SEGMENTATION DE SIGNAUX AUDIO : UNE NOUVELLE APPROCHE UTILISANT LE CRITERE D'ALIGNEMENT

Lamya Fergani, Belkacem Fergani et Amrane Houacine
Université des sciences et de la technologie Houari Boumédiène (USTHB)
Faculté d'Electronique et d'Informatique, **L**aboratoire de **C**ommunication **P**arlée et de **T**raitement du **S**ignal
BP 32, El alia, Alger, ALGERIE
E-mail : lamifer@msn.com, bfergani@gmail.com, a-houacine@lycos.com

RESUME

La discrimination de classes sonores dans un système d'indexation audio est indispensable et conditionne les performances de celui-ci. En effet, étant donné la complexité de la bande sonore d'un document audiovisuel il est souvent recherché l'accès rapide à des bruits ou des évènements sonores particuliers comme des passages musicaux ou des locuteurs particuliers ou des mots clefs préétablis. Cet objectif fait appel à une étape préalable de discrimination classe/ non classe. Nous proposons dans cet article un algorithme original permettant la segmentation semi supervisée de signaux audio. Cet algorithme met en œuvre une **A**nalyse en **C**omposantes **P**rincipales (ACP) combinée avec le critère d'alignement de noyaux introduit en apprentissage statistique. Cet algorithme ne nécessite pas une modélisation des données ni aucune connaissance préalable du contenu des fichiers audio analysés. Les résultats obtenus sur une base de données de sons radiodiffusés multi sources montrent clairement la pertinence de cette approche. Sa simplicité de mise en œuvre et d'interprétation permettent la possibilité de son exploitation dans un processus de décision en ligne.

SUMMARY

In audio indexing systems it's always needed to access directly to the particular acoustical event like musical record or a speaker excerpt, then we must to a priori design a binary based audio algorithm which permit to segregates the acoustic classes. This paper addresses a new method which combine the classical Principal Component Analysis (PCA) with the Alignment criterion introduced and often used in machine learning problems. This new method is model free and easy computed, we demonstrate its achievement and show their promising results which in return permits their use on DSP and FPGA platforms.

1. INTRODUCTION

Le développement de systèmes d'indexation de bases de données audio est un domaine de recherches toujours en évolution et focalise davantage l'intérêt de la communauté scientifique. Ceci est motivé par l'accroissement et la diversité de sources multimédia. La mise en œuvre de ces systèmes nécessite souvent une étape préalable et cruciale de discrimination de classes acoustiques. Le suivi temporel de cette classification est la tâche reconnue comme la segmentation audio.

La bande sonore d'un document audiovisuel regroupe plusieurs types de signaux : parole, musique, parole+musique, jingles, bruits, etc. Selon le type d'application ciblé différentes segmentations sont envisageables :

- Segmentation musique /non musique pour la classification en genre ou par type d'instruments de musique [Tzanetakis 2002, Essid 2005],
- Séparation parole / fond musicale des segments parole+musique pour des applications de séparation de sources ou de mixage audio [Meignier 2004],
- Segmentation parole /non parole pour la transcription orthographique et de l'indexation audio [Lu 2001, Pinquier 2004, Meignier 2004].

Les systèmes de segmentation de l'état de l'art sont généralement basés sur une modélisation des données et font appel aux modèles de Markov cachés (HMM) [Rabiner 1989, O'Shaughenssy 2008], les modèles de Mélanges de Gaussiennes (GMM), les k plus proches voisins (KNN), les réseaux de neurones et plus récemment les Machines à Vecteurs de Support (SVM) [Pinquier 2004, Essid 2005, Fergani 2007]. Dans un contexte de traitement de la parole, une modélisation GMM est généralement adoptée. Bien que ces techniques aient prouvé leur efficacité et donnent de bonnes performances, elles sont néanmoins coûteuses en temps de traitement et en charges de calculs pour des signaux audio issues de bases de données professionnelles et sont souvent tributaires d'une adéquation avec un modèle mathématique difficilement conciliable.

Le but de cet article est de présenter un algorithme de segmentation original indépendant de la modélisation des données et facilement mis en œuvre permettant son éventuel implantation sur des processeurs spécialisés. Cet algorithme réalise une séparation classe / non classe puis une segmentation parole, musique, bruit d'un fichier audio. Le

reste de cet article est structuré comme suit, la section suivante présente un état de l'art global des méthodes de segmentation audio. Les sections 3 et 4 introduisent notre méthode afin que la section 5 en donne les détails de son algorithme enfin la section 6 est dédiée
aux différentes expériences et évaluations de la méthode. Nous terminerons par une conclusion et perspectives.

2. LA SEGMENTATION AUDIO : ETAT DE L'ART

Le flux audio est généralement l'enregistrement d'un signal acoustique provenant de plusieurs sources sonores. Le document ainsi obtenu est constitué de plusieurs composantes dont les plus fréquentes sont la parole, la musique et le bruit. Cette dernière composante regroupe en fait toutes les composantes non identifiables à de la parole ou de la musique.

