochains paragraphes.

3.3.5 Maximisation et minimisation de l'information

Le calcul du TVR à partir du TVE peut être obtenu de deux façons :

a) **Calcul du TVE par le critère de maximisation de l'information :**
Le TVR est obtenu en augmentant le TVE de T_+. Dans ce cas on dit qu'on maximise l'information et le TVR = TVE_{i+1}.

b) **Calcul du TVE par le critère de minimisation de l'information**
Le TVR est obtenu en diminuant le TVE de T.. Dans ce cas, on dit qu'on minimise l'information et le TVR = TVE_i.

Dans notre cas, nous avons effectué le choix suivant :

- Utiliser le cas de maximisation de l'information.
- Dans le cas particulier où TVR atteint 100%, nous avons jugé inutile d'appliquer l'ACP.

3.3.6 Détermination du coût introduit par le TVR

Le TVR réellement utilisé introduit un coût dans l'algorithme MSAAB. À chaque application de l'ACP dans l'algorithme MSAAB. La valeur de T+ (resp. T-) introduit un coût. Le nœud i tel que $T_+ = TVE_{i+1} - TVE$ dépend du locuteur et de la valeur de TVE. Une valeur de T+ plus importante correspond aussi à une importante CP du nœud (i+1) et par conséquent le dernier nœud choisi par application de l'ACP est plus représentatif pour un locuteur donné. Par contre il est possible pour un autre locuteur que ce même nœud correspond à une valeur CP plus faible. Dans ce cas il serait plus approprié de maximiser l'information pour le premier locuteur et de la minimiser pour le second. Le cas idéal pour minimiser ce coût est d'utiliser un critère adéquat à chaque application de l'ACP. Le coût introduit est minimal si :

- Les valeurs $(TVE_{i+1} - TVE_i)$ et CP_i sont plus importantes.

  Ceci est possible pour une valeur plus faible de TVE. Dans le cas AL ou AP l'information retenue est minimale. Il a été vérifié lors de nos différentes simulations que l'arbre obtenu contient un nombre de nœuds insuffisant pour le calcul des paramètres DCT (dans notre cas au minimum12 nœuds). Dans le cas ALN pour une valeur faible de TVE, et les nœuds sélectionnés sont plus représentatifs pour chaque niveaux.
- La valeur de (TVE_{i+1}-TVE) est très faible.

  Le choix de la valeur de TVE est très important. Par test de performance nous allons trouver une valeur optimale $TVE_{optimal}$.

3.3.7 Valeur optimale de TVE en utilisant MSAAB

La valeur optimale du TVE est la valeur qui correspond à une meilleure performance de reconnaissance. Pour cela, nous proposons un nouvel algorithme utilisant MSAAB dont l'organigramme est présenté à la Figure 20.
Les étapes suivantes permettent de déterminer la valeur optimale de TVE.

Étape 1 : Appliquer MSAAB pour une valeur donnée de TVE.

Étape 2 Choisir un pas pour varier TVE.

Étape 3 Comparer les performances du système de reconnaissance correspondant à chaque TVE.

L'algorithme MSAAB est appliqué à l'étape 1 pour une valeur de TVE qui correspond à un maximum de l'information initiale. Le maximum est un paramètre prédéfinie. Le TVE est incrémenté d'un pas T et des tests de performance sont réalisés. Le $TVE_{optimal}$ est celui qui correspond à la meilleure performance du système de reconnaissance. Nos expériences ont montré que ce pas est de l'ordre de grandeur de la troisième ou de la quatrième CP utilisant ALN3

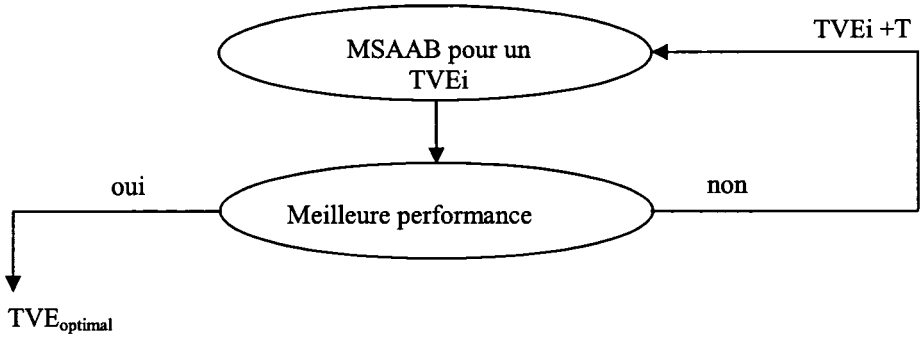


Figure 21 Estimation du taux de variance optimal.

3.3.8 Interprétation en terme de segmentation fréquentielle

Dans la décomposition en paquet d'ondelettes nous utilisons des filtres QMF (Quadrature Miroir Filter). La bande de fréquences du signal à chaque niveau d'échelle de l'arbre est subdiviser en deux parties égales. À partir de l'algorithme MSAAB, deux types de segmentions fréquentielles sont possibles comme le montre la Figure 21. La première correspond au MSAAB utilisant AL et AP et la seconde correspond à celle utilisant ALN.

3.3.8.1 Segmentation fréquentielle par bande totale

À partir d'une bande de fréquence d'un signal [0,fmax] nous pouvons obtenir directement une meilleure organisation de l'information. En effet chaque nœud de l'arbre obtenu par application du MSAAB correspondant au cas du AL est une sous-bande de fréquences plus représentatives du locuteur. Schématiquement, ce cas correspond au cas (2) dans la Figure 21. Dans ce cas, la meilleure segmentation fréquentielle est obtenue directement à partir de tous les nœuds qui correspondent à une décomposition du signal original. Utilisant les facteurs échelles obtenus dans la grille dyadique tel que vu dans la section 1.1.2 du chapitre 1. La sélection des bandes de

fréquences est obtenue à partir d'un ensemble qui regroupe 5 ensembles de bandes de fréquences. Chacun de ces ensembles est déduit de l'autre en utilisant un facteur d'échelle plus fin.

3.3.8.2 Segmentation fréquentielle par bande de niveaux de l'arbre

Nous procédons par une segmentation sur des segments plus au moins grossier. D'abord une première segmentation utilisant une échelle de valeur 2^2. Les nœuds obtenus sont de la même génération. Ils correspondent à un même niveau de résolution. En suite nous changeons l'échelle pour en obtenir un plus fin que le précédent (2^3, 2^4 et 2^5). Les nœuds obtenus forment une nouvelle génération plus jeune (au sens père et fils) que la précédente. Le processus est répété jusqu'au niveau d'échelle le plus fin que nous avons choisi au départ (5 dans notre cas). Les bandes de fréquences qui forment l'arbre sont celles obtenues pour chaque génération.

