Etude de techniques de classement "Machines à vecteurs supports" pour la vérification automatique du locuteur

Jamal Kharroubi

Thèse

Présentée pour obtenir le grade de docteur de l'Ecole nationale supérieure des télécommunications

Jamal Kharroubi

Etude de Techniques de Classement " Machines à Vecteurs Supports " pour la Vérification Automatique du Locuteur

Soutenue le 03 juillet 2002 devant le jury composé de

Jean-Paul Haton	Président
Younès Bennani	Rapporteurs
Christian Wellekens	
Bernadette Dorizzi	Examinateurs
Dijana Petrovska-Delacretaz	
Laurence Likforman-Sulem	
Marc Sigelle	
Gérard Chollet	Directeur

Ecole Nationale Supérieure des Télécommunications

A toute ma famille
et mes amis...

Remerciements

Je remercie tous ceux qui ont contribué de près ou de loin à ce travail, en particulier :

Mon directeur de thèse, *Gérard Chollet*, de m'avoir encadré et dirigé attentivement mes travaux de recherche.

Tous les membres de jury : *Jean-Paul Haton* qui m'a fait l'honneur de présider mon jury de thèse. *Christian Wellekens* pour avoir accepté de rapporter mon travail. Ma gratitude va tout particulièrement à *Younès Bennani* en tant que rapporteur de ce travail et surtout pour ces précieux conseils et son soutien aux moments ou j'en avais besoin. Je remercie également *Bernadette Dorizzi, Laurence Likforman-Sulem, Marc sigelle* et *Dijana Petrovska-Delacretaz* pour tous le temps qu'ils ont passé à lire et relire ce document de thèse et pour leurs conseils et remarques pertinentes.

Tous ceux qui j'ai pu côtoyer durant ce travail de thèse : les membres du consortium ELISA et qui sont devenus des amis (*Jeff, Ivan, corinne, Téva, Sylvain, Raphaël, Mohamadou* et *Fréderic*). Tous les membres du projet PICASSO qui a été à l'origine du financement de ma thèse et sans quoi ce travail n'aurai pas pu voir le jour. Tous les membres du projet BIOMET.

Tous les membres de département TSI et plus particulièrement le chef de département *Henri Maître*, *Patricia Friedrich, Catherine Vazza, Laurence Zelmar (LoLo)* et *Bahman Nabati* pour leurs aides et sympathies.

Tous mes collègues et amis du département (les membres du groupe labotsi) qui se reconnaîtront, *Monica et Guig*.

Mes amis : *Moulay, Karim, Ilias* et tous les autres d'être à mes côtés tout au long de cette aventure.

Toute ma famille pour leur soutien financier et moral dont je serai reconnaissant toute ma vie.

Résumé

Les SVM (Support Vector Machines) sont de nouvelles techniques d'apprentissage statistique proposées par V. Vapnik en 1995. Elles permettent d'aborder des problèmes très divers comme le classement, la régression, la fusion, etc... Depuis leur introduction dans le domaine de la Reconnaissance de Formes (RdF), plusieurs travaux ont pu montrer l'efficacité de ces techniques principalement en traitement d'image.

L'idée essentielle des SVM consiste à projeter les données de l'espace d'entrée (appartenant à deux classes différentes) non-linéairement séparables dans un espace de plus grande dimension appelé espace de caractéristiques de façon à ce que les données deviennent linéairement séparables. Dans cet espace, la technique de construction de l'hyperplan optimal est utilisée pour calculer la fonction de classement séparant les deux classes.

Dans ce travail de thèse, nous avons étudié les SVM comme techniques de classement pour la Vérification Automatique du Locuteur (VAL) en mode dépendant et indépendant du texte. Nous avons également étudié les SVM pour des tâches de fusion en réalisant des expériences concernant deux types de fusion, la fusion de méthodes et la fusion de modes.

Dans le cadre du projet PICASSO, nous avons proposé un système de VAL en mode dépendant du texte utilisant les SVM dans une application de mots de passe publics. Dans ce système, une nouvelle modélisation basée sur la transcription phonétique des mots de passe a été proposée pour construire les vecteurs d'entrée pour notre classifieur SVM. En ce qui concerne notre étude des SVM en VAL en mode indépendant du texte, nous avons proposé des systèmes hybrides GMM-SVM. Dans ces systèmes, trois nouvelles représentations de données ont été proposées permettant de réunir l'efficacité des GMM en modélisation et les performances des SVM en décision. Ce travail entre dans le cadre de nos participations aux évaluations internationales NIST.

Dans le cadre du projet BIOMET sur l'authentification biométrique mené par le GET (Groupe des Écoles de Télécommunications), nous avons étudié les SVM pour deux tâches de fusion. La première concerne la fusion de méthodes où nous avons fusionné les scores obtenus par les participants à la tâche "One Speaker Detection" aux évaluations NIST'2001. La seconde concerne la fusion de modes menée sur les scores obtenus sur les quatre différentes modalités de la base de données M2VTS.

Les études que nous avons réalisées représentent une des premières tentatives d'appliquer les SVM dans le domaine de la VAL. Les résultats obtenus montrent que les SVM sont des techniques très efficaces et surtout très prometteuses que ce soit pour le classement ou la fusion.

Abstract

The Support Vector Machines (SVM) are new techniques of the statistical learning theory proposed by Vapnick in 1995 and developed from the Structural Risk Minimization (SRM) theory . They belong to the family of universal learning machines that implement the strategy of keeping the empirical risk fixed and minimizing the confidence interval. The SVM's achieve the structural risk minimization inductive principle by mapping the input vectors into high-dimensional feature space using a non-linear transformation chosen a priory. In this space an optimal Separating Hyperplane is considered, and the goal is to minimize the bound on the generalization error of a model.

In this thesis, we studied the usability of the SVM as classification techniques for automatic Speaker Verification (SV) in text-dependent and text-independent modes. We also studied the SVM for fusion tasks by realizing some experiments concerning two fusion types : the fusion of methods and the fusion of modes.

Within the framework of the European project PICASSO, we proposed a new system for text-dependent SV using the SVM in an application of public passwords. In this system, a new modelization based on the phonetic transcription of the password was proposed to construct the input vectors of the SVM classifier. Concerning our studies of the usability of SVM in text-independent SV, we proposed three hybrid systems GMM-SVM. In these systems, three new data representations were proposed allowing to use the effectiveness of the GMM in modelization and the performances of the SVM in the decision. This work was in the framework of our participations to the international evaluations NIST.

Within the framework of the BIOMET project carried out by GET (Groupe des Écoles de Télécommunications) concerning the biometric authentication, we studied the SVM for two tasks of fusion. The first one concerns the fusion of methods where we used the scores obtained by ten different participants in the "One Speaker Detection" of NIST'2001 evaluations to take the final decision.

The second one concerns the fusion of modes using the scores of four different modalities of M2VTS database.

The studies that we realized represent one of the first attempts to apply the SVM in the SV field. The results obtained show that the SVM are very effective and promising techniques for both classification or fusion tasks.

Table des matières

Table des figures

Abréviations

DET	Detection Error Trade-off
DTW	Dynamic Time Warping
ERM	Empirical Risk Minimisation
FA	Fausse Acceptation
FR	Faux Rejet
GMM	Gaussian Mixture Models
HMM	Hidden Markov Models
HO	Hyperplan optimal
HTER	Half Total Error Rate
IAL	Identification Automatique du Locuteur
LLR	Log Likelihood Ratio
LPCC	Linear Prediction Cepstral Coefficients
M2VTS	Multi Modal Verification for Teleservices and Security applications
MAP	Maximum A Posteriori
MFCC	Mel Frequency Cepstral Coefficients
MLLR	Maximum Likelihood Linear Regression
MLP	Multi-Layer Perceptrons
NIST	National Institute of Standards and Technologies
PICASSO	Pioneering Caller Authentication for Secure Service Operation
RAL	Reconnaissance Automatique du Locuteur
RAP	Reconnaissance Automatique de la Parole
RdF	Reconnaissance de Formes
RN	Réseaux de Neurones
ROC	Receiver Operating Characteristic
SRM	Structural Risk Minimisation
SVM	Support Vector Machines

TEE Taux d'Égale Erreur
TFA Taux de Fausses Acceptations
TFR Taux de Faux Rejets
TIC Taux d'Identification Correcte
VAL Vérification Automatique du Locuteur
VQ Vector Quantization

Introduction

La parole est sans doute le moyen de communication le plus simple et le plus efficace chez les humains. Depuis le début de la recherche dans le domaine du traitement du signal, les chercheurs ont toujours eu une attention particulière pour le signal de parole. Grâce aux développements dans les domaines de l'informatique, de la linguistique, ..etc, le rêve de communiquer avec des machines est devenu de plus en plus réalisable. Ainsi, chercheurs et industriels se sont intéressés au développement des applications utilisant la parole comme un moyen de communication homme-machine. Dans les dix dernières années, deux applications du domaine du traitement du signal de parole ont connu une progression considérable : La Reconnaissance Automatique de la Parole (RAP) qui consiste à reconnaître le message prononcé et la Reconnaissance Automatique du Locuteur (RAL) qui consiste à reconnaître l'identité du locuteur derrière le signal de parole présenté.

Ce travail de thèse représente une des contributions au domaine de la RAL et plus particulièrement de la Vérification Automatique du Locuteur (VAL), une des tâches majeures de la RAL.
Un système de Vérification Automatique du Locuteur (VAL) est un système qui permet de décider à partir d'un signal de parole, appelé segment de test, et une identité proclamée si le signal provient de l'identité proclamée ou non. Il est composé de trois modules principaux :

- La paramétrisation qui consiste à transformer le signal de parole en une suite de vecteurs appelés trames.
- La modélisation permet de construire un modèle pour chaque locuteur utilisant les caractéristiques des paramètres extraits en paramétrisation.
- La décision qui consiste à utiliser des mesures de similarités entre le modèle de l'identité proclamée et le segment de test, afin de décider si ce signal provient de l'identité proclamée ou non.

On peut distinguer deux modes de VAL : le mode dépendent du texte où le locuteur doit prononcer un texte bien défini et le mode indépendant du texte où il n'y a aucune contrainte sur le texte prononcé par le locuteur.

Motivations

Depuis les premiers travaux en VAL, de nombreux approches du Reconnaissance de Formes (RdF) on été appliquées (cf. chapitre 3)

- Approche vectorielle : elle Consiste à représenter un locuteur par un ensemble de vecteurs représentatifs de l'espace acoustique construit à partir des trames obtenues en phase d'apprentissage (ex : Dynamic Time Warping (DTW), Vector Quantization (VQ)).
- Approche Connexionniste : Dans cette approche un locuteur est représenté par un réseau de neurones ou plusieurs réseaux de neurones permettant de le discriminer par rapport à un autre ensemble de locuteurs (ex : Multi-Layer Perceptrons (MLP) , Learning Vector Quantization (LVQ), Radial Basis Function (RBF)).
- Approche statistique : elle consiste à représenter un locuteur par un modèle statistique à long terme. Les distributions multi-gaussiennes sont les plus utilisées (ex : Hiden Markov Models (HMM) appliqués principalement en VAL en mode dépendant du texte, Gaussian Mixture Models (GMM) appliqués en mode indépendant du texte)

Dance cette grande gamme d'approches utilisées, seules l'approche statistique a pu prouver son efficacité. Ces excellentes performances leur a permis de devenir l'état de l'art en VAL. Ainsi la plupart des systèmes présentés dans la littérature sont basés sur les techniques HMM (en mode dépendant du texte) et GMM (en mode indépendant du texte) en modélisation accompagnés par la technique bayésienne connue sous le nom du Log du rapport du vraisemblance (LLR : Log Likelihood Ratio) en phase de décision.

Malgré les excellentes performances de ces systèmes, il reste beaucoup de travail et de développement à faire surtout en module de décision où la technique LLR domine dans la plupart des systèmes présentés. Cette technique est sans doute très simple mais en même temps très limitée dans son utilisation (puisqu'on ne peut pas aller plus loin dans le développement de la technique elle même). C'est pourquoi dans les cinq dernières années, la plupart des travaux de recherche en

VAL se sont focalisés sur les techniques de normalisation de score pour améliorer les performances des systèmes de VAL.

Dans le cadre de ce travail de thèse, nous avons essayé d'introduire dans le domaine de VAL les nouvelles techniques d'apprentissage statistique appelées Support Vector Machines (SVM). Ces techniques, proposées par V. Vapnik en 1995 dans [91], permettent d'aborder des problèmes très divers comme le classement[1], la régression, la fusion, etc. Nous nous intéressons principalement à ces techniques pour des tâches de classement et fusion.