Ces documents audio sont généralement issues d'émissions radiophoniques ou télévisuelles et constituent le cadre général d'un système d'indexation audio. Cet environnement est caractérisé par une forte variabilité acoustique, on peut ainsi rencontrer de larges segments de parole en bande élargie ou en bande téléphonique, de très brefs segments de jingles et des segments de quelques minutes de musique instrumentale ou de la voix chantée.

Les méthodes de l'état de l'art décomposent ce problème en plusieurs classes : classes binaires parole /non parole, musique / non musique, classe bruit et classe silence.

Un système d'indexation se décompose généralement en deux étapes : une étape d'extraction de descripteurs et une étape de modélisation statistique de ces descripteurs. Dans une perspective de décomposition parole/non parole les descripteurs MFCC (Mel Frequency Cepstral Coefficients) sont généralement adoptés et donnent des résultats satisfaisants. Les méthodes de classification de l'état de l'art sont généralement basées sur modélisation des données telles que les méthodes par mélange de gaussiennes (GMM) ou les chaînes de Markov Cachées (HMM) [Duda 2001], des méthodes géométriques comme les k plus proches voisins, les histogrammes [Theodoridis 2003] ou les méthodes neuronales (RNN) et récemment les Méthodes par Vecteurs de Supports (SVM) représentant les méthodes de classification discriminatives [Burges 1998, Lu 2001 2002, Llorente 2005, Bishop 2006].

Les méthodes génératives telles que les GMM ont largement fait leur preuve avec succès dans beaucoup d'applications pratiques [Seck 2001, Ajmera 2004, Ulusay 2006], bien quelles soient aujourd'hui surpassées par les méthodes discriminatives en terme de performances de classification. Ceci est principalement due aux caractéristiques des données modernes : la grande dimension et le nombre d'exemples limité. En effet les méthodes génératives sont particulièrement sensibles à ces deux facteurs, alors que les méthodes discriminatives ne sont généralement pas car opérant dans un espace de grande dimension. Actuellement, les méthodes discriminatives sont donc particulièrement utilisées pour résoudre les problèmes modernes car elles fournissent des résultats quantitativement très bons mais au prix d'une faible adaptabilité et de charges de calculs assez lourds [Ulusay 2006, Herbrich 2002]. En particulier, les méthodes discriminatives à noyaux, de type SVM, sont très utilisées pour le traitement du signal, de l'image et du traitement de la parole [Fergani 2007, Herbrich 2002] car elles fournissent d'excellents résultats nonobstant les inconvénients cités en amont.

3. UNE METHODE HYBRIDE : ACP ET LE CRITERE D'ALIGNEMENT

Cet algorithme est basé sur une analyse du signal audio en composantes principales afin d'estimer la direction de variance maximale des données et d'en extraire les structures principales. Le critère d'alignement permet de choisir un hyperplan optimal permettant la discrimination des classes sonores. Avant de décrire plus explicitement cet algorithme, nous pensons qu'il est plus pédagogique de résumer les notions essentielles du critère d'alignement développé et exploité essentiellement dans les méthodes à noyaux [Cristiannini 2002, Kandolla 2002].

4. LE CRITERE D'ALIGNEMENT

Soit $D = \{(x_1, y_1), (x_2, y_2), \ldots (x_m, y_m)\}$ un ensemble de m données d'apprentissage et leurs étiquettes associés. L'apprentissage statistique inductif permet d'exploiter la relation existante entre la distribution géométrique des données et celle des étiquettes de classes. Dans ce cadre, il est aussi démontré que la matrice de Gram K ou matrice noyau, symétrique semi-definie positive permet de caractériser les ressemblances (ou dissemblances) entre les paires de données d'entrée en rapport avec la métrique induite par une fonction noyau [Cristiannini 2002, Kandolla 2002, Vert 2004]. Chaque élément k_{ij} est le résultat de l'application d'une fonction noyau à la paire de donnée (x_i, x_j).D'autre part, si nous considérons que les étiquettes de classes notées (± 1)sont contenues dans un vecteur colonne y, nous pouvons définir la matrice $Y = y \times y^T$. Y est une matrice symétrique de même dimension que la matrice K telle que : $y_{ij} = +1$ si $y_i = y_j$ et $y_{ij} = -1$ si $y_i \neq y_j$. Ainsi Y résume toute l'information véhiculée par les données d'apprentissage représentés par leurs étiquettes. Puisque K représente la similarité géométrique des données et T représente la similarité des étiquettes de ces données, il est raisonnable de penser que ces deux matrices présentent aussi des ressemblances entre elles. Cette propriété est alors mesurée et quantifiée par le critère d'alignement. Celui-ci n'est en fait qu'une généralisation aux matrices, du produit scalaire normalisé de deux vecteurs (x , y) :

$$\cos(\varphi) = \frac{x^T y}{\sqrt{x^T x * y^T y}} \qquad (1)$$

Dans le cas de deux matrices A et B cette généralisation traduit le critère d'alignement qui s'écrit :

$$\mathrm{A}(A,B)=\frac{\langle A,B\rangle_F}{\sqrt{\langle A,A\rangle_F}*\sqrt{\langle B,B\rangle_F}} \quad (2)$$