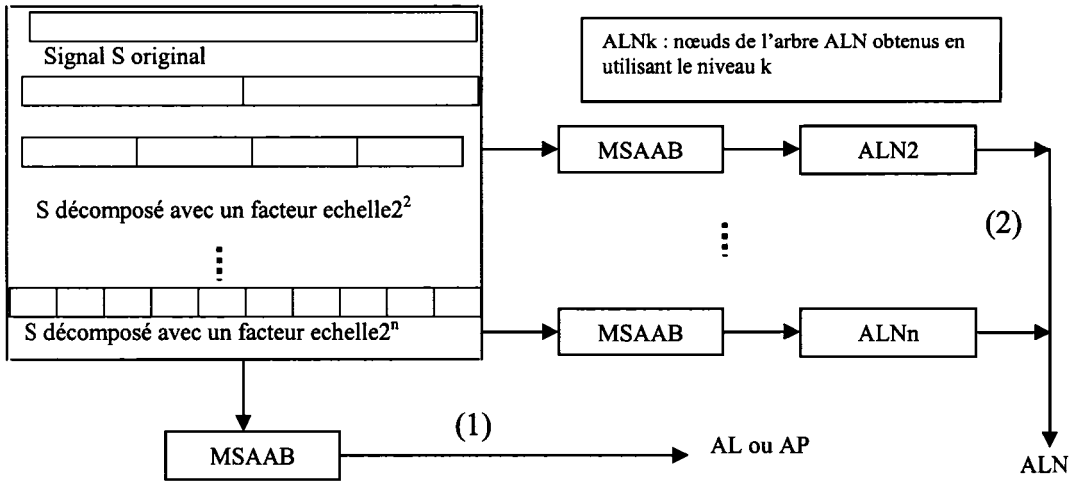


Figure 22 (1) segmentation fréquentielle à partir de la décomposition complète en ondelettes, (2) segmentation fréquentielle par niveaux de décomposition.

3.3.9 Exemple d'exploitation de MSAAB et cas particuliers

Afin d'alléger le texte, nous allons adopter les notations suivantes :

- AL_τ : arbre AL utilisant un taux de variance TVE = τ.
- ALN_τ : arbre ALN utilisant un taux de variance TVE = τ.

Dans cette section nous allons, designer un nœud par un nombre naturel j. Le nœud j est le nœud C_n^k avec k et n sont définis tel que : $2^n-1 \leq j \leq 2^{n+1}-2$, et $k=j-1-2^n$.

À partir d'un même exemple nous allons élaborer d'une façon détaillée la procédure qui permet la construction de l'arbre abstrait pour AL et ALN. Pour cela nous avons effectué le choix suivant (Tableau I) afin d'appliquer l'algorithme MSAAB :

Tableau I

Paramètres utilisés pour appliquer l'algorithme MSAAB.

Attribut	Valeur
Ondelette utilisée	Daubechie 8
Profondeur de l'arbre	5
Critère choisi	énergie

3.3.9.1 Construction de l'arbre AL pour différents TVE

La sélection des nœuds d'un arbre AL est basée sur la richesse en information que ceux-ci renferment. Dans le Tableau II nous remarquons que seul 6 composantes principales ont un taux de variance cumulé de 71.28%, par contre il y a des CP qui ne portent aucune information (de valeur propre nulle) et il y en a d'autre qui portent peut d'information. Leur pourcentage d'information cumulé est très petit et ne dépasse même pas 1%. Il faut 39 CP pour obtenir un taux de variance cumulé de 3.81%. Cet algorithme

ne peut être appliquer qu'à partir d'un taux cumulatif seuil. Ce seuil correspond à celui des 12 premiers nœuds. Dans cet exemple le seuil est de TVE_{seuil}=84.96%. Notons que presque la totalité de l'information 96.19% est renfermée dans 24 nœuds.

Tableau II

Composantes principales qui permettent d'obtenir AL.

CP	Pourcentage de variance (%)	Pourcentage de variance Cumulé (%)
CP1	31.70	31.70
Cp2	12.16	43.87
CP3	10.98	54.86
CP4	6.91	61.77
CP5	5.11	66.89
CP6	4.39	71.28
CP7	3.19	74.47
CP12	1.53	84.96
CP1à24	-	96.19
CP24à63	-	3.81

L'une des particularités de l'arbre abstrait est l'obtention de nœuds qui peuvent ne pas être des feuilles comme le montre la figure suivante.

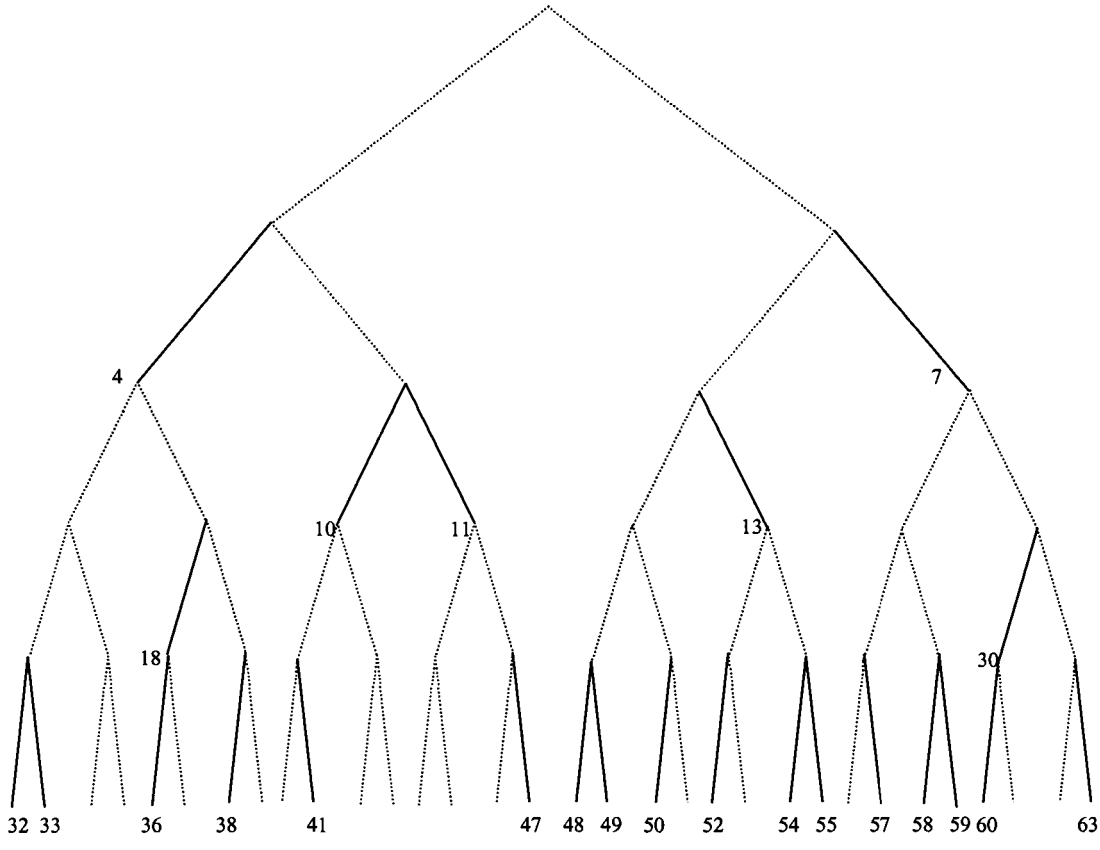


Figure 23 Arbre abstrait AL obtenu avec une valeur TVE=96%.

3.3.9.2 Construction de l'arbre ALN pour différents TVE

La répartition de l'information dans chaque niveaux est donné par les tableaux III à VI :

Tableau III

Répartition de l'information dans les nœuds du niveau 2.

Composante principale	Pourcentage de variance	Pourcentage devariance cumulé
CP1	54.2834	54.28
CP2	21.1323	75.41
CP3	16.2257	91.63
CP4	8.3586	100.00

Tableau IV

Répartition de l'information dans les nœuds du niveau 3.

Composante principale	Pourcentage de variance	Pourcentage de variance cumulé
CP1	39.3496	39.3496
CP2	15.7651	55.1147
CP3	14.8454	69.9601
CP4	9.5628	**79.5229**
CP5	7.9447	87.4676
CP6	5.1140	92.5816
CP7	4.3043	96.8859
CP8	3.1141	100.0000

Tableau V

Répartition de l'information dans les nœuds du niveau 4.