Le choix de l'utilisation des SVM est soutenu par leur souplesse et surtout par une grande gamme de fonctions que ces techniques proposent afin d'approcher au mieux la fonction de classement réelle. Cela m'empêche pas que ces techniques présentent d'autres avantages :

- Ce sont des techniques discriminantes qui permettent de construire des surfaces de décision non-linéaires alors que la plupart des autres techniques utilisées dans le domaine de la RAL sont limitées à des solutions linéaires, comme la technique LLR qui est la plus utilisée et la plus performante actuellement.
- Ce sont des techniques adaptatives. Elles permettent aux systèmes d'évoluer en fonction des spécificités de la tâche qu'ils doivent réaliser, principalement dans le cas d'un apprentissage incrémental dans les applications réelles de la reconnaissance du locuteur.
- Elles sont basées sur des techniques d'estimation statistique à partir d'exemples de classes connues (techniques de classement supervisé) unifiant deux théories : *Minimisation du risque empirique* et *Capacité d'apprentissage d'une famille de fonctions*. C'est la **Minimisation du Risque Structurel** (cf. chapitre 2).
- Des bons résultats sont obtenus par ces techniques en traitement d'image [32] [33][66] et en fusion des experts pour l'authentification biométrique [8] [45] qui sont deux domaines très proches de la vérification du locuteur.

[1] Un classement est une tâche de discrimination entre classes utilisant des algorithmes d'apprentissage supervisé

Support Vector Machines

L'idée principale des SVM consiste à projeter les données dans un espace de plus grande dimension appelé, *espace de caractéristiques,* afin que les données non-linéairement séparables dans l'espace d'entrée deviennent linéairement séparables dans l'espace de caractéristiques. En appliquant dans cet espace la technique de construction d'un hyperplan optimal séparant les deux classes, on obtient une fonction de classification qui dépend d'un produit scalaire des images des données de l'espace d'entrée dans l'espace des caractéristiques. Ce produit scalaire peut être exprimé, sous certaines conditions, par des fonctions définies dans l'espace d'entrée, qu'on appelle les noyaux. Ce multiple choix de noyaux rend les SVM plus intéressantes et surtout plus riches puisqu'on peut toujours chercher de nouveaux noyaux qui peuvent être mieux adaptés à la tâche qu'on veut accomplir. Les trois noyaux les plus utilisés sont : le noyau linéaire, le noyau polynomial et le noyau gaussien noté aussi RBF "Radial Basis Function". On trouvera plus de détails sur les SVM dans le chapitre 1 et le chapitre 2 de ce document ainsi que dans les références [17][91].

Application des SVM à la vérification du locuteur

La première tentative d'utilisation des SVM en reconnaissance du locuteur a été réalisée par Schmidt en 1996 [82][83]. Dans cette application, Schmidt a utilisé les trames obtenues en phase de paramétrisation comme vecteurs d'entrée pour les SVM. Les résultats obtenus sont encourageants mais pas suffisamment performants. Les trames contiennent beaucoup d'informations concernant la parole, l'environnement, les émotions, etc, ce qui rend la tâche très difficile aux SVM pour en extraire les informations pertinentes concernant le locuteur. On en déduit donc que l'utilisation directe des trames n'est pas la bonne solution pour utiliser les SVM. Après cette première tentative d'autres laboratoires se sont intéressés à ces techniques comme IBM [30].
Sachant que les techniques SVM exigent des vecteurs d'entrée de taille fixe et que l'utilisation directe des trames n'est pas très efficace d'après les travaux de M. Schmidt [82][83], l'utilisation des techniques SVM ne devient possible que si on passe par de nouvelles formes de représentation des données.

Dans ce travail de thèse, nous avons étudié les techniques SVM pour la vérification du locuteur en mode dépendant et indépendant du texte comme des

techniques de classement ainsi que pour la fusion des données comme techniques de fusion. En vérification du locuteur en mode indépendant du texte, nous avons proposé trois nouvelles formes de représentation de données permettant de réunir l'efficacité des GMM en modélisation et la performance des SVM en décision. Pour expérimenter les SVM avec ces nouvelles formes de représentation de données, nous avons utilisé la base de données NIST'1999 et NIST'2001 qui sont des sous ensembles de la base SWITCHBOARD.
Dans le cadre du projet PICASSO, nous avons mené des expériences d'adaptation des SVM pour la vérification du locuteur en mode dépendant du texte sous forme d'application utilisant des mots de passe publics. Dans ces expériences, nous avons proposé une modélisation qui permet de construire des vecteurs d'entrée pour les SVM de taille fixe en se basant sur la transcription phonétique des mots de passe.
Nous avons testé également l'efficacité des SVM comme techniques de fusion. Cette étude concerne deux types de fusion. La première est une fusion de méthodes qui consiste à fusionner les scores obtenus par les différents participants à la tâche "One Speaker detection" aux évaluations NIST'2001 et la deuxième est une fusion de modes qui consiste à fusionner les scores obtenus suite au traitement de 4 modalités (image de face, image profil contour, image profil luminance et la parole) de la base de données M2VTS [70][71].

Organisation du document

Ce document est organisé comme suit : dans le premier chapitre, nous présentons une description de la théorie de Vapnik suivie d'une description des techniques SVM dans le chapitre 2. Le chapitre 3 est consacré à l'état de l'art du domaine de la Reconnaissance Automatique du Locuteur. Dans le chapitre 4, nous présentons la description de notre approche concernant l'adaptation des SVM pour la vérification du locuteur en mode dépendant du texte. Le chapitre 5 présente une description des approches que nous avons proposées pour adapter les SVM à la vérification du locuteur en mode indépendant du texte suivi du chapitre 6 sur l'étude des techniques SVM pour la fusion de données avant de finir par les conclusions et les perspectives.

Chapitre 1

Théorie d'apprentissage de Vapnik

L'objectif de l'apprentissage statistique à partir d'exemples étiquetés appelé aussi apprentissage supervisé est de construire une fonction qui permet d'approcher au mieux une fonction inconnue qui génère des données aléatoires, indépendantes et identiquement distribuées (iid), et dont nous ne disposons que de quelques exemples.
Un système d'apprentissage statistique à partir d'exemples est composé de trois modules principaux [91][92] :

- Un générateur qui génère des données aléatoires appelés les vecteurs d'entrée. Ces vecteurs sont indépendants et identiquement distribués suivant une distribution de probabilité inconnue $\mathrm{P}(x)$.
- Un superviseur qui associe pour chaque vecteur d'entrée x une sortie y (la classe) suivant une distribution de probabilité également inconnue $\mathrm{P}(x,y)$.
- Une machine d'apprentissage qui permet d'implémenter une famille de fonctions $f_\alpha(x)\,;\ \alpha \in \Lambda$ où Λ est un ensemble de paramètres. Ces fonctions doivent produire pour chaque vecteur d'entée x une sortie $\hat{y}$ la plus proche possible de la sortie y du superviseur.

La figure 1.1 empruntée à V. Vapnik [91] représente ces trois modules. Pour construire un tel système, deux approches sont possibles. La première consiste à construire un modèle en utilisant les données d'apprentissage, c'est l'*inférence inductive*. Le modèle construit sera utilisé pour estimer la sortie y' de chaque donnée de test x', c'est l'*inférence déductive*. La deuxième approche consiste à estimer la sortie y' de chaque donnée de test x' en utilisant directement les données d'apprentissage sans passer par un modèle, c'est l'*inférence transductive*.
La figure 1.2 empruntée à R. Fernandez [29] résume ces deux approches.

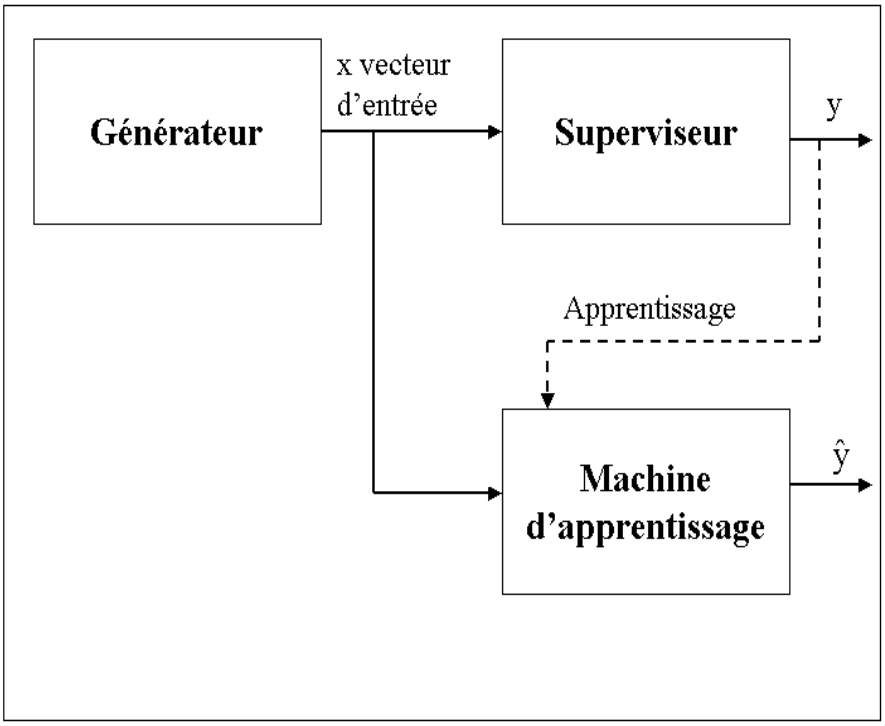


FIG. 1.1 – *Les modules d'un système d'apprentissage*

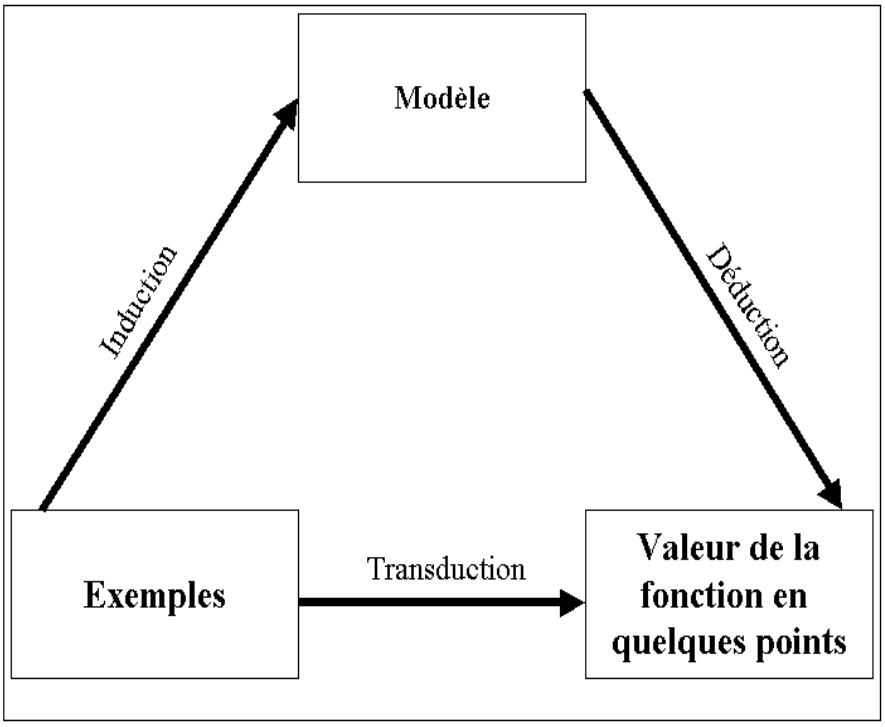


FIG. 1.2 – *Les trois modules d'inférences*

Le problème de l'apprentissage statistique à partir d'exemples apparaît dans plusieurs domaines divers et variés par exemple : la prédiction des termes des séries temporelles, régression, reconnaissance de formes, la fusion, etc.
Dans le cadre de ce travail de thèse nous nous intéressons principalement au problème d'apprentissage pour la Reconnaissance de Formes (RdF).

1.1 Apprentissage statistique supervisé pour la reconnaissance de formes

Le problème type auquel nous sommes confrontés se présente souvent sous la forme suivante : étant donné l'ensemble de mesures suivant :

$$D = \{\ (x_i,\, y_i) \in \Re^n \times \{-1, 1\} \text{ pour i} = 1, .., \text{m}\ \}$$

ces mesures sont tirées suivant une distribution de probabilité inconnue $\mathrm{P}(x,\, y)$.
Soit F une famille de fonctions tel que :

$$F = \{\ f_\alpha(x)\ /\ \alpha \in \Lambda\ \}$$

L'objectif est de trouver $f_{\alpha^*} \in F$ telle que l'estimation $f_{\alpha^*}(x) = \hat{y}$ soit la meilleure possible. Ainsi, le choix de cette fonction dépend des m données d'apprentissage i.i.d suivant la distribution de probabilité $\mathrm{P}(x,\ y) = \mathrm{P}(x)\mathrm{P}(y/x)$. Afin de bien sélectionner la fonction f_{α^*}, nous aurons besoin d'une fonction de coût $L(f_\alpha(x), y)$. La forme de cette fonction dépend principalement de la tâche à accomplir. Pour la reconnaissance de forme, la fonction du coût prend la forme suivante :

$$L(f_\alpha(x), y) = \begin{cases} 0 & si \quad y = f_\alpha(x) \\ 1 & si \quad y \neq f_\alpha(x) \end{cases}$$

Ainsi la fonction f_{α^*} correspondra à celle qui minimise ce qu'on appelle *le Risque fonctionnel* $R(\alpha)$ définie par :

$$R(\alpha) = \int L(f_\alpha(x), y) dP(x, y)$$

Comme on peut le constater, le calcul du risque fonctionnel dépend de la famille des fonctions F et de la distribution de probabilité $\mathrm{P}(x,\ y)$. Par suite, le calcul direct de la quantité $R(\alpha)$ est impossible. On doit donc l'estimer en utilisant l'ensemble des données D par ce qu'on appelle *le risque empirique*.