Le terme $\langle A,B\rangle_F$ signifie le produit scalaire généralisé aux matrices et l'indice F traduit la norme de Frobenius de la matrice A. Ainsi la formule précédente de l'alignement se réécrit comme :

$$\mathrm{A}(A,B)=\frac{\langle A,B\rangle_F}{\left(\|A\|_F*\|B\|_F\right)^{\frac{1}{2}}} \quad (3)$$

De ce fait pour mesurer l'alignement entre la matrice des données K et la matrice Y des données d'apprentissage étiquetés, nous évaluons la quantité $\mathrm{A}_{k,d}(Y,K)$ telle que définie par l'expression (2) ou (3).

$$\mathrm{A}_{k,d}(Y,K)=\frac{\langle Y,K\rangle_F}{\left(\|Y\|_F*\|K\|_F\right)^{\frac{1}{2}}} \quad (4)$$

5. L' algorithme de segmentation acp_alignement

Etant donné $D=\{(x_1,y_1),(x_2,y_2),\ldots(x_m,y_m)\}$ un ensemble constitué de m données d'apprentissage étiquetées (exemple d'un segment de signal constitué de deux classes sonores : parole + musique). Il s'agit d'évaluer la matrice de covariance des données puis d'en extraire les valeurs propres et vecteurs propres. On considère alors seulement le premier vecteur propre correspondant à la plus grande valeur propre. On détermine par la suite **l'hyperplan optimal** perpendiculaire à la direction principale donnée par le premier vecteur propre en ayant recours au critère d'alignement entre la matrice des étiquettes de données d'apprentissage et celle des étiquettes estimées par la fonction de décision. La maximisation de cet alignement permet d'évaluer le paramètre optimal de l'hyperplan puis d'en déduire ainsi la valeur du vecteur étiquette des données test et par conséquent la classe recherchée de données inconnues analysés. Suite à l'étape de paramétrisation, nous obtenons le signal des descripteurs suivant : $X=\{x_1^{\,d},x_2^{\,d},\ldots,x_m^{\,d}\}$ l'ensemble de données exemples de d paramètres (MFCC) avec $d<<m<<n$ taille du signal test.

Etape 1 : Calculer la matrice de Covariance des données

$$M_X=Cov(X) \quad (5)$$

Etape 2 :Evaluer les valeurs propres et vecteurs propres associés : λ_i et $V_i \quad i\in\{1,..,d\}$

Etape 3 : Déterminer le premier vecteur propre correspondant à la plus grande valeur propre :

$$V_1=\max(V_i) \quad (6)$$

Etape4 : Déterminer Hyperplan Optimal H

$$x\in H \text{ si } x.V_1-\alpha_{opt}=0 \quad (7)$$

$$f(x)=y_{est}=sign(x.V_1-\alpha) \quad (8)$$

Faire varier le paramètre α revient à glisser l'hyperplan H le long de la direction de variance maximale des données (Figure 3). Soient y_{app} le vecteur étiquette de données apprentissage et y_{test} le vecteur étiquette estimée de la donnée apprentissage :

Pour $\alpha=\alpha_1\rightarrow\alpha_2$

Calculer $y_{est}=sign(x.V_1-\alpha)\ \forall x\in D$

Evaluer, $Y_{est}=y_{est}*y_{est}^{\ t}$ et $Y=y_{app}*y_{app}^{\ t}$

Calculer $\mathrm{A}_\alpha(Y_{est},Y)=\frac{\langle Y_{est},Y\rangle_F}{\sqrt{(\|Y_{est}\|_F*\|Y\|_F)}}$ (9)

Fin Pour

Evaluer $\mathrm{A}_{\max}(\alpha)\rightarrow\alpha_{opt}$ (le max. de A en fct. de α)