Composante principale	Pourcentage de variance	Pourcentage de variance cumulé
CP1	30.69	30.69
CP2	12.53	43.22
CP3	12.41	55.64
CP4	8.28	63.92
CP5	6.21	**70.13**
CP6	5.85	75.98
CP7	4.16	80.15
CP8	3.60	83.75
CP9 à 16	-	100.00

Tableau VI

Répartition de l'information dans les nœuds du niveau 5

Composante principale	Pourcentage de variance	Pourcentage de variance cumulé
CP1	23.44	23.4445
CP2	10.46	33.9139
CP3	9.83	43.7525
CP4	6.53	50.2874
CP5	5.06	55.3483
CP6	4.72	60.0725
CP7	3.59	63.6644
CP8	2.97	66.6436
CP9	2.89	69.5336
CP10	2.59	**72.1312**
CP11	2.14	74.2713
CP12	2.03	76.3078
CP13 à CP32		100.00

À partir des tableaux III à VI, nous pouvons déduire le nombre de nœuds nécessaires à chaque niveau pour avoir un taux de variance cumulatif TVE=70%. Ces valeurs sont présentées par la Tableau suivante :

Tableau VII

Répartition du nombre de nœuds dans les niveaux de l'arbre ALN_{70}.

niveau	Nombre de nœuds	nœuds totaux
2	2	21
3	4	
4	5	
5	10	

Nous remarquons qu'il y a 21 nœuds dans ALN_{70} alors qu'il y a seulement 12 nœuds dans AL_{84}.

Les arbres obtenus pour ALN_{70} et ALN_{96} sont présentés dans les figures 23 et 24.

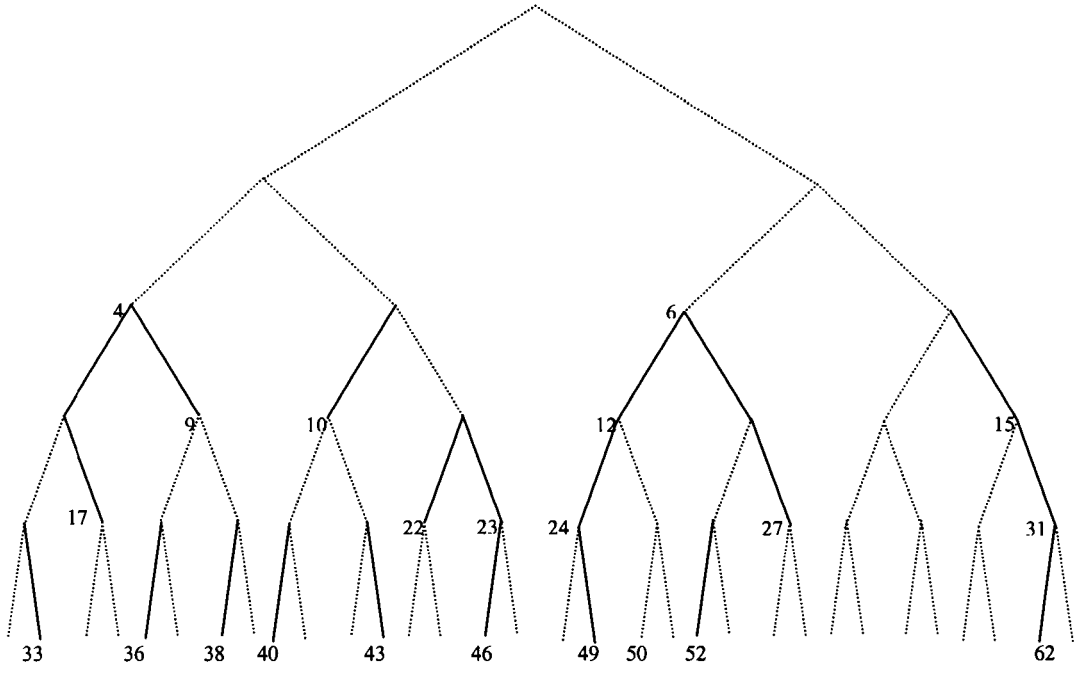


Figure 24 Arbre abstrait ALN obtenu avec une valeur TVE=70%.

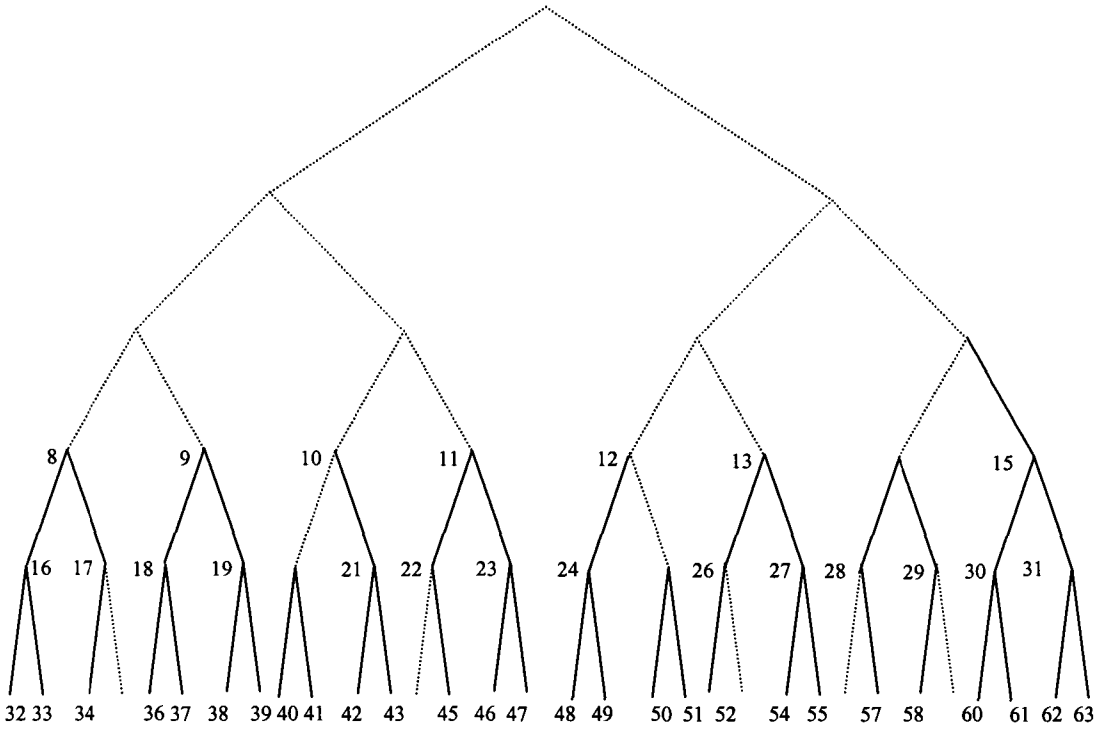


Figure 25 ALN obtenu pour TVE=96%.

3.3.9.3 Analyse des résultats de l'exemple présenté

Comme il a été mentionné au paragraphe 3.2.4, le TVR du ALN_{96} du niveau 2 du tableaux III atteint 100%. Ce niveau ne participe pas à la construction de l'arbre ALN_{96}. Nous remarquons que l'arbre ALN_{96} ne contient aucun nœud du niveau 2.