1.2 Minimisation du Risque Empirique (ERM)

En reconnaissance des formes, le risque empirique est défini comme suit :

$$R_{emp}(\alpha) = \frac{1}{m} \sum_{i=1}^{m} L(f_\alpha(x), y)$$

Par conséquence, pour bien sélectionner la fonction f_{α^*}, il faut trouver l'ensemble des paramètres $\alpha^* \in \Lambda$ telle que $R_{emp}(\alpha)$ soit minimale. C'est le principe de l'approche du *Minimisation du Risque Empirique (ERM)*. Cette approche consiste donc à minimiser R_{emp} sur Λ en s'appuyant sur le fait que $R_{emp}(\alpha)$ converge vers $R(\alpha)$ lorsque la taille des données tend vers l'infini (la loi des grands nombres). Cela définit la consistance de l'approche.

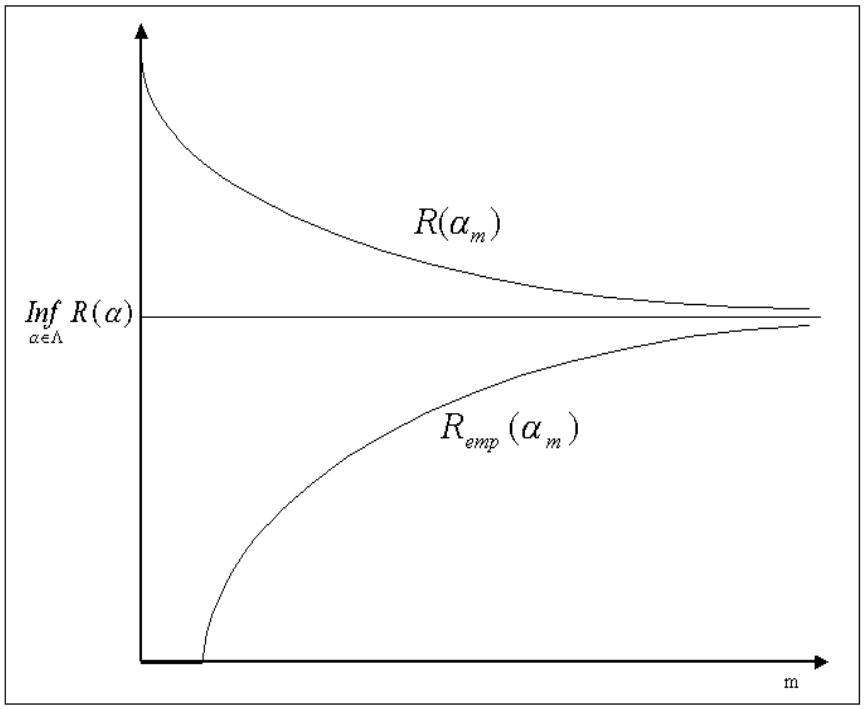


FIG. 1.3 – *Consistance de l'approche ERM*

1.2.1 Consistance de l'approche ERM

Définition 1 :

L'approche ERM est consistante pour F, L et P si :

$$R(\alpha_m) \longrightarrow \inf_{\alpha\in\Lambda} R(\alpha) \tag{1.1}$$

$$R_{emp}(\alpha_m) \longrightarrow \inf_{\alpha\in\Lambda} R(\alpha) \tag{1.2}$$

C'est à dire que la suite du risque fonctionnel et celle des risques empiriques convergent en probabilité vers la même limite qui est la plus petite valeur du risque fonctionnel $R(\alpha)$. La figure 1.3 empruntée à V.Vapnik [91] représente *une interprétation visuelle* de cette propriété : suivant Vapnik, la consistance définie pas les équations (1.1) et (1.2) est une consistance triviale. Pour s'en convaincre, il suffit de considérer l'existence d'une fonction ϕ tel que :

$$\forall(x,y) \in D \qquad \inf_{\alpha\in\Lambda} L(f_\alpha(x), y) > \phi(x,y)$$

Si on considère la famille de fonctions $F' = \{\{f_\alpha \in F\} \cup \phi\}$. L'approche ERM satisfait trivialement les conditions des équations (1.1) et (1.2) sur l'ensemble F'. une définition plus générale de la consistance de l'approche ERM a été proposée par Vapnik qu'il a nommé *Consistance non triviale*.

Définition 2 :

$$Soit \qquad \Lambda(c) = \{\alpha \in \Lambda, \int L(f_\alpha(x), y) dP(x, y) > c, \quad c \in \Re\}$$

L'approche ERM est non trivialement consistante si on a la convergence en probabilité suivante pour tout Λ(c) non vide, c $\in \Re$:

$$\inf_{\alpha \in \Lambda(c)} R_{emp}(\alpha) \longrightarrow \inf_{\alpha \in \Lambda(c)} R(\alpha)$$

On peut constater que cela met fin à la consistance triviale. Pour cela, il suffit de prendre $c^* > \int \phi(x, y) dP(x, y)$. Ainsi la fonction ϕ ne se trouvera pas dans l'ensemble $\Lambda(c^*)$. Le théorème fondamental sur la consistance a été démontré en 1989 par Vapnik et Chervonenkis [89][90][91].

Théorème 1 :

Soit $\{ f_\alpha, \alpha \in \Lambda \}$ un ensemble de fonctions satisfaisant la condition :

$$a \leq \int L(f_\alpha(x), y) dP(x, y) \leq b$$

avec a, b $\in \Re$. L'approche ERM est consistante si et seulement si

$$\lim_{m \to \infty} P\{\sup_{\alpha \in \Lambda}(R(\alpha) - R_{emp}(\alpha)) > \epsilon\} = 0 \quad \forall \epsilon > 0$$

Cette convergence doit être entendue comme convergence uniforme sur l'ensemble $\{ f_\alpha, \alpha \in \Lambda \}$.

Il est intéressant de remarquer que cette formulation déduite par V. Vapnik dans le théorème précédent montre que la consistance est déterminée par le pire des cas.

Bien que $R_{emp}(\alpha)$ est un estimé non-biaisé de $R(\alpha)$. L'ensemble des paramètres α^* qui optimise $R_{emp}(\alpha)$ n'est pas le même qui optimise $R(\alpha)$. Ainsi, en minimisant le risque empirique sur l'ensemble d'apprentissage, nous obtenons un modèle qui est efficace sur cet ensemble, mais dont nous n'avons a priori aucune garantie de performance sur de nouveaux exemples. Ce problème est bien connu sous le nom de *sur-apprentissage* (*overfitting*)[14]. La figure 1.4 représente une interprétation du phénomène du sur-apprentissage. Pour résoudre ce problème,

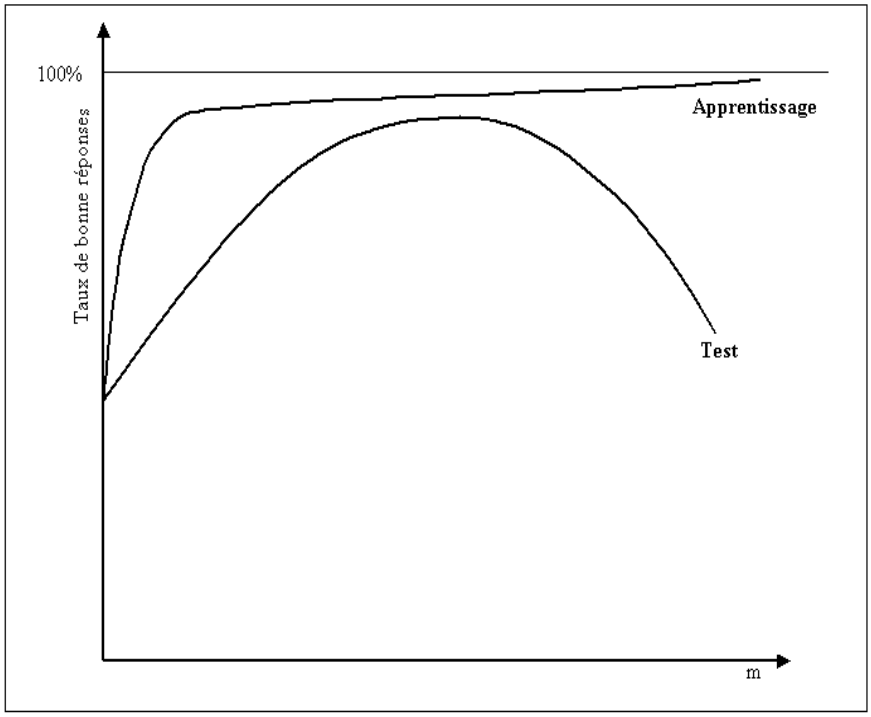


FIG. 1.4 – *L'effet du phénomène de sur-apprentissage*

Vapnick et Chervonenkis ont mit en place toute une approche qu'ils ont nommée *Minimisation du risque Structurel* (SRM).

1.3 Minimisation du Risque Structurel

Vers la fin des années soixante, Vapnik et Chervonenkis ont introduit un nouveau concept nommé *VC-dimension* qui présente une sorte de mesure de capacité de calcul d'une famille de fonctions F. Ce concept a été proposé dans un premier temps pour les fonctions indicatrices avant d'être étendue au fonctions à valeurs réelles.

Soit un ensemble de fonctions indicatrices F de l'ensemble D. On dit que F sépare $A \subseteq D$ si $\forall\ B \subseteq A\ \exists\ f_B \in F$ tel que l'on ait : $f_B(A/B) = 0$ et $f_B(B) = 1$. Ainsi la dimension de Vapnik peut être définie comme le cardinal maximal des sous-ensembles A séparables par F.

Définition 3 :

La dimension de Vapnik-Chervonenkis où VC-dimension d'une famille de fonctions F est le nombre maximum d'exemples d'un ensemble de données D qui peuvent être séparables par la famille de fonctions F (pour plus de détails consulter [44][90][91][92]).

Notons que la dimension d'une famille peut être infinie.

Après avoir donné la définition de VC-dimension, nous pouvons énoncer le théorème suivant :

Théorème 2 :

Pour que le principe de ERM soit consistant, indépendamment de la distribution de probabilité des exemples, il suffit que la famille de fonctions que le système d'apprentissage est capable d'implémenter ait une VC-dimension **finie** .

Ce théorème donne seulement **une condition suffisante** concernent la consistance de l'approche ERM.

La théorie développée par Vapnik et Chervonenkis fournie également des bornes sur la vitesse de convergence du processus d'apprentissage. Ainsi, ils ont démontré que $\forall\ \alpha \in \Lambda$ et pour $m > h$ où h est la VC-dimension de la famille F et m est la taille de l'ensemble d'exemples, l'inégalité suivante sera vérifiée avec une probabilité au moins égale à 1-η :

$$R(\alpha) \leq R_{emp}(\alpha) + \sqrt{\frac{h[Log(\frac{2m}{h}) + 1] - Log(\frac{\eta}{4})}{m}} \tag{1.3}$$

Le membre droit de l'inégalité (1.3) appelé **le Risque garanti** est composé de deux termes : le risque empirique et une quantité qui dépend du rapport $\frac{m}{h}$ appelée **Intervalle de confiance** puisqu'il représente la différence entre le risque empirique $R_{emp}(\alpha)$ et le risque fonctionnel $R(\alpha)$

Si le rapport $\frac{m}{h}$ est suffisamment grand, l'approche du minimisation du risque empirique suffit pour garantir une faible valeur du risque fonctionnel $R(\alpha)$. Par contre, lorsque le rapport $\frac{m}{h}$ n'est pas suffisamment grand l'intervalle de confiance prend une valeur importante. Ainsi l'approche du minimisation empirique n'est pas suffisante pour garantir une valeur minimale du risque fonctionnel $R(\alpha)$. Afin de remédier à ce problème, Vapnik et Chervonenkis proposent une nouvelle approche de Minimisation du Risque Structurel [91][92]. Le principe de cette approche est basé sur la minimisation du risque garanti (membre droit de l'inégalité 1.3). c'est à dire la minimisation conjointe des deux causes d'erreur, le risque empirique et l'intervalle de confiance.

Définition 4 :

Soit $F = \{ f_\alpha / \alpha \in \Lambda \}$ une famille de fonctions. On définit une structure sur F comme étant une suite de sous-ensembles emboîtés $F_i = \{ f_\alpha^i / \alpha \in \Lambda_i, \Lambda_i \subset \Lambda \}$ On a donc :

$$F_1 \subset F_2 \subset ... \subset F_i \subset ...$$

D'aprés la définition de la VC-dimension, on déduit que :

$$h_1 \leq h_2 \leq ... \leq h_i \leq ...$$

En pratique, lorsque la VC-dimension augmente le risque empirique décroît tandis que l'intervalle de confiance croît. la figure 1.5 présente le comportement du risque empirique, l'intervalle de confiance et le risque garanti en fonction de la VC-dimension. Le risque garanti atteint son minimum pour une valeur optimale de la VC-dimension. Ainsi, l'objectif de l'approche SRM est de trouver cette valeur optimale qui garanti une faible valeur du risque fonctionnel $R(\alpha)$. Cela revient à chercher un compromis entre la qualité de l'approximation sur l'échantillon et la complexité de la fonction qui réalise l'approximation.