Soit x_{test} une donnée appartenant à l'ensemble de données test (autre fichier que celui ayant servi pour l'entraînement) on détermine alors son étiquette y_{test} en ayant recours simplement à la fonction de décision

$$y_{test}=sign(x_{test}.V_1-\alpha_{opt}) \quad (10)$$

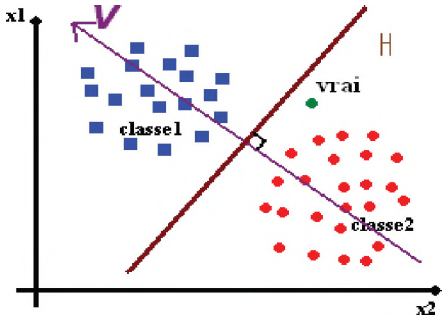

Figure 1. Exemple de données linéairement séparables

5.1 Cas de données linéairement séparables

Dans ce cas (Figure 1), nous avons une marge où l'alignement est maximum, et égale à la valeur maximale qui vaut 1. C'est-à-dire que le pourcentage des données classées faux est nul. La largeur de cette marge égale à (α_2-α_1), elle représente l'équivalent de la marge maximale recherchée par les SVMs et (H_1) : (x•V- α_1), (H_2) : (x•V- α_2) les deux hyperplans canoniques respectivement. α optimal correspond au milieu de la marge de séparation $\alpha_{opt}=(\alpha_2+\alpha_1)/2$, d'où ($H_{opt}$) = (x•V- α_{opt}) est l'hyperplan optimal recherché (Figure 2).

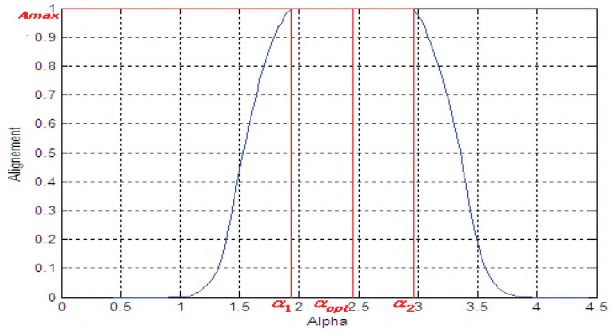

Figure 2. Le paramètre α en fonction de l'alignement dans un cas séparable

5.2 Discussions sur le choix de α_1 et α_2

Soient X_1 et X_2 deux sous ensembles de X correspondants (par exemple) respectivement à la classe parole et musique. Soient $\alpha_1 = V_1 M_1$ et $\alpha_2 = V_1 M_2$ avec M_1 et M_2 resp. les centres d'inertie des nuages de points X_1 et X_2. Ainsi, le principe de notre algorithme ressemble à la philosophie des SVM dans le sens où on utilise un ensemble restreint de données exemples d<<m<<n (la matrice de covariance ne dépend que la taille des descripteurs et non de leurs nombre) afin d'estimer l'hyperplan optimal qui constitue la frontière de décision adéquate permettant de classer tout nouvel élément appartenant à l'une ou l'autre des classes discriminées. Cependant les charges de calcul de notre méthode sont nettement moindres que ceux des méthodes SVM et de par sa simplicité notre algorithme est beaucoup plus rapide permettant son exploitation dans un processus de décision en ligne.

5.3 Cas de données linéairement non séparables

Dans le cas où le nuage de points n'est pas séparable et où les données de deux classes distinctes sont équitablement distribuées (Figure 4), il y aura recouvrement entre les deux nuages ce qui entraîne des erreurs de classification. Dans ce cas, nous n'avons qu'une valeur où l'alignement est maximum, et égale à une valeur inférieure à 1, c'est-à-dire qu'il y a des données qui ont été mal classées , et le taux de recouvrement entre les deux classes égal à 1-A_{max} (Dans l'exemple précédent est de 32%). Cette valeur α_{opt} est optimale, d'où (H_{opt}) est l'hyperplan de classification optimal telle que $f(x) = sign(x.V_1 - \alpha_{opt})$ (Figure 5).

6. Expériences et Résultats

6.1 La base de données

Afin d'évaluer notre méthode nous avons construit une base de données de signaux contenant diverses composantes de mélanges sonores (parole + musique + bruit + silence) divisée en trois parties : Une base de référence contenant les fichiers sonores exemples, une base de fichiers dédiée au développement de la méthode et finalement une base de signaux test afin d'estimer objectivement les performances. Leur description est détaillée dans le tableau suivant (Tableau 1) :

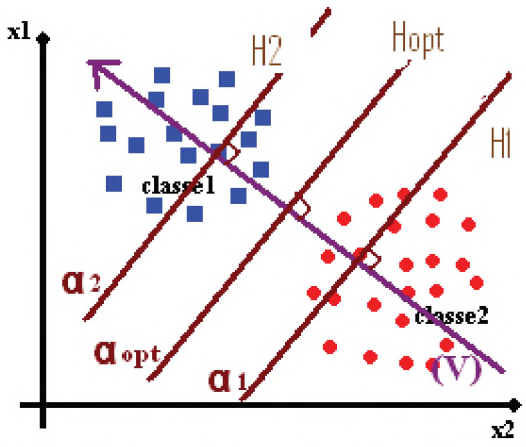

Figure 3. L'influence du paramètre α sur l'hyperplan séparateur.