Dans ce cas, si nous avions choisi de ne pas introduire le niveau 2 dans l'application de l'algorithme MSAAB, l'arbre ALN est obtenu uniquement à partir des niveaux 3, 4 et 5. Si le TVE est plus faible (inférieur à 54.2834+21.1323=75.4157), le niveau 2 participera lui aussi dans la construction de l'arbre ALN_7. En plus, avec un taux de variance fixé à 70, le pourcentage cumulé par les deux composantes principales CP1 et CP2 est supérieur à 70%. L'écart est de 5.4157% et nous devrons choisir entre deux cas :

a) Le pourcentage de variance fixé au départ à 70% est augmenté de 5.4157%. C'est la maximisation de l'information.
b) Le pourcentage de variance de départ est diminué de 21.1323 et nous aurons un pourcentage réel de 54.2834. C'est la minimisation de l'information.

Le coût d'utilisation de l'algorithme MSAAB dépend du nombre de nœuds. En effet, chaque nœud est traduit par un taux de variance correspondant à une composante principale. Plus le nombre de nœuds est important plus le TVE est étalé sur des intervalles de valeurs plus au moins faibles, et ainsi la différence (TVE-TVR) est plus faible. L'utilisation du choix a) ou b) est indifférent et le coût introduit sera plus faible. Ce coût dépend aussi de la valeur de TVE. Il devrait y avoir donc un compromis entre le nombre de nœuds et le TVE.
Le coût dans le ALN apparaît autant de foi qu'il y a de niveaux. Alors qu'il n'apparaît qu'une seule fois lors de la construction de l'arbre AL ou AP.

3.3.10 Arbre pseudo-dyadique

Les nœuds de l'arbre abstrait obtenu par l'algorithme MSAAB subit un traitement supplémentaire suivant :
Un nœud de l'arbre abstrait est retenu

- Si ce nœud est une feuille.
- Toutes les feuilles frères de celles sélectionnées sont ajoutées.

L'arbre obtenu ne contient que des feuilles. Cet arbre est peu différent de celui obtenu par une analyse multirésolution dyadique. Nous allons lui attribuer le nom d'arbre pseudo-dyadique.

La figure suivante présente un exemple d'arbre ALN_{96} pseudo-dyadique

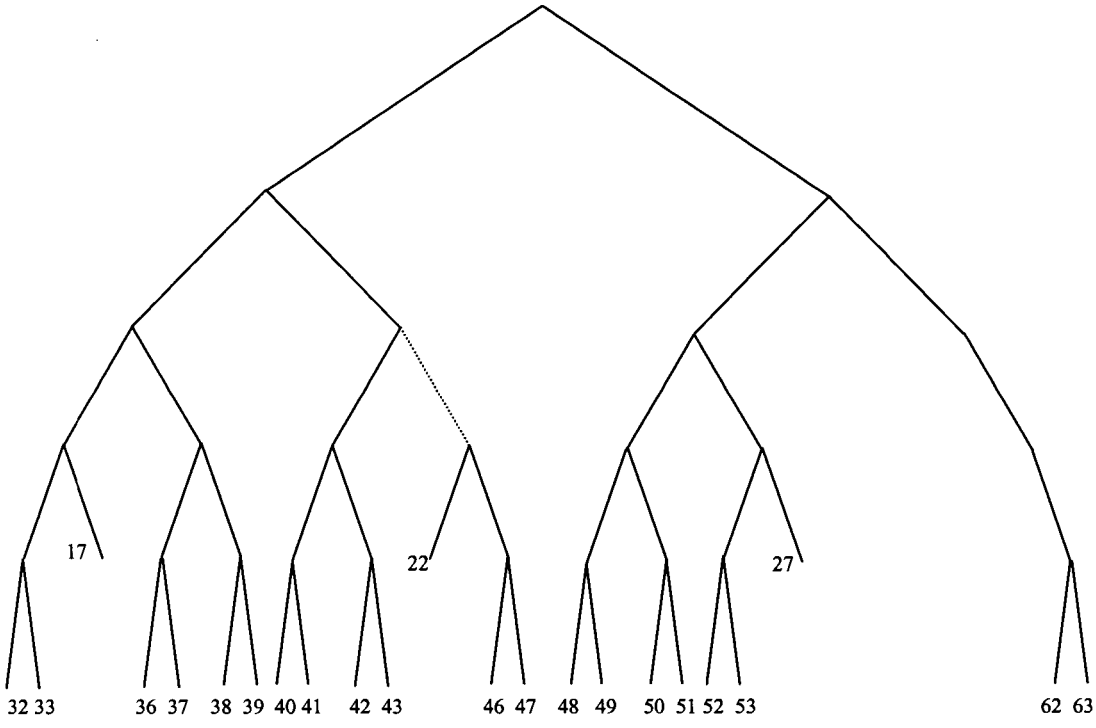


Figure 26 Arbre ALN_{96} pseudo-dyadique.

3.4 Fusion de l'information dans un système de reconnaissance du locuteur

3.4.1 Définition des ensembles de combinaison

Dans cette partie nous combinons deux types de traitement d'un signal vocal : l'analyse de Fourier et l'analyse de la transformée en ondelettes.

Dans la partie modélisation du chapitre 2, nous avons utilisé S streams pour définir la loi de distribution qui décrit un état généré par un modèle HMM. Ces streams doivent être statistiquement indépendants. Dans un environnement HTK, nous avons la possibilité d'utiliser plusieurs streams. Les streams utilisés par ce logiciel sont des paramètres obtenus à partir d'un seul type d'analyse. Ces streams sont les paramètres statiques obtenus à partir d'une analyse donnée, et leurs dérivées première et seconde (voir Figure

26). Au lieu d'utiliser des streams obtenus à partir d'une même analyse, nous utilisons un ensemble de streams obtenus à partir de deux types d'analyses. Les différentes combinaisons possibles sont :

- Dans le cas où nous utilisons deux streams, il y a neuf combinaisons possibles. Le Tableau VIII donne les différentes combinaisons possibles pour deux streams. Par exemple (SF , SW) représente la combinaison des paramètres statiques de Fourier et des ondelettes.

Tableau VIII

Combinaisons possibles utilisant 2 streams.

SF , SW	SF, DW	SF, AW	DF, SW	DF,DW	DF,AW	DF, SW	DF,DW	DF,AW

- Dans le cas où nous utilisons 3 strems, il y a 27 (3^3) combinaisons possibles dont voici quelques unes :

(SF , DF , SW)	(SF , AF , SW)	(SF , SW, DW)	(SF , DF , DW)	(SF , AF , SW)

Une fois la combinaison choisie parmi l'ensemble de celles possibles, nous devons déterminer les poids γ_s qui correspondent à chaque stream. Pour un nombre de stream égal à deux, nous avons utilisé deux poids égaux (γ_s=0.5). Ceci veut dire que nous donnons un poids égal pour chaque type d'analyse. Pour un nombre de streams égale à trois, nous avons procédé manuellement afin de trouver une combinaison de poids optimale qui correspond à la meilleure performance possible, comme décrit dans le paragraphe suivant.