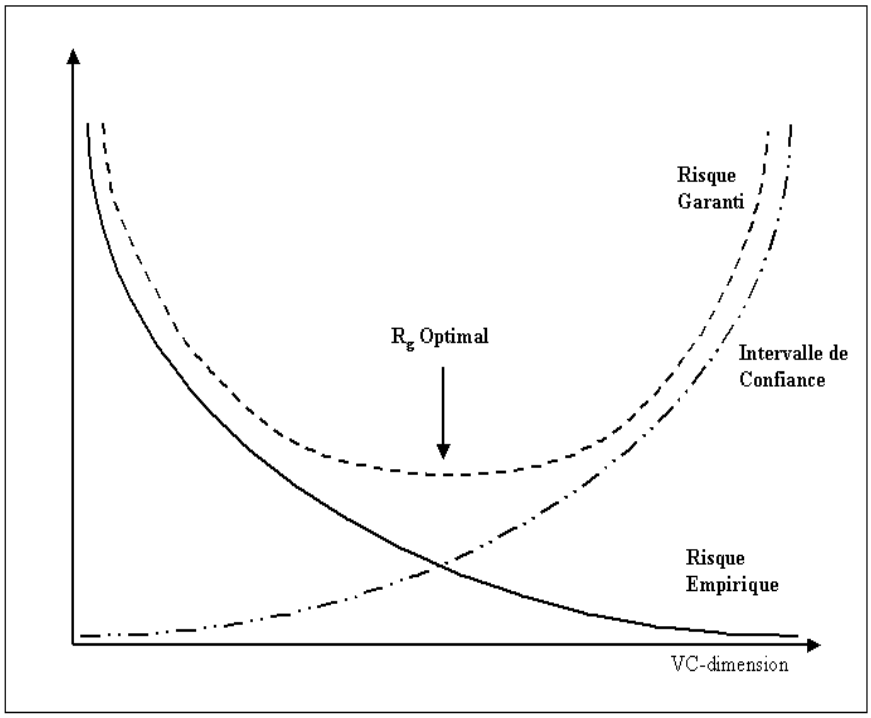


FIG. 1.5 – *Comportement du risque empirique, l'intervalle de confiance et le risque garanti en fonction de la VC-dimension*

Chapitre 2

Machines à Vecteurs Supports

Les Support Vector Machines (SVM) sont des nouvelles techniques discriminantes dans la théorie de l'apprentissage statistique. Elles ont été proposées en 1995 par V. Vapnik dans son livre "The nature of statistical learning theory" [91]. Elles permettent d'aborder plusieurs problèmes divers et variés comme la régression, la classification, la fusion etc.
Les SVM fournissent une approche très intéressante de l'approximation statistique. Souvent, le nombre des exemples pour l'apprentissage est insuffisant pour que les estimateurs fournissent un modèle avec une bonne précision. D'un autre côté, l'acquisition d'un grand nombre d'exemples s'avère être souvent très coûteuse et peut même mener à des problèmes de sur-apprentissage dans le cas ou la capacité du modèle est très complexe. Pour ces deux raisons, il faut arriver à un compromis entre la taille des échantillons et la précision recherchée. Dans ces cas spécifiques comme la reconnaissance de formes, il serait intéressant de trouver une mesure de la fiabilité de l'apprentissage, et d'avoir une mesure du taux d'erreur qui sera commis durant la phase de test. Ces nouvelles techniques unifient deux théories : *Minimisation du risque empirique* et *Capacité d'apprentissage d'une famille de fonctions*. C'est la **Minimisation du Risque Structurel** (cf. chapitre1).
Le principe des SVM consiste à projeter les données de l'espace d'entrée (appartenant à deux classes différentes) non-linéairement séparables dans un espace de plus grande dimension appelé *espace de caractéristiques* de façon à ce que les données deviennent linéairement séparables. Dans cet espace, on construit un hyperplan optimal séparant les classes tel que :

– Les vecteurs appartenant aux différentes classes se trouvent de différents

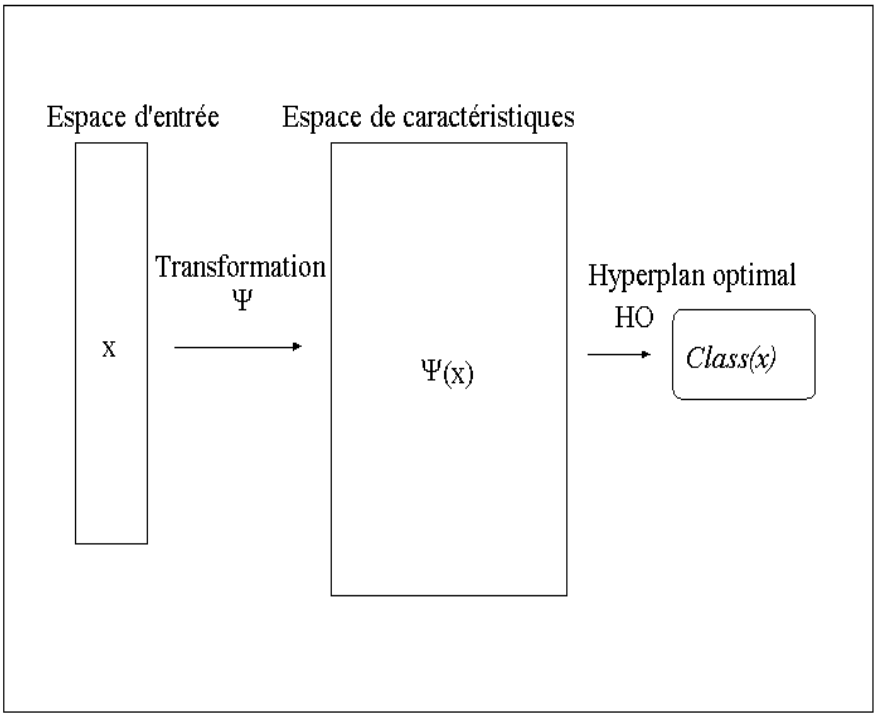


FIG. 2.1 – *Principe des techniques SVM*

côtés de l'hyperplan.

- La plus petite distance entre les vecteurs et l'hyperplan (*la marge*) soit maximale.

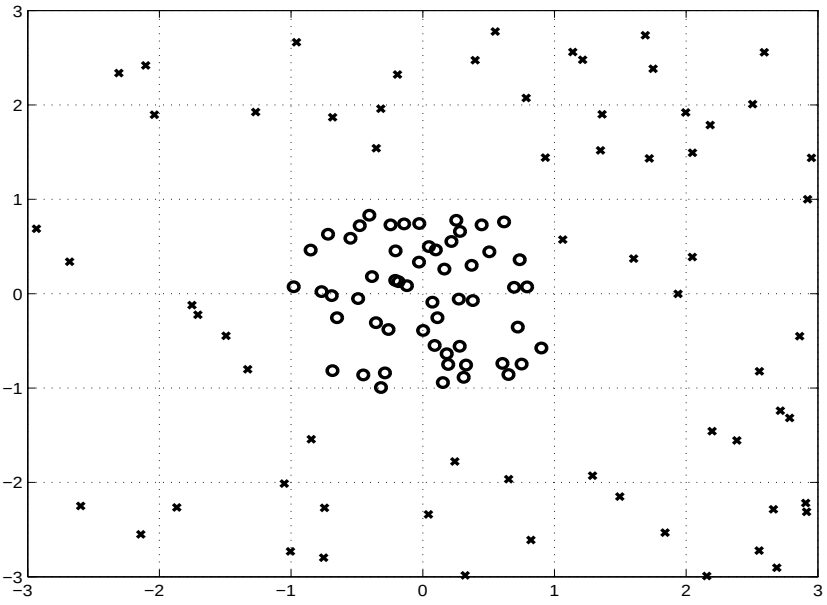
La figure 2.1 représente le principe de la technique SVM.

Exemple :

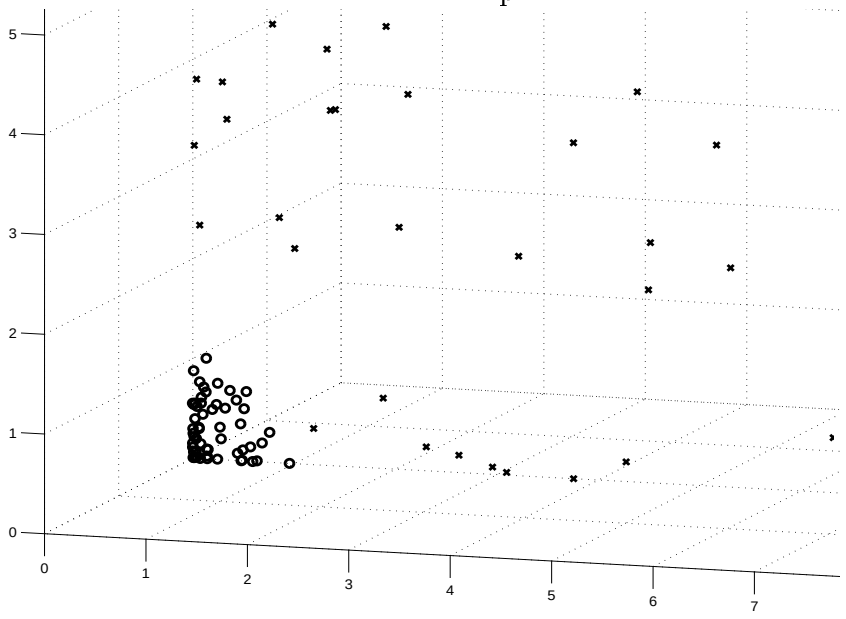
Pour avoir une idée plus claire sur les SVM, voici un exemple inspiré du travail de B. Schôlkof [84] qui met en pratique le principe des SVM. Dans cet exemple, les données non-linéairement séparables dans $\Re^2$ deviennent linéairement séparables dans $\Re^3$ grâce à la transformation ψ définie par :

$$\begin{aligned} \psi \quad : \quad & \Re^2 \longrightarrow \Re^3 \\ & (x_1, x_2) \longrightarrow (x_1^2, \sqrt{2}x_1x_2, x_2^2) \end{aligned}$$

La figure 2.2 représente une simulation de cette transformation que nous avons réalisée avec Matlab. Les données de la figure 2.2.a ont été tirées aléatoirement dans $\Re^2$ et la figure 2.2.b représente l'image de ces données dans $\Re^3$ suivant la transformation ψ.



a : exemples tirés aléatoirement dans $\Re^2$ appartenant à deux classes non-linéairement séparables



b : l'image des exemples de la figure 2.2 (a) dans $\Re^3$ en utilisant la transformation ψ

FIG. 2.2 – *Exemple montrant l'efficacité d'une transformation dans un espace de plus grande dimension pour faciliter le classement*

2.1 Construction de l'hyperplan optimal

Pour bien décrire la technique de construction de l'hyperplan optimal séparant des données appartenant à deux classes différentes dans deux cas différents : Le cas des données linéairement séparables et le cas des données non-linéairement séparables. Nous considérons le formalisme que nous avons proposé dans le chapitre précédent. Soit l'ensemble D tel que :

$$D = \{ \ (x_i, \ y_i) \in \Re^n \times \{-1, 1\} \text{ pour i} = 1, .., \text{ m} \ \}$$

2.1.1 Cas des données linéairement séparables

Dans ce paragraphe nous présentons la méthode générale de construction de l'Hyperplan Optimal (HO) qui sépare des données appartenant à deux classes différentes linéairement séparables. La figure 2.3 donne une représentation visuelle de l'HO dans le cas des données linéairement séparables.