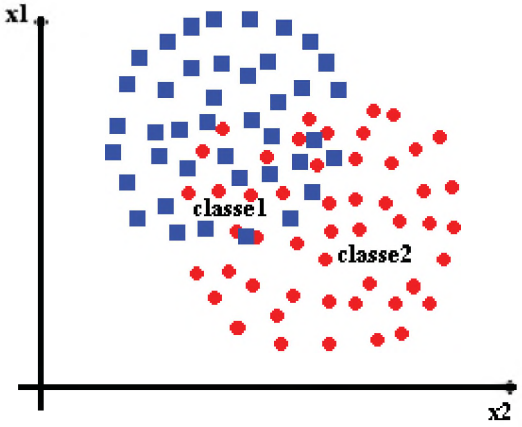

Figure 4. Exemple de données linéairement non séparables

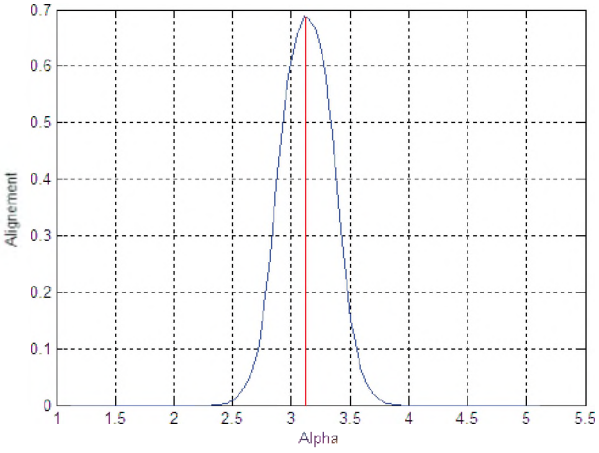

Figure 5. Le paramètre α en fonction de l'alignement dans un cas non séparable.

Fichier	Durée (s)	%M	%P	%S	%B
Mref.wav	-	100	0	0	0
Pref	-	0	100	0	0
Sref	-	0	0	100	0
Bref	-	0	0	0	100
Dev1	300	35	50	5	10
Dev2	600	50	35	10	5
Dev3	900	0	90	5	5
Dev4	1200	90	0	5	5
Test1	1800	25	30	20	25
Test2	2700	50	20	15	15

Tableau 1. Description de la base de données

6.2 Le Critère de Performance

Le critère de performance adopté est la mesure F définie comme étant la combinaison de deux métriques appelés Précision et Rappel empruntées à la RI (Information Retrieval) [Pinquier 2004, Mauclair 2006] : Le Rappel mesure la capacité d'un système à sélectionner les hypothèses pertinentes :

$$\text{Rappel}_i = \frac{\textit{Trames correctement attribués à la classe } i}{\textit{Nombre de trames appartenant à la classe } i}$$

La précision mesure la capacité du système à rejeter les hypothèses non pertinentes :

$$\text{Précision}_i = \frac{\textit{Trames correctement attribués à la classe } i}{\textit{Nombre de trames attribué à la classe } i}$$

La combinaison harmonique de ces deux métriques donne F mesure:

$$F = \frac{2 * \textit{précision} * \textit{rappel}}{\textit{précision} + \textit{rappel}} \qquad (11)$$

Les fichiers sonores sont échantillonnés à 35Khz et la paramétrisation est effectuée par l'outil HTK Tools [Young 2002]. Nous avons choisi d'utiliser les coefficients MFCC évalués au centi-seconde (fenêtre de Hamming de 32 ms) et dont le nombre est variable (Tableaux 4,5,6,7). Dans ce qui suit nous allons présenter les différentes expériences réalisées afin de quantifier les performances de notre méthode. La décision est prise trame par trame.

6.3 Expériences sur les signaux de développement

a- Influence de la normalisation de l'énergie

Ci-dessous sont présentes les résultats pour des fichiers sans normalisation de l'énergie (Tableau 2), puis avec normalisation de l'énergie (Tableau 3) :

Nous observons que les résultats s'améliorent sensiblement avec la normalisation de l'énergie.

F.mes / Fichier	Bruit	Silence	Musique	Parole
Dev1	83.22%	92.77%	76.32%	82.32%
Dev2	83.22%	95.16%	72.16%	87.12%
Dev3	79.34%	87.13%	Pas de Musique	89.67%
Dev4	87.32%	86.36%	91.17%	Pas de Parole.

Tableau 2. Données avec énergies non normalisées

F mes. / Fichier	Bruit	Silence	Musique	Parole
Dev1	93.55%	96.77%	97.17%	98.32%
Dev2	93.55%	95.16%	95.64%	95.24%
Dev3	85%	87.13%	P.M.	99.07%
Dev4	94.49%	86.36%	98.83%	P.P.