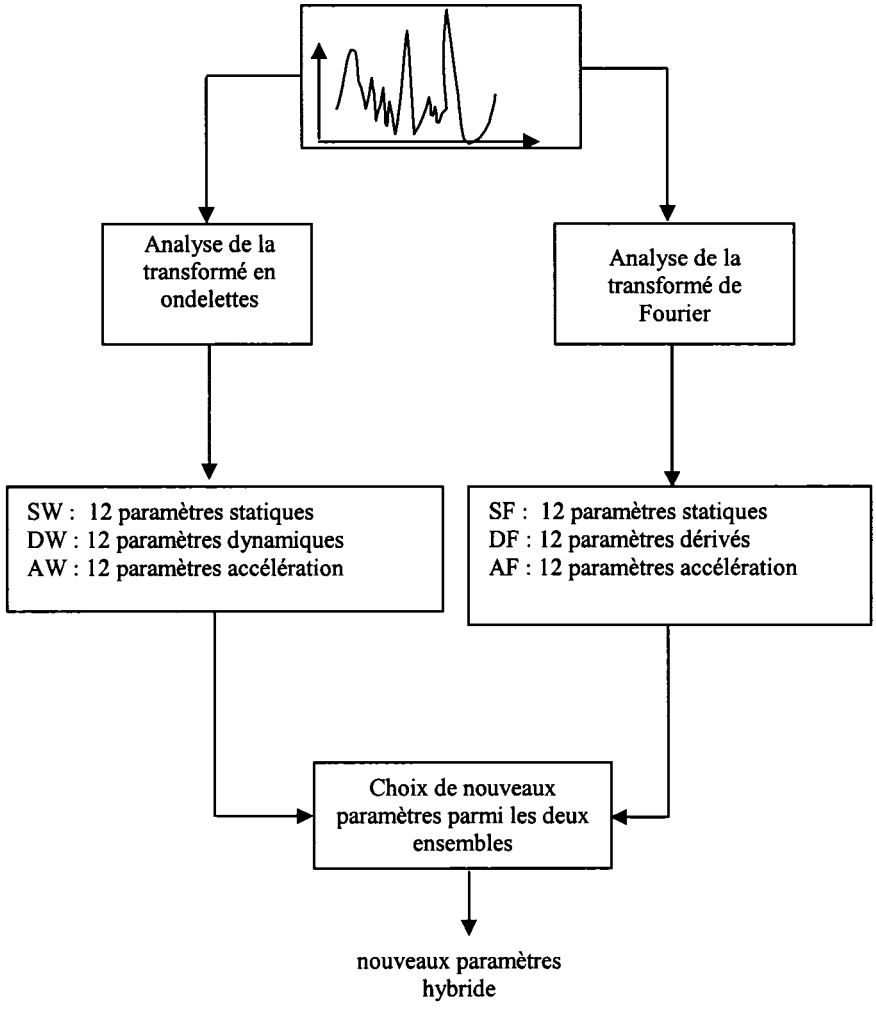


Figure 27 Schème général permettant d'obtenir des paramètres hybrides.

3.4.2 Estimation des poids

Nous avons procédé de la façon suivante pour trouver les poids optimaux :

- Initialiser les valeurs des poids γ_s à 1.
- Garder deux poids fixes et varier le troisième en utilisant un pas de 1. À chaque fois, nous devons comparer les performances. Le processus est répété à tour de

rôle pour les trois poids. De cette façon nous allons obtenir d'une façon grossière les valeurs optimales.

- Refaire la même procédure que précédemment en utilisant les valeurs optimales obtenues avec des pas plus petits. La construction de chaque type de paramètre permettra d'affecter un poids à chaque type d'analyse.

3.5 Conclusion

Dans ce chapitre nous avons détaillé notre principale contribution, qui consiste en la proposition d'une nouvelle technique de segmentation. À partir d'une décomposition en paquet d'ondelettes, l'application de l'analyse en composante principale permet de sélectionner une meilleure structure d'arbre. Contrairement aux arbres vus dans la littérature, la particularité de l'arbre obtenu est qu'il peut contenir un nœud et ceux qu'il peut engendrer. Ceci a permis de construire un arbre unique pour chaque locuteur. Cette construction permet donc de discriminer les paramètres acoustiques de tous les locuteurs et rendra par conséquent le système de reconnaissance du locuteur plus robuste et fiable.

Le chapitre suivant présente les tests et résultats expérimentaux obtenus en utilisant les méthodes présentées dans ce chapitre. Les expériences sont orientées vers une application de vérification de l'identité de locuteur d'une personne par sa voix.

CHAPITRE 4

RÉSULTATS EXPERIMENTAUX

4.1 Introduction

Dans ce chapitre, nous présentons les résultats expérimentaux obtenus. Deux applications différentes sont effectuées en considérant deux types de corpus, soit le corpus Yoho utilisé pour la vérification de l'identité en mode texte-dépendant (VITD), et le corpus Spidre utilisé pour la vérification de l'identité en mode texte-indépendant (VITI). D'abord, une brève description est donnée pour chacune de ces deux corpus. Ensuite les résultats de performances des systèmes VITD et VITI utilisant notre nouvelle méthode de segmentation (MSAAB) sont présentés. Finalement les résultats de ceux obtenus par combinaison de l'analyse par la transformée de Fourier et par ondelettes sont analysés et discutés.

4.2 Description des corpus utilisés

4.2.1 Corpus Yoho

Nous avons utilisé un ensemble de soixante (60) locuteurs (47 hommes et 13 femmes). Ces derniers sont extraits de la base de donnée non bruitée de Yoho. Chaque locuteur prononce 96 phrases dans la phase d'entraînement et 40 phrases dans la phase de vérification. La durée d'une phrase prononcée est approximativement de 6 secondes.

4.2.2 Corpus Spidre

Nous avons utilisé un sous-ensemble de 45 locuteurs (27 hommes et 18 femmes) extraits à partir du corpus Spidre. Quatre conversations sont disponibles pour chaque locuteur. Trois conversations (deux utilisant le même combiné téléphonique, la troisième utilise un combiné différent) sont utilisées pour l'entraînement. Une conversation (combiné téléphonique différent de ceux utilisés pour l'apprentissage) pour la vérification. Chaque phrase prononcée a approximativement une durée de 55 secondes. Le procédé de vérification a été appliqué sur différentes longueurs du signal.

4.3 Analyse en ondelettes

Après application de la décomposition en ondelettes pour chaque fenêtre d'analyse du signal, le log de l'énergie correspondant à chaque bande de fréquences est calculé. Ces bandes de fréquences sont déterminées de la façon suivante :

- À partir d'un banc de filtre correspondant à l'échelle de Mel.
- À partir d'un banc de filtre donné par Farouq [11] obtenu par approximation de l'échelle de Mel.
- À partir d'un banc de filtre obtenu par la Sélection de la Meilleure Base (BBS) basée sur le critère de l'entropie minimale établi par Coifman [7], et appliqué avec succès par Badri [1].
- À partir d'un arbre abstrait qui représente notre contribution. Celui-ci est obtenu par application de la Meilleure Structure d'Arbre Abstrait MSAAB.

En suite la transformée discrète en cosinus (DCT) est appliquée pour obtenir 12 coefficients DCT statiques et 12 autres dynamiques. Les étapes de cette analyse sont résumées par le mode opératoire donné par la figure suivante :