Soit $H : (w \cdot x) + b$ l'hyperplan qui satisfait les conditions suivantes :

$$\begin{cases} w \cdot x_i + b \geq 1 & si \quad y_i = 1 \\ w \cdot x_i + b \leq -1 & si \quad y_i = -1 \end{cases}$$

ce qui est équivalent à :

$$y_i(w \cdot x_i + b) \geq 1 \quad pour \quad i = 1, .., m \tag{2.1}$$

Comme nous l'avons déjà mentionné, un HO est un hyperplan qui maximise la marge M qui représente la plus petite distance entre les différentes données des deux classes et l'hyperplan. Maximiser la marge M est équivalent à maximiser la somme des distances des deux classes par rapport à l'hyperplan. Ainsi, la marge a l'expression mathématique suivante :

$$\begin{aligned} M &= \min_{x_i | y_i = 1} \frac{w \cdot x + b}{||w||} - \max_{x_i | y_i = -1} \frac{w \cdot x + b}{||w||} \\ &= \frac{1}{||w||} - \frac{-1}{||w||} \\ &= \frac{2}{||w||} \end{aligned}$$

Trouver l'hyperplan optimal revient donc à maximiser $\frac{2}{||w||}$. Ce qui est équivalent à minimiser $\frac{||w||^2}{2}$ sous la contrainte 2.1. Ceci est un problème de minimisation

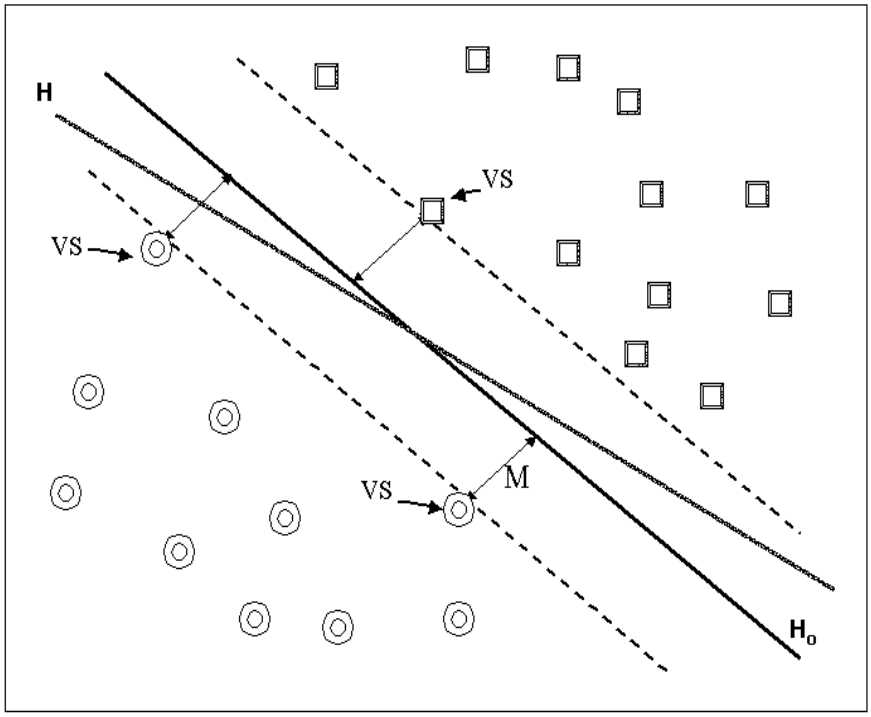


FIG. 2.3 – *Hyperplans séparateurs : H est un hyperplan quelconque, H_o est l'hyperplan optimal et M est la marge qui représente la distance entre les différentes classes et H_o (VS sont les Vecteurs Supports).*

d'une fonction objective quadratique avec contraintes linéaires.

Principe de Fermat (1638)

Les points qui minimisent où maximisent une fonction dérivable annule sa dérivée. Ils sont appelés *points stationnaires*.

Principe de Lagrange (1788)

Pour résoudre un problème d'optimisation sous contrainte, il suffit de rechercher un point stationnaire z_o du lagrangien $L(z, \alpha)$ de la fonction g à optimiser et les fonctions C_i^g exprimant les contraintes

$$L(z, \alpha) = g(z) + \sum_{i=1}^{m} \alpha_i C_i^g(z)$$

où les $\alpha_i = (\alpha_1, \ldots, \alpha_m)$ sont des constantes appelés *coefficients de Lagrange*

Principe de Kuhn-Tucker (1951)

Avec des fonctions g et C_i^g convexe, il est toujours possible de trouver un *point-selle* (z_o, α^*) qui vérifie

$$\min_z L(z, \alpha^*) = L(z_o, \alpha^*) = \max_{\alpha \geq 0} L(z_o, \alpha)$$

Pour plus de détails sur ces trois principes, le lecteur peut consulter les références [23][31][62].

En appliquant le principe de Kuhn-Tucker, on est amené à rechercher un point-selle (w_o, b_o, α^o). Le lagrangien correspondant à notre problème est :

$$L(w, b, \alpha) = \frac{1}{2} w \cdot w - \sum_{i=1}^{m} \alpha_i [y_i [(x_i \cdot w) + b] - 1] \tag{2.2}$$

Le lagrangien doit être minimal par rapport à w et b et maximal par rapport à $\alpha \geq 0$.

- $L(w,b,\alpha)$ est minimal par rapport à b :

$$\frac{\partial L(w, b, \alpha)}{\partial b} = 0 \quad \Longleftrightarrow \quad \sum_{i=1}^{m} \alpha_i y_i = 0 \tag{2.3}$$

- $L(w,b,\alpha)$ est minimal par rapport à w :

$$\frac{\partial L(w, b, \alpha)}{\partial w} = 0 \quad \Longleftrightarrow \quad w - \sum_{i=1}^{m} \alpha_i x_i y_i = 0 \tag{2.4}$$

– $L(w,b,\alpha)$ est maximal par rapport à $\alpha \geq 0$:
En remplaçant (2.3) et (2.4) dans le lagrangien (2.2) on aura

$$L(w,b,\alpha) = \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{m} \alpha_i \alpha_j y_i y_j (x_i \cdot x_j) \tag{2.5}$$

Ainsi notre problème est de maximiser $L(w,b,\alpha)$ sous la contrainte :

$$\sum_{i=1}^{m} \alpha_i y_i = 0; \qquad \alpha_i \geq 0$$

Soit la solution $\alpha^o = (\alpha_1^o, \ldots, \alpha_m^o)$. D'après le théorème de Kuhn-Tucker une condition nécessaire et suffisante pour que α^o soit optimal est :

$$\alpha_i^o [y_i[(w_o \cdot x_o) + b_o] - 1] = 0 \qquad pour \quad i = 1, .., m$$

ce qui veut dire que : $\alpha_i^o = 0$ où $y_i[(w_o \cdot x_i) + b_o\ ] = 1$

Définition

On définit **les Vecteurs Supports VS** tout vecteur x_i tel que $y_i[(w_o \cdot x_i) + b_o\ ] = 1$. Ce qui est équivalent à :

$$VS = \{x_i \quad | \quad \alpha_i > 0\} \quad pour \quad i = 1, .., m$$

Ainsi, on peut facilement calculer w_o et b_o :

$$w_o = \sum_{VS} \alpha_i^o y_i x_i$$

$$b_o = -\frac{1}{2}[(w_o \cdot x^*(1))] + [(w_o \cdot x^*(-1))]$$

la fonction de classement $class(x)$ est définit par :

$$class(x) = sign[(w_o \cdot x) + b_o] \tag{2.6}$$

$$= sign[\sum_{x_i \in VS} \alpha_i^o y_i (x_i \cdot x) + b_o] \tag{2.7}$$

Si $class(x)$ est inférieure à 0, x est de la classe -1 sinon il est de la classe 1.

2.1.2 Cas des données non-linéairement séparables

Dans ce cas où les données sont non-linéairement séparables figure 2.4, l'hyperplan optimal est celui qui satisfait les conditions suivantes :

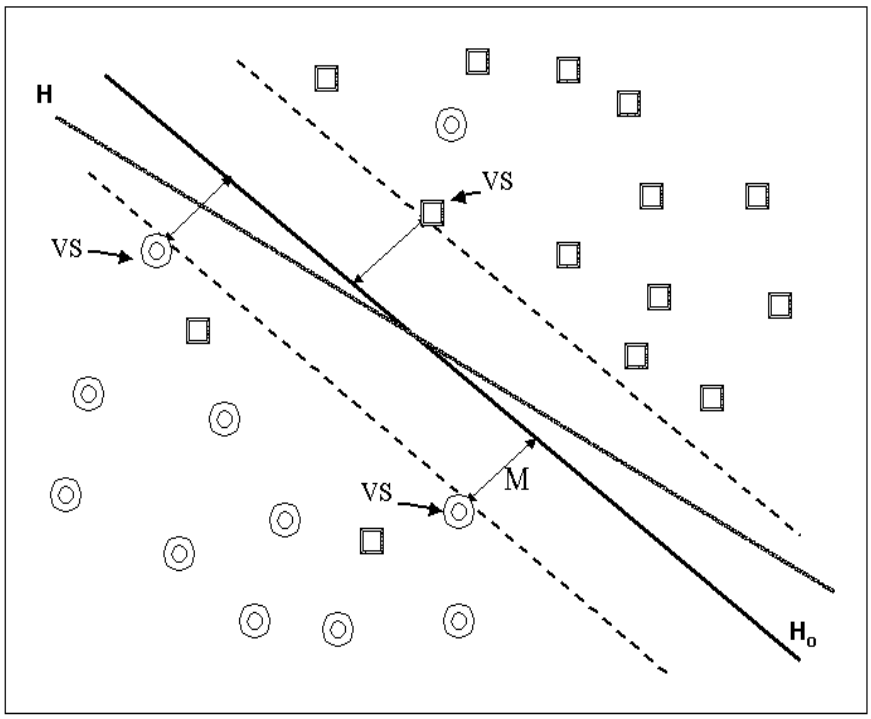


FIG. 2.4 – *Hyperplans séparateurs dans le cas de données non-linéairement séparables(VS sont les Vecteurs Supports).*

- La distance entre les vecteurs bien classés et l'hyperplan optimal doit être maximale.
- la distance entre les vecteurs mal classés et l'hyperplan optimal doit être minimale.

Pour formaliser tout cela, on introduit des variables de pénalité non-négatives ξ_i pour i = 1, .. ,m appelées variables d'écart. Ces variables transforment l'inégalité (2.1) comme suit :

$$y_i(w \cdot x_i + b) \geq 1 - \xi_i \quad pour \quad i = 1, .., m$$

L'objectif est de minimiser la fonction suivante :

$$\Psi(w, \Xi) = \frac{1}{2} w \cdot w + C \sum_{i=1}^{m} \xi_i$$

où C est un paramètre de régularisation. Elle permet de concéder moins d'importance aux erreurs. Cela mène à un problème dual légèrement différent de celui du cas des données linéairement séparables. Maximiser le lagrangien donné par l'équation (2.5) par rapport à α_i sous les contraintes suivantes :

$$\sum_{i=1}^{m} \alpha_i y_i = 0 \quad avec \quad 0 \leq \alpha_i \leq C \quad pour \quad i = 1, .., m.$$

Le calcul de la normale w_o, du biais b_o et de la fonction de classification $class(x)$ reste exactement le même que pour le cas des données linéairement séparable.

2.2 Principe des SVM

Les classifieurs SVM utilisent l'idée de l'HO (Hyperplan Optimal) pour calculer une frontière entre des nuages de points. Elles projettent les données dans l'*espace de caractéristiques* en utilisant des fonctions non-linéaires. Dans cet espace on construit l'HO qui sépare les données transformées. L'idée principale est de construire une surface de séparation linéaire dans l'espace des caractéristiques qui correspond à une surface non-linéaire dans l'espace d'entrée.
Le problème principal à relever ici est comment bien manipuler la transformation de tous les vecteurs d'entrée dans l'espace des caractéristiques de façon à éviter une augmentation du coût en nombre de paramètres libres.
Soit l'ensemble D' l'image de l'ensemble D, défini dans la section 2.1, par la transformation ψ.

$$D' = \{ \ (\psi(x_i),\, y_i) \in \Re^p \times \{-1, 1\} \text{ pour i} = 1, .., \text{m} \mid p \geq n \ \}$$

En construisant un HO dans l'espace des caractéristiques suivant la technique expliquée dans la section 2.1. On aura la fonction de classement suivante :

$$class(x) = sign[\sum_{x_i \in VS} \alpha_i^o y_i(\psi(x_i) \cdot \psi(x)) + b_o] \tag{2.8}$$

On peut remarquer que la fonction de classement dépend du produit scalaire dans l'espace des caractéristiques. Ainsi, pour que le coût de calcul reste pratiquement inchangé et le nombre de paramètres libres du système n'augmente pas, il faut que la fonction ψ satisfasse la condition suivante :

$$\psi(u) \cdot \psi(v) = K(u, v)$$

C'est à dire le produit scalaire dans l'espace des caractéristiques va être représentable comme un noyau de l'espace d'entrée. Le classifieur est donc construit sans utiliser explicitement la fonction ψ.
Suivant la théorie de Hilbert-Schmidt, une famille de fonctions qui permet cette représentation et qui sont très appropriées aux besoins des SVM peut être définie comme l'ensemble des fonctions symétriques qui satisfont la condition suivante :
Théorème (Mercer) :
Pour être sûr qu'une fonction symétrique $K(u, v)$ admet un développement de la forme suivante :

$$K(u, v) = \sum_{k=1}^{+\infty} \beta_k \psi_k(u) \cdot \psi_k(v)$$

tel que les $\beta_k > 0$ (i.e $K(u, v)$ décrit un produit interne dans l'espace des caractéristiques) il est nécessaire et suffisant que la condition suivante soit satisfaite :

$$\int \int K(u, v) g(u) g(v) du dv \geq 0$$

pour toute fonction g $\neq 0$ avec :

$$\int g^2(z) dz \geq 0$$

On appelle ces fonctions les *noyaux de Hilbert-Schmidt*. Plusieurs noyaux ont été utilisés par les chercheurs, en voici quelques uns :

- Le noyau linéaire :

$$K(u, v) = u.v$$

– Le noyau Polynomial :

$$K(u,v) = [(u.v) + 1]^d$$

où d est le degré du polynôme à déterminer par l'utilisateur.

– Le noyau RBF (Radial Basis Function) :

$$K(u,v) = exp(-\frac{||u-v||^2}{2\sigma^2})$$

où σ est à déterminer.