Tableau 3. Données avec énergies normalisées

b- Influence de la dimension des descripteurs

Dans l'expérience qui suit on fait varier la dimension des descripteurs MFCC et on observe l'évolution de F mesure correspondante à la détection de chaque classe. Les tableaux qui suivent illustrent les résultats en fonction de la variation des fichiers de développement :

Dim. / Fichier	8	12	16	20	24	30	35	40
Dev1	36.25	**93.55**	**93.55**	93.55	36.25	34.52	93.55	93.55
Dev2	16.57	**93.55**	**93.55**	93.55	16.57	16.20	93.55	93.55
Dev3	85	**85**	**85**	85	85	70.83	85	85
Dev4	10.25	**94.49**	**94.49**	10.25	10.32	10.32	94.49	94.49

Tableau 4. Influence la dimension des descripteurs : F mesure correspondant à la classe Bruit

Dim. / Fichier	8	12	16	20	24	30	35	40
Dev1	16.13	ND	**96.77**	**96.77**	19.87	25.21	96.77	ND
Dev2	35.19	8.82	**95.16**	**95.16**	40.27	47.97	95.16	8.82
Dev3	9.75	17.24	**87.13**	**87.13**	9.75	10.01	87.13	17.24
Dev4	86.36	15.79	**87.10**	**87.10**	86.36	86.36	86.36	15.79

Tableau 5. Influence la dimension des descripteurs : F mesure correspondant à la classe Silence

Dim. / Fichier	8	12	16	20	24	30	35	40
Dev1	ND	97.17	97.17	97.17	ND	ND	97.17	97.17
Dev2	ND	95.64	95.64	94.22	ND	ND	95.64	95.64
Dev3	PM	PM	PM	PM	PM	PM	PM	PM
Dev4	ND	98.83	98.83	98.83	98.8	1.47	98.83	98.83

Tableau 6. Influence la dimension des descripteurs : F mesure correspondant à la classe Musique

Dim. / Fichier	8	12	16	20	24	30	35	40
Dev1	ND	98.32	98.32	98.32	37.84	56.46	98.32	98.32
Dev2	ND	95.24	95.24	93.46	36.29	57.72	95.24	95.24
Dev3	1.22	98.7	98.7	99.22	1.22	3.14	99.07	98.19
Dev4	PP	PP	PP	PP	PP	PP	PP	PP

Tableau 7. Influence la dimension des descripteurs : F mesure correspondant à la classe Parole

c- Influence de la durée des fichiers de référence

Dans ces expériences on varie la durée des fichiers de référence et observe l'évolution de la F mesure correspondant à chaque classe. Nous rappelons que les fichiers de référence sont considérés comme des échantillons de classes sonores et qu'il est souhaitable que leur taille soit la plus petite possible afin de ne pas peser sur les charges de calculs de l'algorithme.

Durée(s) / Fichier	*10*	*20*	30	40	50
Dev1	***93.55***	***93.55***	36.25	34.52	93.55
Dev2	***93.55***	***93.55***	16.96	16.20	93.55
Dev3	***85***	***85***	85	65.38	85
Dev4	***94.49***	***94.49***	10.32	10.25	94.49

Tableau 8. Influence de la durée des fichiers de référence : F mesure correspondant à la classe Bruit

Durée (s) / Fichier	*10*	*20*	30	40	50
Dev1	***96.77***	***96.77***	16.13	16.85	96.77
Dev2	***95.16***	***95.16***	36.04	37.85	95.16
Dev3	***87.13***	***87.13***	9.75	10.01	87.13
Dev4	***87.10***	***87.10***	81.43	86.36	87.10

Tableau 9. Influence de la durée des fichiers de référence : F mesure correspondant à la classe Silence

Durée(s) / Fichier	*10*	*20*	30	40	50
Dev1	***93.64***	***97.17***	ND	ND	97.17
Dev2	***91.94***	***94.22***	ND	ND	94.37
Dev3	***PM***	***PM***	PM	PM	PM
Dev4	***98.83***	***98.83***	ND	ND	98.83

Tableau 10. Influence de la durée des fichiers de référence : F mesure correspondant à la classe Musique

Durée (s) / Fichier	*10*	*20*	30	40	50
Dev1	***95.50***	***98.32***	ND	ND	98.32
Dev2	***88.89***	***93.46***	14.10	14.10	93.20
Dev3	***87.54***	***87.70***	1.22	1.22	96.12
Dev4	***PP***	***PP***	PP	PP	PP

Tableau 11. Influence de la durée des fichiers de référence : F mesure correspondant à la classe Parole

d- Influence de la taille de la fenêtre de lissage

Suite à l'étape de classification un fichier de labels est crée. Il s'agit de regrouper d'abord les labels identiques adjacents, ensuite appliquer une fenêtre de lissage glissante dans le temps afin d'éliminer les segments issus de l'assemblage et qui sont non significatifs, c'est-à-dire dont la durée est inférieur à une durée minimale critique. Dans les expériences suivantes on fait varier la taille de la fenêtre de lissage et on relève la valeur de F mesure correspondant à la détection de chaque classe. L'objectif est d'ajuster ce paramètre à la valeur minimale qui donne un maximum de F mesure selon la classe considérée.