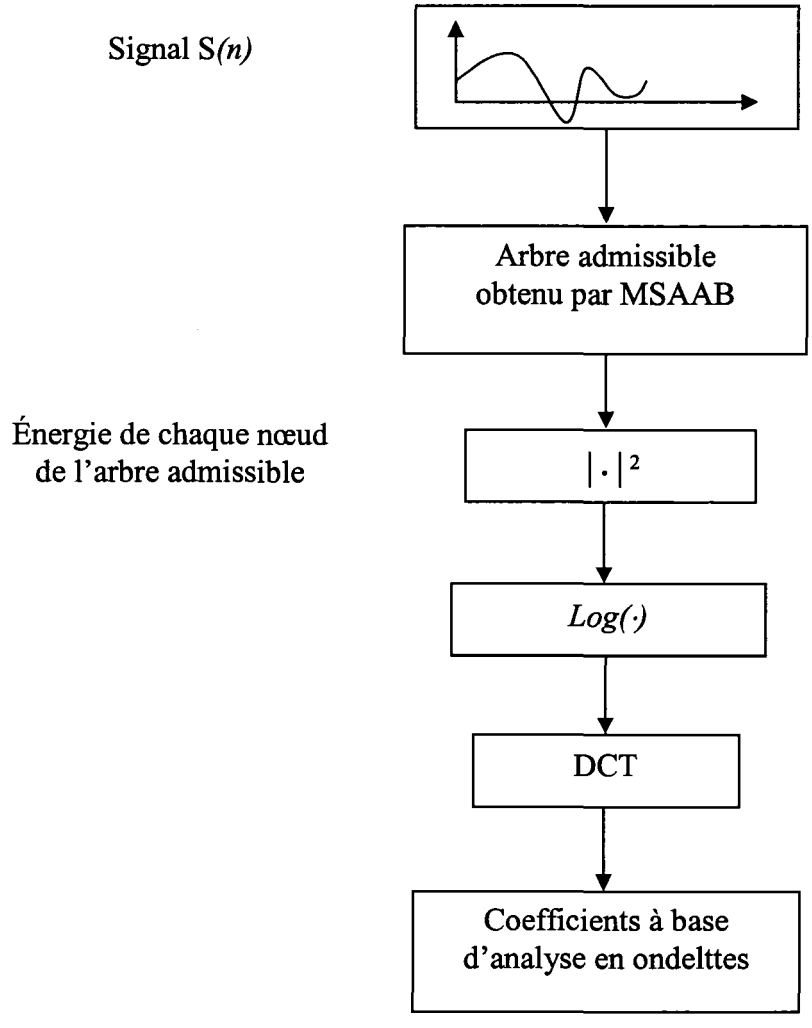


Figure 28 Diagramme bloc d'extraction des paramètres acoustiques à base d'analyse en ondelettes.

4.4 Estimation du modèle de chaque locuteur et du modèle global

Dans le cas utilisant le corpus Yoho, nous avons utilisé 96 phrases d'entraînement pour obtenir une modélisation Markovienne. Pour chaque locuteur, le modèle HMM est ré-estimé deux fois pour obtenir le modèle sur lequel est appliqué le processus de vérification. Nous avons utilisé un modèle HMM à trois états de deux streams (un stream pour les paramètres statiques et un pour les dynamiques). Chaque stream est modélisé par 8 mixtures. Chaque mixture est une matrice de covariance diagonale. Le modèle global a été obtenu à partir de l'ensemble total des phrases d'entraînements.
Dans le cas utilisant le corpus Spidre, nous avons construit des GMM (Gaussien Markov Model) ayant les même caractéristique que les modèles HMM décrits précédemment.

4.5 Performances de reconnaissance

Les performances de reconnaissance sont obtenues en utilisant toutes les phrases de vérification disponibles. À partir du classement de probabilité de reconnaissance d'une phrase (test) prononcée par un locuteur donné, les résultats possibles sont : Fausse acceptation (FA), Faux Rejet (FR). Les performances totales de chaque système de vérification sont évaluées comme suit :

$$FR(\%) = \frac{\sum_{i=1}^{N} fr(i)}{M \times N} \times 100 \qquad (5.1)$$

M est le nombre de phrases de vérifications, N est le nombre de locuteurs, et *fr(i)* est le nombre de faux rejets du i^{éme} locuteur.
De la même façon le nombre total de FA est donnée par la relation suivante :

$$FA(\%) = \frac{\sum_{i=1}^{N} fa(i)}{M \times N} \times 100 \qquad (5.2)$$

fa(i) correspond au nombre de fausses acceptations du $i^{ème}$ locuteur.

La performance globale du système est définit par :

$$SC(\%) = 100 - FA(\%) - FR(\%) \tag{5.3}$$

Enfin le seuil utilisé pour évaluer les probabilités de reconnaissance a été fixé à 0.

4.6 Résultats expérimentaux

4.6.1 Première partie : évaluation du MSAAB

À partir de l'ensemble des phrases d'entraînement d'un locuteur donné nous avons appliqué l'algorithme de sélection de la meilleure structure d'arbre MSAAB pour construire un arbre admissible. Chaque arbre admissible est relatif à un locuteur donné.
Les paramètres utilisés pour l'analyse du signal vocal par ondelettes et par celle de Fourier (MFCC) ont été définit comme suit :

Tableau IX

Type de traitement utilisé.

	Valeur	
Paramètre	Fourier	Ondelettes
Pré-accentuation	$1\text{-}0.97z^{-1}$	$1\text{-}0.97z^{-1}$
Longueur de la fenêtre d'analyse	25ms	25ms
Décalage de la fenêtre d'analyse	10ms	10ms
Nombre de paramètre DCT	24	24
Nombre de coefficient cepstrale	22	NL>13
normalisation des coefficients cepstrals	oui	-
Fenêtrage de Hamming	oui	oui
Ordre de Daubechie	-	8

NL : Nombre de nœuds de l'arbre admissible relatif à chaque locuteur.

L'ensemble des paramètres employés lors de l'application de l'algorithme MSAAB sont données par le tableau suivante.

Tableau X

Paramètres utilisés pour l'application de l'algorithme MSAAB.

Paramètre	Valeur
Taux de variance	96, 80, 70
Critère	Énergie, entropie
Arbre possible	AP, AL, ALN et Pseudo-dyadique

4.6.1.1 Expérimentation utilisant le corpus Yoho

Durant cette partie nous allons adopté les notation suivantes :

ALN_t (resp. AL_t) : Arbre ALN (resp. AL_t) obtenu pour une valeur de TVE égale à t
ALN_t pseudo dyadique (resp. AL_t pseudo dyadique) : Arbre ALN_t (resp. AL_t) rendu pseudo dyadique.

Les tableaux XI et XII présentent les performances obtenues en utilisant seulement 5 et 25 locuteurs respectivement.

Tableau XI
Performances obtenues en utilisant le critère énergie
(5 locuteurs).

Arbre	FR(%)	FA(%)	SC(%)
ALN_{96} et ALN_pseudo dyadique	4.5	0.5	95.0
AL_{96} et AL_pseudo dyadique	0.0	0.0	100.0

Dans le cas AL_{96} le système de vérification VITD est efficace pour 5 locuteurs. Cependant dans le cas de la segmentation fréquentielle utilisant la méthode ALN_{96} le système est confus. Il ne peut pas discriminer même 5 locuteurs. Nous avons tenté de façon approximative de transformer l'arbre ALN en un autre pseudo-dyadique, mais sans grand succès.

Tableau XII
Performances obtenues en utilisant le critère énergie
(25 locuteurs).

	FR(%)	FA(%)	SC(%)
AL_{96}	0.0	0.0	100.0
AL_pseudo dyadique	0.0	8.0	92.0
ALN_{96}	5.0	2.5	91.5
ALN_pseudo dyadique	2.3	2.8	95.0

Dans un test de reconnaissance pour 25 locuteurs, le système VITD reste encore efficace en considérant la méthode AL_{96}. Cependant les résultats sont insatisfaisants dans le cas ALN_{96} et ALN_{96} pseudo-dyadique. Comme la méthode AL domine celle utilisant AL pseudo-dyadique, cette dernière ne sera plus considérée. Ainsi seule la méthode ALN est en mesure de concurrencer la méthode AL.
Il est à noter qu'avec 25 locuteurs, les performances utilisant le critère entropie sont identiques à celles utilisant le critère énergie.

Dans tous les résultats qui suivent, nous allons utiliser uniquement le critère d'énergie.

Le tableau XIII présente une comparaison des performances obtenues en utilisant ALN_{96} et AL_{96} pour 60 locuteurs.