Maintenant que nous avons défini ce qu'est un noyau, la fonction de classement (2.8) devient :

$$class(x) = sign[\sum_{x_i \in VS} \alpha_i^o y_i K(x_i, x) + b_o] \tag{2.9}$$

Reprenons l'exemple qu'on a évoqué au début de ce chapitre. Donc on a la transformation ψ tel que :

$$\begin{aligned} \psi \quad : \quad & \Re^2 \longrightarrow \Re^3 \\ & X = (x_1, x_2) \longrightarrow (x_1^2, \sqrt{2}x_1x_2, x_2^2) \end{aligned}$$

D'où :

$$\begin{aligned} K(U,V) &= \psi(U) \cdot \psi(V) \\ &= \left(u_1^2, \sqrt{2}u_1u_2, u_2^2\right) \cdot \begin{pmatrix} v_1^2 \\ \sqrt{2}v_1v_2 \\ v_2^2 \end{pmatrix} \\ &= \left(u_1^2v_1^2 + 2u_1v_1u_2v_2 + u_2^2v_2^2\right) \\ &= (u_1v_1 + u_2v_2)^2 \\ &= \left[(u_1, u_2) \cdot \begin{pmatrix} v_1 \\ v_2 \end{pmatrix}\right]^2 = (U \cdot V)^2 \end{aligned}$$

Comme on peut le constater le noyau correspondant à la transformation ψ de notre exemple proposé au début de ce chapitre n'est autre qu'un noyau polynomial de degré 2.

Chapitre 3

Reconnaissance Automatique du Locuteur

La communication entre l'homme et la machine est actuellement un des sujets de recherche les plus intéressants. Et sans doute la parole est le moyen de communication le plus naturel et le plus rapide.
Le signal de la parole est un signal très complexe dont les caractéristiques varient au cours du temps. L'objectif de son traitement est l'extraction des informations imbriquées qu'il contient (ex : message, locuteur, environnement, etc). Au début, la recherche s'est limitée au traitement proprement dit du signal. Mais peu à peu d'autre disciplines (comme l'intelligence artificielle, l'informatique, la reconnaissance de formes, la phonétique, la linguistique, etc) sont intervenues afin de concevoir et développer des systèmes experts utilisant la parole comme moyen de communication. Les principaux objectifs du traitement du signal de parole sont :

- Un codage efficace du signal pour sa transmission ou son enregistrement.
- La reconnaissance automatique de la parole.
- La reconnaissance automatique du locuteur.
- La synthèse du signal de la parole.
- Certaines applications médicales
- Certaines applications pour l'étude des langues.

Dans le cadre de ce travail de thèse, nous nous intéressons au problème de la reconnaissance du locuteur et plus particulièrement au problème de la vérification du locuteur. La figure 3.1 donne un schéma général du domaine de la communication parlée homme-machine dans lequel se situe le contexte de notre étude. La Reconnaissance Automatique du Locuteur (RAL) est une des disciplines du

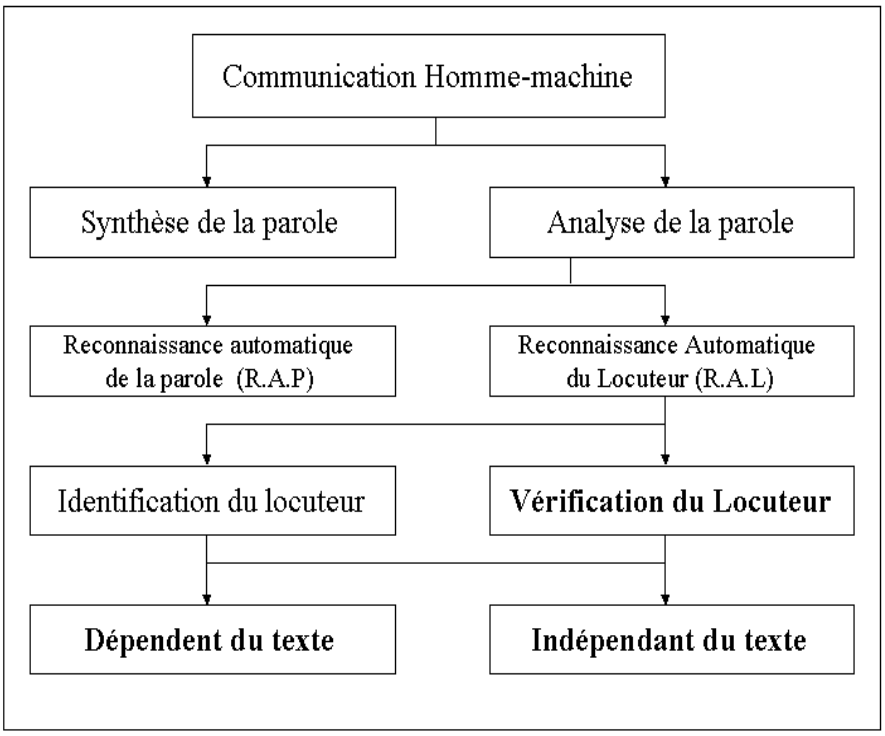


FIG. 3.1 – *Contexte de l'étude*

traitement du signal de la parole qui a pour objectif d'extraire les informations concernant le locuteur à partir de son signal vocal afin de pouvoir le reconnaître ultérieurement. On peut distinguer deux mode de RAL :

- RAL en mode dépendant du texte : dans ce genre de système, la reconnaissance du locuteur est réalisée à l'aide d'un message connu a priori par le système que le locuteur doit prononcer (mot de passe, code PIN, phrase,...) [20]. Ce message peut être choisi par le locuteur comme dans les systèmes qui utilisent des mots de passe personnalisés [51], ou imposé par le système lui même comme dans les systèmes utilisant des codes PIN [69]. Ce message peut également être imposé par le système sous forme visuelle ou auditive.
- RAL en mode indépendant du texte : dans ce genre de système, il n'existe aucune contrainte sur le message que le locuteur doit prononcer ni sur la langue qu'il peut utiliser.

3.1 Les différentes tâches en RAL

L'identification Automatique du Locuteur (IAL) et la Vérification Automatique du Locuteur (VAL) sont les deux tâches les plus répandues dans le domaine de la RAL. Récemment, pour des applications plus spécifiques, d'autre tâches ont vu le jour comme l'indexation du locuteur qui consiste à indiquer à quel moment chaque locuteur intervenant dans une conversation a pris la parole. Une application connexe est la détection d'un locuteur lors d'une conversation. Dans cette section, nous allons décrire principalement les deux tâches principales de la RAL : IAL et VAL.

3.1.1 Identification Automatique du locuteur (IAL)

Dans cette tâche, le système doit fournir l'ensemble des locuteurs de la base les plus proches du locuteur qui a produit le signal de parole de test. Pour cela, le système calcule des mesures de similarités entre ce signal et tous les modèles des locuteurs de la base. La figure 3.2 représente un schéma modulaire d'un système d'IAL. Dans le cas où le système doit fournir un ensemble d'au moins un locuteur, on parle d'une identification dans un ensemble fermé. Mais dans certaines applications, le système peut être amené à fournir un ensemble vide. C'est l'identification en ensemble ouvert. Dans ces applications, le système n'ajoute un locuteur dans l'ensemble des locuteurs les plus proches que si le score de test du

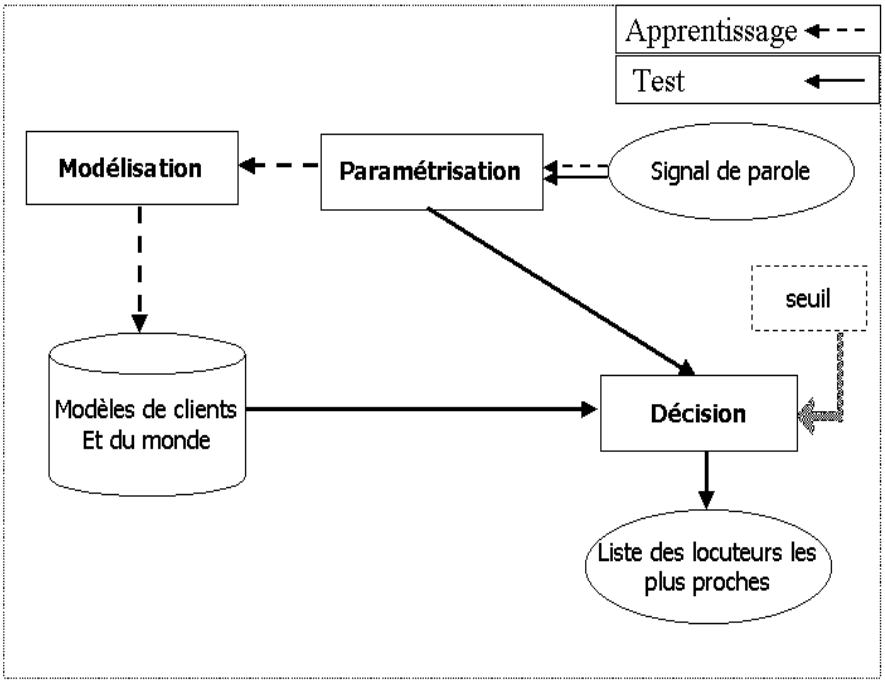


FIG. 3.2 – *Schéma modulaire d'un système d'IAL*

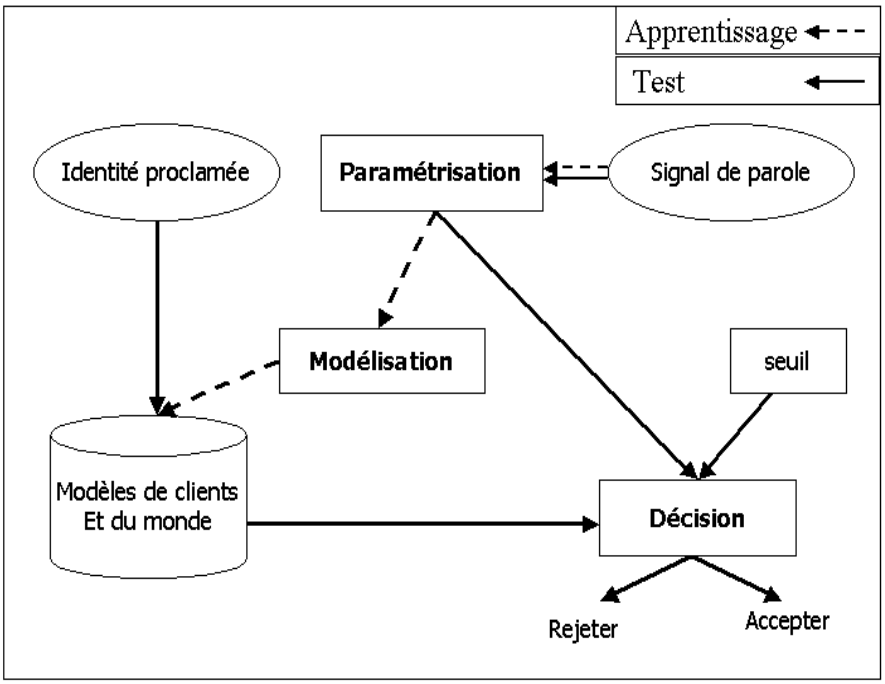


FIG. 3.3 – *Schéma modulaire d'un système d'VAL*

signal de parole présenté sur le modèle de ce locuteur est supérieur à un seuil défini à priori. En pratique, la plupart des systèmes d'IAL fournissent un ensemble d'un seul locuteur qui représente le locuteur le plus proche.