Taille(s) / Fichier	1	*3*	*5*	8
Dev1	PD	***96.67***	***96.67***	93.55
Dev2	PD	***96.67***	***96.67***	93.55
Dev3	PD	***90.24***	***89.41***	85
Dev4	PD	***97.56***	***97.56***	94.49

Tableau 12. Influence de la taille de la fenêtre de lissage : F mesure correspondant à la classe Bruit

Taille(s) / Fichier	1	*3*	*5*	8
Dev1	PD	***93.33***	***93.33***	96.77
Dev2	PD	***100***	***98.33***	95.16
Dev3	PD	***91.49***	***91.49***	87.13
Dev4	PD	***96.72***	***96.67***	87.10

Tableau 13. Influence de la taille de la fenêtre de lissage : F mesure correspondant à la classe Silence

Taille(s) / Fichier	1	3	5	8
Dev1	PD	PD	97.14	97.17
Dev2	PD	PD	97.33	95.64
Dev3	PM	PM	PM	PM
Dev4	PD	PD	99.44	98.83

Tableau 14. Influence de la taille de la fenêtre de lissage : F mesure correspondant à la classe Musique

Taille(s) / Fichier	1	3	5	8
Dev1	PD	PD	98.01	98.32
Dev2	PD	PD	96.21	95.24
Dev3	PD	PD	98.44	98.7
Dev4	PP	PP	PP	PP

Tableau 15. Influence de la taille de la fenêtre de lissage : F mesure correspondant à la classe Parole

Des expériences de développement qui précèdent nous concluons qu'une paramétrisation MFCC au nombre de 16 avec énergie normalisée est celle qui donne de meilleures performances quant à la descrimination des classes (Fmesure élevé) pour l'ensemble des fichiers de développement. Le choix du type de paramétrisation et du nombre de paramètres peut amener à ne pas détecter la classe ciblée (ND :Non Détection). D'autre parts, une taille moyenne de 10 ou 20 secondes des fichiers de référence donne des performances acceptables alors qu'une fenêtre de lissage de 5 secondes est la plus appropriée. En effet pour des longueurs de fenêtre de filtrage inférieur à cette valeur critique on peut rater la détection d'une classe (PD : Pas de Détection, PP : Pas de Parole et PM ;pas de Musique). De ce fait, les expériences sur les fichiers de développement ont permis de mettre en évidence les paramètres optimaux de la méthode eu égard aux fichiers sonores de la base de donnée considérée. Ce paramètres sont alors : Une paramétrisation acoustique MFCC de 16 paramètres avec énergie normalisée, une taille de fichiers de référence de 10 secondes et une taille de la fenêtre de lissage de 5 secondes.

6.4 Expériences sur les signaux de Test

Dans cette partie nous considérons deux fichiers sonores de taille et complexité plus importants par rapport aux fichiers de développement (Tableau 1).

Ces fichiers serviront à la validation de la méthode en tenant compte de l'ajustement des paramètres effectué lors de la phase de développement. Le critère de performance adopté dans ce cas est la matrice de confusion. Les tableaux ci-dessous (Tableaux 16, 17) traduisent les résultats de la méthode pour le fichier Test1.wav avec les paramètres suivants :(D=16 Coefficients MFCC avec normalisation d'énergie ; la durée des fichiers de référence est de 20 secondes ; la taille de la fenêtre de lissage est de 5 secondes avec un recouvrement de 50%). Ces résultats confirment clairement la segmentation correcte des classes sonores.

Classe Estimée / Classe réelle	Bruit	Silence	Musique	Parole
Bruit	**89.41**	2.83	3.53	4.23
Silence	2.66	**94.70**	2.64	0
Musique	1.98	0.57	**96.23**	1.12
Parole	2.65	0	2.23	**95.12**

Tableau 16. Matrice de confusion pour le fichier Test1.wav

Classe Estimée / Classe réelle	Bruit	Silence	Musique	Parole
Bruit	**88.36**	3.33	4.03	4.23
Silence	3.26	**94.02**	2.74	0
Musique	0.98	0.57	**97.23**	1.12
Parole	2.77	0	3.23	**94.00**

Tableau 17. Matrice de confusion pour le fichier Test2.wav

7. Conclusions et perspectives

Nous venons de présenter dans cet article de recherche une méthode originale de segmentation sonore permettant une indexation audio en classes multiples (parole, musique, bruit et silence). Cette méthode de classification semi supervisée fait appel à une méthode combinant l'analyse en composantes principales et le critère d'alignement de noyaux introduit et souvent exploité dans l'apprentissage statistique [Cristiannini 2002, Vert 2004, Kandola 2002]. Cette contribution constitue une des rares applications de ce critère en traitement de signaux audio.