Tableau XIII

Performances de ALN_{96} et AL_{96} (60 locuteurs).

	SC(%)	FR(%)	FA(%)
ALN_{96}	87.4	0.1	12.5
AL_{96}	97.1	2.9	0.0

L'efficacité du système VITD utilisant ALN_{96} se dégrade de manière notable. Cependant le système présente une meilleure performance avec la méthode AL_{96}.

Au cours de ces expériences, d'importantes constatations ont pu être établies :

- Le nombre de nœuds de l'arbre ALN_{96} est beaucoup plus grand que celui de AL_{96}
- Le système VITD reconnaît de façon efficace un locuteur à partir d'un nombre de nœuds relativement réduit par rapport au cas d'un locuteur non reconnu.
- Les probabilités d'une reconnaissance efficace d'un locuteur par la méthode ALN_{96} semblent être très élevées.
- Les probabilités de reconnaissance étant classifiées par ordre décroissant, un écart de probabilité important entre les deux meilleures probabilités caractérise une reconnaissance efficace du locuteur.

En tenant compte de ces constatations, nous avons essayé de réduire le nombre de nœuds de l'arbre ALN en appliquant l'ACP dans MSAAB en considérant un taux de variance moins élevé.

Le tableau XIV montre l'utilisation de ALN avec différentes valeurs de TVE pour 5, 25 et 60 locuteurs.

Tableau XIV
Performances du système VITD
(5, 25 et 60 locuteurs).

	TVE	SC(%)	FR(%)	FA(%)
ALN	80	100%	0	0
	70	**100%**	0	0

Comme nous l'avons prévu, la réduction du TVE donne une efficacité de 100% de réussite aussi bien pour 5 et 25 que pour 60 locuteurs.

Remarque

Lorsque la valeur de TVE est inférieure à un certain seuil minimal (pour lequel le nombre de nœuds est 12), la méthode AL_{TVE} ne peut être appliquée pour tous les locuteurs.

La table XV donne un aperçu sur les performances du système VITD utilisant les différentes méthodes d'analyse.

Tableau XV

Performances des différentes méthodes (60 locuteurs).

	ALN_{70}	MFCC	MFDWC	BBS
FR(%)	100.0	0.7	0.9	0.2
FA(%)	0.0	2.5	13.7	55.5
SC(%)	0.0	96.8	85.5	44.4

Ce tableau montre l'efficacité de la méthode proposée comparée à l'analyse de Fourier (MFCC) ainsi que la méthode fondée sur le critère de l'entropie minimale (Best Basis Select (BBS)) et la méthode utilisant l'approximation de l'échelle de Mel (MFDWC Mel Frequency Discret Wavelet Coefficients).

Le Tableau XVI donne un aperçu sur les écarts de probabilité obtenus par ALN comparés aux autres types de paramétrisation.

Tableau XVI

Statistiques illustrant les scores de probabilités pour Yoho.

	TVE(%)	μP_{BS1}(log)	ΔP_{BS} (log)
ALN	96	-16.1	0.9
	80	-12.9	2.1
	70	**-11.3**	**3.1**
MFCC	-	-55.6	0.3
MFDWC	-	-28.4	0.5
BBS	-	-36.0	0.2

$\mu P_{BS1} = (1/N) \Sigma P_{BS1}$ avec P_{BS1} moyenne de la meilleure probabilité.

$\Delta P_{BS} = (1/N) \Sigma |P_{BS1} - P_{SB2}|$, moyenne de la différence entre les deux meilleurs scores.

Ce tableau montre l'accentuation de l'écart de probabilité du ALN_{70} comparé à toutes les autres méthodes concurrentes.

L'efficacité du ALN_{70} fait de lui le candidat retenu pour la suite des expériences.

4.6.1.2 Expérimentation utilisant le corpus Spidre

Les expériences ont été réalisées pour différentes durées du signal, à savoir 2, 4 secondes et ainsi que le signal en entier. Les résultats sont présentés dans le Tableau XVII.

Tableau XVII

Performances de VITI utilisant MFCC, BBS, MSAAB et MFDWC.

	ALN_{70}	MFCC	BBS	MFDWC
all	**100%**	84.4%	91.2%	76.6%
4sec	**100%**	64.5%	79.2%	58.6%
2sec	**99.8%**	56.8%	69.6%	55.2%

En utilisant une durée de 2 secondes du signal de test, le système VITI utilisant ALN_{70} ne commet que 2 fausses acceptations.

Le Tableau XVIII donne un aperçu sur les écarts de probabilités obtenus par ALN_{70} comparés aux autres types de paramétrisation.

Tableau XVIII

Statistique illustrant les scores de probabilités pour Spidre.

		μP_{BS1} (log)	ΔP_{BS} (log)
ALN_{70}	all	-12.2	**7.48**
	4sec	-12.3	7.46
	2sec	-12.3	7.35
MFCC	All	-60.6	0.28
	4sec	-60.5	0.27
	2sec	-60.5	0.25
BBS	All	-19.2	2.60
	4sec	-19.0	2.08
	2sec	-19.0	2.07

Les statistiques faites sur les probabilités de reconnaissance (Tableau XIX) montrent une meilleure discrimination des locuteurs. L'écart des meilleurs scores de probabilité est de valeur moyenne de 7.48 dans le cas ALN_{70}, alors qu'il est de 2.60 dans le cas BBS et de valeur nettement faible (0.28) dans le cas MFCC.

4.6.2 Deuxième partie : combinaison de paramètres

Le corpus Yoho est utilisé pour tester les performances du système de vérification par combinaisons de l'analyse de Fourier et celui des ondelettes.

Dans le cas où le modèle HMM est construit à partir d'un nombre de streams égal à deux, nous avons réalisé les tests suivants :

Tableau XX

Combinaison de l'analyse de Fourier et d'ondelettes utilisant 2 streams statiques de poids égaux.

Type de paramètre	Nombre de locuteurs	score	FR	FA
12MFCC+12MFDWCC statique	5	96.5	3.5	0.0
12MFCC+12MFDWCC statiques	20	94.4	5.3	0.4

Avec 3 streams, les résultats obtenus pour l'estimation des poids qui permet une meilleure combinaison des deux analyses est donnée par le tableau suivant :

Tableau XXI

Combinaison de paramètres : 12 statiques et 12 dynamiques MFCC avec 12 statiques MFDWC (20 locuteurs).

Poid	FR	FA	Performance (%)
[2.2 1.2 0.8]	27	1	96.50
[2.2 1.6 0.8]	28	1	96.37
[2.2 2 0.8]	28	1	96.37
[2.2 1 0.8]	**24**	**1**	**96.88**
[2 1 0.8]	27	1	96.50
[1 1 0.8]	35	1	95.50
[1 2 1]	39	2	94.57
[1 1.2 1]	35	1	95.50
[1 1 1]	33	1	95.75
[1 2.9 1]	33	2	95.63

La combinaison des deux analyses permet d'améliorer les performances, et d'éliminer les fausses acceptations. Les résultats sont présentés par le tableau suivant

Tableau XXII

Combinaison de MFCC et BBS utilisant les poids optimaux [2.2 1 0.8] (60 locuteurs).

	FR(%)	FA(%)	SC(%)
MFCC	1.8	0.5	97.7
BBS	6.5	2.3	91.2
MFCC+BBS	1.4	0	98.6

4.7 Conclusion

Dans ce chapitre des expériences utilisant deux types de corpus ont été utilisés : des données propres et bruitées. Les résultats obtenus sont plus qu'encourageant. Ceci confirme la robustesse des algorithmes proposés dans ce projet de recherche.