3.1.2 Vérification Automatique du locuteur (VAL)

Un système de VAL doit vérifier à partir d'un signal de parole et d'une identité proclamée qui appartient à la base de donnée si le signal présenté provient de l'identité proclamée ou non. La figure 3.3 représente un schéma modulaire d'un système de vérification du locuteur. Un système de RAL est composé de trois étapes principales : la paramétrisation, la modélisation et la décision

3.2 La paramétrisation

Cette étape permet de transformer un signal de parole en une suite de vecteurs appelés trames. Le signal de parole, de par sa complexité (multitude d'informa-

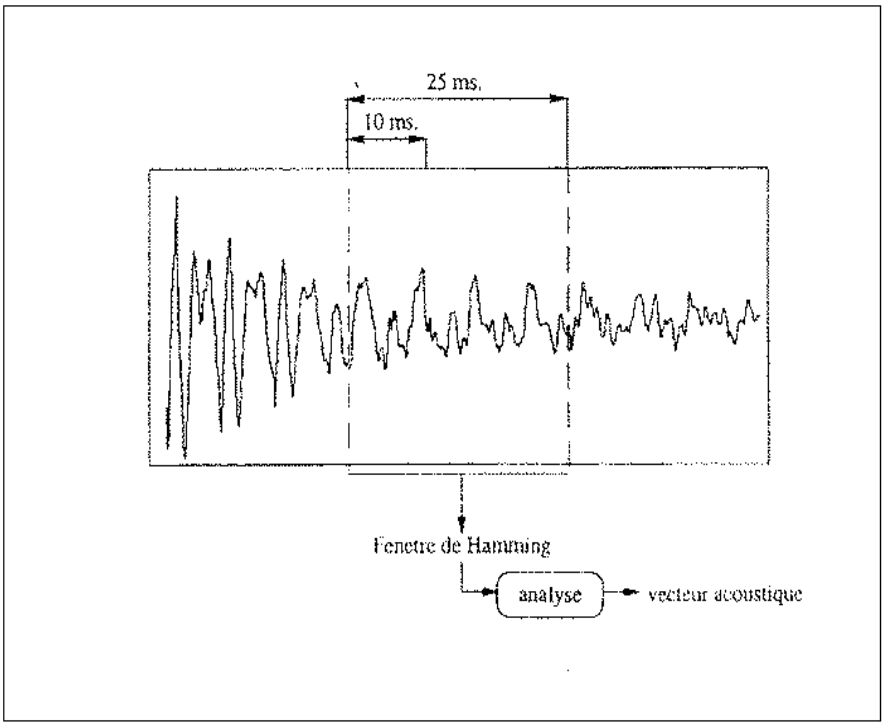


FIG. 3.4 – *Exemple de fenêtrage de Hamming (fenêtre de 25ms glissement de 10ms)*

tions et redondance) et par sa forme analogique est difficilement exploitable. La paramétrisation d'un signal de parole a pour but de proposer une représentation plus simple sous forme de vecteurs de paramètres acoustiques afin de faciliter l'extraction des informations désirées. Le signal de parole a comme caractéristique de varier lentement au cours temps, ce qui permet de le considérer comme quasi-stationnaire. Le calcul des paramètres acoustiques est ainsi réalisé en glissant avec une cadence régulière (ex : 10ms) une fenêtre de pondération d'une longueur bien définie sur tout le signal. Généralement, la longueur de la fenêtre de pondération peut varier de 20ms à 32ms. On connaît plusieurs type de fenêtrage (ex : Hamming, Hanning, Blackman, etc). En générale, le fenêtrage Hamming est le plus utilisé en traitement de signal de parole (figure 3.4). Chaque fenêtre nous permet d'avoir une trame. Les trames obtenues sur tout le signal de parole sont traitées par la suite afin de produire les vecteurs de paramètres acoustiques. Dans la littérature, il existe trois grandes catégories de paramètres :

– Paramètres issus de l'analyse spectrale : C'est la paramétrisation la plus

utilisée en RAL. Elle représente les caractéristiques physique de l'appareil phonatoire de chaque individu (exp : LPCC, MFCC) [20][48][75].

- Paramètres prosodiques : Ces paramètres illustrent en général le style d'élocution, vitesse d'élocution (débit), durée et la fréquence de pauses, pitch, l'énergie etc [1].
- Paramètres dynamiques : Le vecteur de paramètres issus des paramétrisations précédentes peut être complété par le vecteur correspondant au dérivées du premier et second ordres de ces paramètres. Ces dérivées sont calculées à partir de plusieurs trames adjacentes. Elles permettent d'introduire une information concernant le contexte temporel d'une trame courante [32].

3.3 La Modélisation

Le processus de VAL se base principalement sur la phase de modélisation des caractéristiques des locuteurs. On peut distinguer trois approches principales pour construire les modèles des locuteurs : approche vectorielle, approche connexionniste et approche statistique

3.3.1 Approche vectorielle

Elle consiste à représenter un locuteur par un ensemble de vecteurs issus directement de la phase de paramétrisation. Cette approche comporte deux techniques principales : l'alignement temporel dynamique et la quantification vectorielle.

L'Alignement Temporel Dynamique (DTW : Dynamic Time Warping)

Utilisée en mode dépendant du texte, cette technique effectue la comparaison entre la forme d'entrée à reconnaître et la forme de référence en calculant la distance entre les paramètres des deux formes. Elle détermine le meilleur chemin reliant le début et la fin des deux blocs de paramètres. Ainsi cet algorithme permet de trouver un alignement temporel optimal entre la forme d'entrée et la forme de référence. Cet alignement est réalisé par une technique de programmation dynamique [11]. Malgré les bonnes performances obtenues par cette technique, elle reste très sensible à la qualité de l'alignement et notamment le choix du point de départ des deux formes à comparer.

La Quantification Vectorielle (VQ : Vector Quantization)

La quantification vectorielle ou VQ est une méthode non paramétrique qui permet de décrire un ensemble de données par un faible nombre de vecteurs formant un dictionnaire (codebook) associé aux données. Le dictionnaire est en général calculé de façon à ce que la distance moyenne entre un vecteur issu des données et son plus proche voisin dans le dictionnaire soit la plus petite possible. En reconnaissance de locuteur ce dictionnaire est réalisé à partir des vecteurs spectraux provenant de l'analyse du signal de parole de chaque locuteur. Les performances et la rapidité de cette technique dépendent fortement de la taille du dictionnaire. En effet, plus la taille du dictionnaire est grande meilleures sont les performances, mais le processus de test devient trop lent.

3.3.2 Approche connexionniste

Les systèmes connexionnistes ou Réseaux de Neurones (RN) qui furent redécouverts et développés dans la fin des années 80, ont suscité beaucoup d'intérêt dans plusieurs domaines. Cette approche comprend une grande famille de méthodes très différentes. Chaque méthode est représentée par un réseau qui implémente une fonction de transfert globale spécifiée par l'architecture et les fonctions élémentaires du réseau. Ainsi les réseaux de neurones peuvent être considérés comme un des modèles d'approximation de fonctions générales non-paramétriques. La plupart des recherches dans le domaine reposent sur les Perceptrons Multi-Couches (MLP : Multi-Layer Perceptrons) [91] ou sur des systèmes similaires comme Radial Basis Fonctions (RBF).
Dans cette approche, un locuteur est représenté par un ou plusieurs réseaux de neurones appris directement des trames obtenues en phase de paramétrisation et permettant de le discriminer par rapport un autre ensemble de locuteurs [4] [5] [6].
Malgré la capacité des RN d'implanter des techniques discriminantes très efficaces, et leurs performances de classement, ils restent incapables de résoudre leur principal problème qui est la durée d'apprentissage importante et nécessaire pour une grande population. Le principal avantage de ces modèles est leur capacité discriminante qui n'exige pas beaucoup d'hypothèses ni beaucoup de connaissances sur l'application et qui leur permet d'être très efficaces. Par contre, il est toujours très difficile d'insérer de nouvelles données d'apprentissage sans refaire

l'apprentissage d'une grande partie du système.

3.3.3 Approche statistique

A part la DTW qui reste très sensible à l'alignement et au choix du point de départ des formes à comparer, un défaut commun à la plupart des techniques présentées précédemment est qu'elles ne prennent pas en compte l'ordre dans lesquels les vecteurs de paramètres sont présentés. L'introduction de l'approche statistique a résolu ces problèmes en utilisant des techniques permettant de construire des modèles qui tiennent compte de l'aspect temporel du signal de parole [72] [73]. Dans cette approche, les vecteurs acoustiques issus de la paramétrisation sont utilisés pour créer des modèles statistiques à long terme. Les deux techniques statistiques les plus utilisés en VAL sont : les modèles de Markov cachés (HMM) utilisés en VAL en mode dépendant du texte et les modèles de mélange de gaussiennes (GMM) qui sont utilisés en VAL en mode indépendant du texte.

Les Modèles de Markov Cachés HMM : Hidden Markov Models

Dans une modélisation par HMM, on suppose que la suite des vecteurs acoustiques d'observation est stationnaire par blocs. Ainsi, les vecteurs acoustiques d'un bloc suivent la même loi de probabilité. La modélisation d'un bloc de vecteurs acoustiques représente un état du modèle HMM. Dans cette approche, chaque entité est modélisée par une machine d'états (automate), appelée machine Markovienne et qui est composée d'un ensemble d'états et de transitions qui permettent de passer d'un état à un autre. Un modèle HMM est un modèle statistique séquentiel qui suppose que les caractéristiques observées forment une succession d'états distincts.
Soit λ un modèle Markovien de n états et $Q = (q_1,q_2,..,q_t,..,q_T)$ une séquence d'états correspondant à l'observation $O = (o_1, o_2, .., o_t, .., o_T)$ où q_t est le numéro de l'état atteint par le processus à l'instant t. L'état du modèle de Markov λ qui correspond à q_t n'étant pas directement observable, on dit qu'il est caché. D'où le nom modèle de Markov caché (Hidden Markov Model : HMM). La figure 3.5 représente un exemple de modèle de Markov. Un tel modèle est définit par :

- Un ensemble d'états $E = \{e_i \mid \ 1 \leq i \leq n\}$. sachant que le processus part de l'état e_1 à l'instant t=0 et fini à l'état e_n à l'instant t=T
- Un ensemble de probabilités initiales de se trouver dans chaque état $\Pi = \{\pi_i \mid \pi_i = P(q_1 = e_i)\ i = 1, .., n\}$.

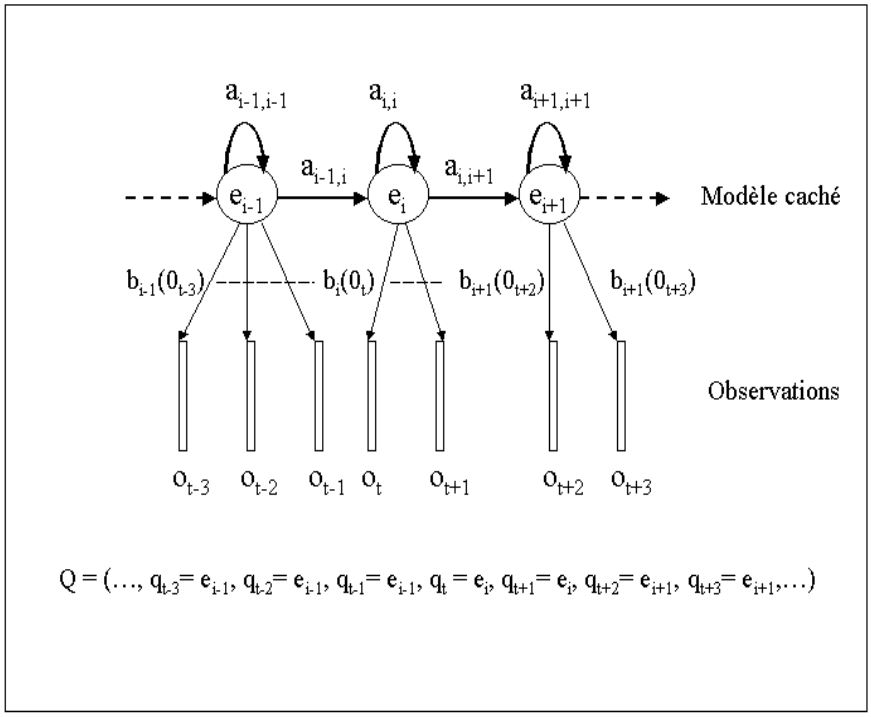


FIG. 3.5 – *Exemple d'une machine Markovienne*

- La matrice de probabilité de transition $A = \{a_{i,j}$ probabilité de transition de l'état e_i à l'état $e_j \;\; i,j = 1,..,n\}$,

$$a_{i,j} \;\; = P(q_t = e_j | q_{t-1} = e_i \;).$$

- Un ensemble de fonctions de densités associées à chaque état du modèle de Markov $B = \{b_i(.) \mid i = 1,..,n\}$. Ces fonctions de densités sont en général des mélanges de gaussiennes, mais on peut trouver aussi des lois discrètes ou encore des perceptrons multi-couches comme pour le cas des modèles HMM hybrides présentés dans [46].

Trois problèmes se posent avec les modèles de Markov :

- L'évaluation : elle consiste à calculer la probabilité de générer une séquence d'observation, $P(O \mid \lambda)$. Il existe deux méthodes pour résoudre ce problème. La méthode dite directe et qui consiste à calculer cette probabilité en énumérant toutes les séquences d'états possibles de même longueur que l'observation. Cette technique est très peu utilisée parce qu'elle demande beaucoup de temps de calcul. Un moyen plus rapide pour calculer cette probabilité est l'utilisation de l'algorithme Forward-Backward connu aussi sous le nom de l'algorithme de Baum-Welch [50].
- Décodage : le deuxième problème posé avec les HMM est le décodage qui consiste à chercher la séquence $Q = (q_1, q_2, .., q_t, .., q_T)$ d'état qui maximise la probabilité $P(O, Q \mid \lambda)$. Pour cela, l'algorithme Viterbi est le plus utilisé [72]. Il permet de chercher la séquence d'états cachés la plus probable en ne gardant que les états e_i qui maximisent la probabilité à chaque instant t. Ces probabilités sont calculées d'une façon récursive en utilisant l'équation suivante :

$$P(o_1,.., o_t, q_t = e_i | \lambda) = \max_{q_1,..q_{t-1}} P(o_1,.., o_t, q_1,.., q_t = e_i | \lambda)$$

  Une description plus détaillée du décodage de Viterbi peut être trouvée dans [50].
- Apprentissage : C'est le problème principal d'un modèle HMM. En effet, la qualité d'un système utilisant une modélisation HMM dépend principalement de la qualité de ses modèles. C'est pourquoi l'étape d'apprentissage qui consiste à estimer les paramètres des modèles HMM est très importante. Il existe plusieurs méthodes pour résoudre ce problème, les plus utilisées sont :
  - L'algorithme de Viterbi associé à des estimateurs empiriques : l'algorithme de Viterbi sert à déterminer la séquence d'états cachés la plus vrai-

semblable correspondant aux données d'apprentissage. Les paramètres des densités de probabilité de chaque état peuvent être alors ré-estimés en utilisant des estimateurs empiriques et les observations associées à chaque état le long du chemin de Viterbi.