Nous avons montré la faisabilité de cette méthode et évalué ces performances sur une base de signaux sonores issues d'enregistrements radiodiffusés dont nous avons varié les durées et le contenu. Les résultats établis sont prometteurs et permettent d'envisager son utilisation alternative aux autres méthodes d'indexation de l'état de l'art [Essid (2005), Fergani (2007)]. Son principal avantage par rapport aux méthodes génératives et discriminatives (GMM,HMM et SVM par exemple) est son indépendance vis-à-vis de la modélisation des classes et sa faible charge de calculs qui traduit par conséquent une rapidité de mise en œuvre puis de décision, ce qui permet d'envisager son implantation sur des processeurs DSP ou FPGA et par conséquent son exploitation dans un processus de décision en ligne. Un travail en perspective concerne l'adéquation de la méthode avec les descripteurs acoustiques autres que les MFCC ainsi que sa robustesse par rapport au recouvrement temporel de classes sonores.

BIBLIOGRAPHIE

Ajmera J. (2004), ''Robust Audio Segmentation''
PhD Thesis, Ecole Polytechnique Fédérale de Lausane (EPFL).

Bishop M. C. (2006), '' Pattern Recognition and Machine Learning'' Springer, Jordan M., Kleinberg J., Schölkopf B. Eds.

Burges C. (1998), ''A tutorial on Support Vector Machines for pattern recognition'' Data Mining and Knowledge Discovery 2 (2).

Cristiannini N., Shaw-Taylor J., Elisseef A. and Kandola J. (2002). '' On kernel -target alignement '' Advances in Neural Informations Processing Systems, Vol.14, pp. 367-373.

Duda R.O.,Hart P.E., Stork G. (2001) ''Pattern Classification'' 2nd Ed.,Weily,NY.

Essid S.. (2005). '' Classification automatique de signaux audio-fréquence : reconnaissance des instruments de musique '',PhD Thesis, Université Pierre et Marie Curie .

Fergani B. (2007) '' Application des Méthodes à Vecteurs de Support pour l'indexation en locuteurs de documents audio'' PhD Thesis Université Technique d'Alger (USTHB).

Fergani B., Davy M., Houacine A., (2007) ''Segmentation en locuteurs de documents audio : Une nouvelle approche basée sur les méthodes à vecteurs de support mono classe'', Canadian Acoustics, Vol 35 N°4, pp. 3-10.

Herbrich R. (2002), ''Learning Kernel Classifiers: Theory and Algorithms'' MIT Press, Cambridge USA.

Kandola J., Shaw-Taylor J. and Cristianini N., (2002) "On the extensions of kernel alignement'', Dept. of Computer Science, University of London,Tech. Rep. 120

Llorente Godino J. I. and Co (2005), ''Discriminative methods for the detection of voice disorders'' NoLisp, April 2005.

Lu L, Li S. and Zhang H.-J. (2001), ''Content-based audio segmentation using Support Vector Machines'' ACM Multimedia Conference, Canada.

Lu L. and Zhang H.-J. (2002), ''Content analysis for audio classification and segmentation'' IEEE Trans. SAP 10(7):504-516.

Mauclair Julie (2006), ''Mesures de confiance en traitement automatique de la parole et applications'', PhD Thesis Université du Maine, Laboratoire d'Informatique de l'Université du Maine.

Meignier S., Moraru D., Fredouille C., Besacier L., Bonastre J.F. (2004) '' Benefits of prior acoustic segmentation for automatic speaker segmentation'' IEEE ICASSP 2004, Montreal Canada.

O'Shaughenssy D. (2008), ''Automatic Speech Recognition: History, Methods and Challenges'' Pattern Recognition , Elsevier.

Pinquier J.. (2004). '' Indexation Sonore: recherche de composantes primaires pour une structuration audiovisuelle '',PhD Thesis, Université Paul Sabatier , Toulouse III.

Rabiner L. (1989), ''A tutorial on hidden markov models and selected applications in speech recognition'' Proc. IEEE 77(2): 257-286.

Seck M., Magrin-Chagnolleau I. and Bimbot F., (2001) ''Experiments on speech tracking in audio documents using GMM'' IEEE ICASSP USA.
Theodoridis S. and Koutroumbas K. (2003), ''Pattern Recognition, 2nd Ed.'', Elsevier.

Tzanetakis G. and Cook P. (2002) ''Musical Genre Classification of Audio Signals'' IEEE Transactions on Speech and Ausio Processing, 10(5):293-302.

Ulusay I., Bishop M. C. (2006), ''Comparison of Generative and Discriminative Techniques for Object Detection and Classification'' LNCS4170 Springer-Verlag.

Vert J.P., Tsuda K., Schölkopf B. (2004). '' A primer on kernel methods'' Kernel Methods in Computational Biology, Schölkopf B, Tsuda K and Vert J.P. Eds. Cambridge MA: the MIT Press, pp.35-70.

Young S. and all (2002), ''The HTK Book" Ver.3.2.1