En raison de la chronologie de la recherche, les simulations utilisant la combinaison de paramètres ont été réalisées avant la découverte et la mise au point de l'algorithme MSAAB. Une fois ce dernier développé, il a donné de meilleure performances que celles espérées (avec un 100% d'efficacité). La combinaison du MSAAB avec une quelconque autre méthode n'aurait aucun intérêt à ce stade de l'expérimentation.

CONCLUSION

Dans ce travail nous avons tester et comparer 4 méthodes pour caractériser le timbre vocal d'un locuteur en vu de la vérification de son identité. Par application de ces 4 méthodes, les informations caractéristiques du locuteur présentes dans le signal de parole sont obtenus : a) par application de l'algorithme MFCC (Mel frequency cepstrals coefficients), b) à partir d'une analyse à base d'ondelettes. Cette dernière est utilisée pour les techniques suivantes : MFDWC (Mel Frequency Discret Wavelet Coefficients), BBS (Best Basis Select) et enfin celle qui représente la contribution de ce travail appelée la Meilleure Structure d'Arbre Abstrait (MSAAB). L'algorithme qui permet de mettre en marche chacune des ces méthodes est le même, mais utilisant un arbre admissible différent. La méthode MFDWC utilise l'échelle de Farooq obtenu par une approximation linéaire de l'échelle de Mel. BBS permet d'obtenir un arbre admissible propre à chaque locuteur selon le critère de l'entropie minimale.

La méthode MSAAB que nous avons proposé est similaire à celle utilisée dans le cas BBS. Celle-ci réalise une décomposition d'un signal vocal en paquet d'ondelettes de profondeur bien déterminée. La sélection des bandes de fréquences est effectuée selon le taux d'information qu'il représente dans un ensemble de bandes d'un même niveau de décomposition. La sélection des bandes de fréquences est assurée par l'application de l'analyse en composante principale, et faisant usage d'un taux de variance optimal. Les bandes de fréquences obtenues sont relative à chaque niveaux de décomposition.

À partir des résultats obtenus, nous avons constaté que les performances de reconnaissance de la méthode MSAAB dans le cas utilisant un arbre par locuteur générer à partir des niveaux (ALN) s'améliore lorsque nous utilisons un taux de variance moins important, cela s'explique du fait que seules les bandes de fréquences qui portent l'information pertinente sont retenues.

Dans le cadre de ce travail nous avons aussi traiter la combinaison de l'analyse de Fourier et celle des ondelettes. Les résultats obtenus sont satisfaisants car par cela nous avons pu réduire les fausses acceptation à zéro en combinant les paramètres MFCC avec les BBS en utilisant la base de donnée Yoho avec 60 locuteurs. Malheureusement nous n'avons pas pu tester cette technique avec notre algorithme proposé vu que celui-ci obtient une excellente performance avec les bases de données utilisées (Spidre et Yoho). Une augmentation du nombre de locuteur de la base de données risque de réduire les performances de notre système de vérification. Les différentes combinaisons possibles pourraient alors améliorer les résultats. Malheureusement le manque de temps ne nous a pas permit de réaliser cette étape de grande envergure.

La principale et importante contribution à la recherche à été la proposition d'une nouvelle méthodologie de segmentation fréquentielle. L'utilisation de cette méthode permet une meilleure exploitation des propriétés des ondelettes, et permet de caractériser un locuteur même en milieu bruité par des paramètres appropriés et discriminants. Une amélioration moyenne de 10%, 20% et 30% par rapport à MFCC, MFDWC et BBS respectivement a été obtenue. Cette technique présente une efficacité de performance supérieure à celle utilisant le critère d'entropie minimal. Mais une combinaison entre les deux méthodes peut mener à des performances plus brillantes. Nous avons vérifié expérimentalement que pour 25 locuteurs par application de cette nouvelle technique de segmentation les nœuds de l'arbre abstrait obtenus vérifient le critère de l'entropie minimal avec un seuil supérieur à 50%. Ce qui peut être encourageant pour de futures applications.

BIBLIOGRAPHIE

[1] Badri, N., Benlahouar, A., Tadj, C., Gargour, C., Ramachandran, V., *(2002), On the use of wavelets and Fourier transform for speaker verification*, 45th IEEE FWSCAS.

[2] Baum L., Petrie T., Soules G., and Weiss N., (1970), *A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains*, Ann. Math Statistics, vol. 41, pp. 164-171.

[3] Belaid, A., & Belaid, Y., (1992), *Reconnaissance des formes*, InterÉditions.

[4] Calliope, (1989), *La parole et son traitement automatique*, Paris : Masson.

[5] *Chao H., Tao C., Stan L., Eric C., & Jianlai Z., Analysis of speaker variability*, Eurospeech, 2001.

[6] Cheriet M., (2002), R*econnaissance de forme et inspection*, note de cours, Automne.

[7] Coifman, R. R., & Wickerhauser, M. V., (1992), *Entropy-based algorithms for best basis selection, IEEE Transactions on Information Theory, 38*(2 pt II), pp. 713-718.

[8] Daubechies, I., (1992), *Ten lectures on wavelets*, Philadelphia, Pa.: Society for Industrial and Applied Mathematics.

[9] Daubechies, I., (1988), *Orthonormal bases of compactly supportedd wavelets, Commun.* Pure Appl. Math., 91, pp. 909-996.

[10] Erkki Oja, (1983), *Subspace methods of pattern recognition*, volume 6 of Pattern recognition and image processing series. John Wiley & Sons.

[11] Farooq, O., & Datta, S., (2001), *Mel filter-like admissible wavelet packet structure for speech recognition, IEEE Signal Processing Letters, 8*(7), pp. 196-198.

[12] Gish, H., & Schimdt, M., (1994), *Text-Independent speacker recognition, IEEE Signal Processing Magazine*, pp. 18-32.

[13] Higging A.L. et. al., (1991), *Speaker verification using Randomized phrase prompting, Digital Signal Processing*, Vol. 1, pp. 89-106.

[14] Kuhn, R., Juanqa J.C., Nguyen & Nieedzielski. N, (2000), Rapid speaker a*daptation in eigenvoice space, IEEE Trans, 8*(7), pp. 196-198.

[15] Mallat S., (1989), *A theory for multiresolution signal decomposition : The wavele representation*, IEEE Trans. Patt. Anal Machine Intell., 11(7), pp. 674-693.

[16] Mallat, S. (1998), *A wavelet tour of signal processing*. New York: Academic Press.

[17] Mallat, S., (1989), *Multiresolution approximations and wavelet orthonormal bases of* $L^2(R)$, Trans. Amer. Math. Soc. 315, pp. 69-87.

[18] Matsui T. and Furui S., (1994), *A new similarity normalization method for speaker verification based on a posterioti probability*, ESCA Workshop on automatic speaker recognition, identification and verification, pp. 59-62.

[19] Rabiner L. R., (1989), *A tutorial on hidden markov models and selected applications in speech recognition,* Proceeding of IEEE, vol. 77, no 2.

[20] Rafael C. Gonzalez and Richard E. Woods, (1992), *Digital image processing,* Addison Wessley Publishing Company.

[21] Rundy.k., Young, (1993) *Wavelet theory and its applications*, Boston, Mass. : Kluwer Academic.

[22] Sarkar, T. K., et al. (1998). *Tutorial on wavelets from an electrical engineering perspective, part 1: discrete wavelet techniques*. IEEE Antennas and Propagation, vol. 40, No. 5.

[23] Young S.J., Woodland P.C. , and Byrne W.J., (1993), *HTK: Hidden Markov Model Toolkit V1.5*, Cambridge University Engineering Department Speech Group and Entropic Research.