- L'algorithme EM (Expectation-Maximisation) : Cet algorithme permet de résoudre le problème d'apprentissage en estimant de manière itérative les paramètres d'un modèle au sens du maximum de vraisemblance [19][24][96].

Le principal avantage de l'approche HMM est sa grande capacité d'apprendre les propriétés statistiques. En reconnaissance de locuteur le choix le plus fréquent consiste à utiliser un modèle dont la distribution conditionnelle dans chaque état est un mélange de gaussiennes. L'utilisation de ces modèles est plus importante dans le mode dépendant du texte [69][72] parce qu'en mode indépendant du texte l'information supplémentaire apportée par les transitions entre états n'améliore pas les performances de la reconnaissance du locuteur [74].

Les modèles de mélange de gaussiennes GMM : Gaussian Mixture Models

Depuis leur introduction par Reynolds en 1992 [74], les modèles GMM sont devenus l'état de l'art en modélisation de locuteur en mode indépendant du texte. Cette approche consiste à modéliser un locuteur par un mélange de gaussiennes qui représente une somme pondérée de M gaussiennes multi-dimensionnelles. Chaque gaussienne g_i est caractérisée par son poids p_i, un vecteur moyen μ_i de dimension d et une matrice de covariance Σ_i de dimension $d \times d$ [74][76]. L'algorithme EM est généralement utilisé lors de l'apprentissage des paramètres des modèles des locuteurs (p_i, μ_i, Σ_i). En phase de test, pour un segment de test $X = (x_1, ..., x_\tau)$, La vraisemblance pour qu'un vecteur x_t soit produit par le modèle λ (i.e : λ est un mélange de gaussiennes) est donnée par l'équation suivante :

$$f(x_t|\lambda) = \sum_{i=1}^{M} p_i \cdot f_i(x_t)$$

$$f_i(x_t) = \frac{1}{(2\pi)^{d/2} \left|\Sigma_i\right|^{1/2}} \exp\left\{-\frac{1}{2}(x_t - \mu_i)^T (\Sigma)^{-1} (x_t - \mu_i)\right\}$$

Ainsi la vraisemblance pour que le segment de test X soit produit par le modèle λ est donnée par :

$$f(X|\lambda) = \prod_{t=1}^{\tau} f(x_t|\lambda)$$

Comme nous l'avons déjà mentionné, la modélisation GMM est la modélisation la plus utilisée en RAL en mode indépendant du texte à cause de ces bonnes performances.

Remarque : *les modèles de mélange de gaussiennes peuvent être considérés comme des modèles HMM à un seul état où la fonction de densité est un mélange de gaussiennes.*

Grâce aux bonnes performances obtenues par l'approche statistique, cette dernière est devenue l'état de l'art dans le domaine de la RAL. Depuis, plusieurs travaux ont été menés pour améliorer les performances des systèmes utilisant cette approche. Ainsi des techniques d'adaptation ont été introduites pour améliorer la qualité des modèles statistiques.

Ces techniques sont basées sur la ré-estimation des paramètres des modèles des clients à partir d'un modèle générique (cf section 3.4) en utilisant les données d'apprentissage propre à chaque client.

Dans la littérature, deux techniques d'adaptation se partagent la plupart des systèmes proposés : l'adaptation MLLR (Maximum Likelihood Linear Regression) et l'adaptation MAP (Maximum A Posteriori). L'algorithme EM est très utilisé dans ces deux techniques d'adaptation.

- L'adaptation MLLR : elle consiste à transformer les paramètres d'un modèle générique en maximisant la vraisemblance des données d'apprentissage par rapport à ce modèle [55]. L'adaptation MLLR revient à rechercher λ_{ML} tel que :

  $$\lambda_{ML} = \arg\max_{\lambda} f(d_{app}|\gamma_\lambda)$$

  avec d_{app} représente les données d'apprentissage, λ représente les paramètres du modèle à estimer, $f(d_{app}|\gamma_\lambda)$ est la vraisemblance de d_{app} considérant le modèle γ_λ (dans la première iteration $\gamma_\lambda = \overline{\lambda}$ le modèle générique).
- L'adaptation MAP : dans cette adaptation, l'estimation des paramètres est réalisée en tenant compte de la connaissance *a priori* sur la distribution des paramètres [37]. L'adaptation MAP consiste à chercher λ_{MAP} tel que :

  $$\lambda_{MAP} = \eta\lambda_{d_{app}} + (1-\eta)\overline{\lambda}$$

avec $\lambda_{d_{app}}$ est le modèle obtenu par une itération sur les données d'apprentissage, $\overline{\lambda}$ est le modèle générique et η une constante à définir. En général, on prend $\eta = 0.75$

3.4 Décision

Étant donné un segment de test X et une identité proclamée λ, un système de VAL a la tâche de décider si le segment X a été prononcé par l'identité proclamée λ ou non. Plusieurs techniques du domaine de la RdF ont été appliquées, mais la plupart sont basées sur le principe de test binaire d'hypothèse suivant :

$$\mathcal{P}(X|\lambda) \underset{H_1}{\overset{H_0}{\gtrless}} \beta$$

où $\mathcal{P}(X|\lambda)$ est la vraisemblance entre X et le modèle λ, H_0 est l'hypothèse que c'est bien λ qui a prononcé le segment X c'est-à-dire qu'il s'agit bien d'un accès client, H_1 est l'hypothèse que le segment X n'a pas été prononcé par l'identité proclamée λ et β est un seuil de décision qui peut être global ou dépendant du locuteur. Cependant, cette règle de décision a été rapidement mise à l'écart et améliorée à cause de la variabilité inter-locuteur (stress, émotion, etc) et intra-locuteur (transmission, bruit de fond, etc). Ainsi une nouvelle règle de décision a vu le jour en s'inspirant du même principe décrit précédemment et qui est connue sous le nom de "similarity domain normalisation". Cette règle de décision prend la forme suivante

$$\frac{\mathcal{P}(X|\lambda)}{\mathcal{P}(X|\overline{\lambda})} \underset{H_1}{\overset{H_0}{\gtrless}} \beta$$

où $P(X/\overline{\lambda})$ est la vraisemblance entre X et un modèle générique $\overline{\lambda}$. Ce modèle générique est connu sous le nom du modèle anti-locuteur.
Dans la littérature, deux approches ont été proposées concernant la construction du modèle $\overline{\lambda}$. La première a été proposée dans [47], elle consiste à calculer la vraisemblance $P(X/\overline{\lambda})$ en utilisant un ensemble de locuteurs $\{\overline{\lambda}_i\}$ appelé l'ensemble des locuteurs cohortes. Dans [55], cette vraisemblance est représentée par le maximum des $P(X/\overline{\lambda}_i)$ et par la moyenne des $P(X/\overline{\lambda}_i)$ dans [68]. L'ensemble des locuteurs cohortes $\{\overline{\lambda}_i\}$ représente l'ensemble des locuteurs les plus proches du modèle λ. Ces locuteurs sont choisis parmi un ensemble de locuteurs plus

général représentant toute la population d'une application . Une deuxième approche a été proposée dans [18][59] qui consiste à utiliser un modèle général $\overline{\lambda}$ représentant toute la population d'une application. Ce modèle est connu sous le nom du modèle indépendant du locuteur ou modèle du monde.
En général, le modèle du client λ est construit par adaptation du modèle du monde $\overline{\lambda}$. Cette adaptation peut être réalisée en utilisant les données d'apprentissage du client λ par une adaptation du type MAP [37] ou MLLR [55].

3.4.1 Techniques de normalisation

Depuis 1996, la plupart des travaux du domaine de la VAL ont été consacré au module de décision et plus particulièrement aux techniques de normalisation des scores. Une des premières techniques de normalisation développée est la technique Znorm [42] [43] [63]. Elle consiste à centrer et réduire les scores des accès imposteurs en réalisant dans la phase de développement des accès pseudo-imposteurs sur le modèle de chaque client.
Supposons que $S_\lambda(X)$ le score obtenu suite au test de X sur le modèle λ. La normalisation de ce score est obtenue par l'équation suivante :

$$S'_\lambda(X) = \frac{S_\lambda(X) - \mu_\lambda}{\sigma_\lambda} \tag{3.1}$$

où μ_λ et σ_λ sont respectivement la moyenne et la variance des accès pseudo-imposteurs sur le modèle λ.
Plusieurs variantes de cette technique de normalisation ont été développées. Elles se basent principalement sur les connaissances a priori qu'on peut avoir et qui peuvent facilement être intégrées. Ainsi et à cause des variations du combiné, D. Reynolds a proposé une normalisation qu'il a nommé Hnorm [78], qui n'est autre que la normalisation Znorm dépendante du combiné. D'autres normalisations ont été proposées dernièrement comme la Tnorm qui consiste à centrer et réduire les scores des accès imposteurs en réalisant des accès pseudo-imposteurs avec chaque segment de test sur des modèles des pseudo-imposteurs [2]. Ainsi la normalisation du score $S_\lambda(X)$ du test de X sur le modèle λ est réalisée de la même façon que dans l'équation (3.1) en utilisant μ_X et σ_X représentant respectivement la moyenne et la variance des accès pseudo-imposteurs du segment de test X sur un ensemble de modèles des pseudo-imposteurs.

3.5 Evaluation des systèmes de RAL

Dans cette section, nous allons décrire les différentes mesures les plus utilisées pour évaluer un système de RAL [12]. Mais avant, voici quelques définitions très utiles que nous utilisons très fréquemment dans le domaine de la RAL :

- Client : est un locuteur de la base de données dont le système dispose de son modèle et qui utilise ce système sous sa vraie identité.
- Imposteur : cette notion est spécifique au système de VAL. Un imposteur est tout locuteur utilisant le système sous une identité qui n'est la pas sienne.
- Pseudo-imposteur : est un locuteur dont le système dispose de quelques segments de son signal de parole qui seront utilisés pour réaliser des accès d'imposteur tout en sachant que c'est un imposteur. Il est utilisé en phase de développement du système.

3.5.1 Typologie d'erreurs et mesures de performances

En RAL, chaque tâche possède ses propres erreurs. Dans cette section nous rappelons la typologie d'erreurs des deux tâches les plus utilisées, IAL et VAL.

- En identification du locuteur, on peut parler de deux types d'erreurs :
  - mauvaise identification : c'est le cas où le système propose une identité qui ne correspond pas à celle du locuteur présenté.
  - non-détection : cette erreur est caractéristique des système d'identification de locuteur dans un ensemble ouvert. Elle correspond au cas où le système n'a pas pu identifier le locuteur présenté alors que ce dernier a son modèle dans la base de référence.

La mesure des performances des systèmes d'identification du locuteur se base sur le Taux d'Identification Correcte (TIC) obtenu en phase de test :

$$TIC = \frac{\#\ tests\ ayant\ amené\ à\ une\ identification\ correcte}{\#\ tests\ total}$$

- En vérification du locuteur, il existe deux type d'erreurs :
  - Fausse Acceptation (FA) : Elle correspond au cas où le système accepte un locuteur qui a proclamée une identité qui n'est pas la sienne. Une fausse acceptation est une erreur où le système accepte un imposteur.

– Faux Rejet (FR) : C'est le cas où le système rejette un locuteur qui a proclamé sa vraie identité. Autrement dit, c'est quand le système rejette un client.

Les mesures de performances d'un système de VAL se base principalement sur le Taux des Fausses Acceptations (TFA) et le Taux des Faux Rejets (TFR) obtenu en phase de test :

$$TFA = \frac{\#\ tests\ ayant\ amené\ à\ une\ fausse\ acceptation}{\#\ tests\ imposteurs}$$

$$TFR = \frac{\#\ tests\ ayant\ amené\ à\ un\ faux\ rejet}{\#\ tests\ clients}$$

Les performances d'un système de VAL peuvent être présentées sous forme d'une seule courbe appelée courbe DET[1] [61] ou encore courbe ROC[2] [64] sur laquelle les TFA sont données en fonction des TFR. Pour construire cette courbe, on calcule un couple (TFA, TFR) pour chaque valeur de seuil de décision variant de la plus petite valeur des scores obtenus en phase de test à la plus grande valeur. Les performances des systèmes de VAL sont souvent comparés selon un point particulier de ces courbes qui est le Taux d'Égale Erreur (TEE) et qui correspond au point de la courbe où $TFA = TFR$. Une autre mesure permet d'évaluer les performances d'un système de VAL est HTER (Half Total Error rate). Cette mesure est utilisée quand le seuil de décision est fixé à priori. Le $HTER$ représente la moyenne de TFA et TFR.

$$HTER = \frac{TFA + TFR}{2}$$

3.5.2 Les évaluations NIST, consortium ELISA

Depuis 1996 NIST[3] organise des évaluations en reconnaissance automatique du locuteur en mode indépendant du texte [à travers des lignes téléphoniques] avec la participation d'une quinzaine de laboratoires de recherche universitaires et industriels [63]. Les objectifs de ces évaluations sont :

– L'exploration des nouvelles idées dans le domaine de la reconnaissance du locuteur.

[1] Detection Error Trade-off
[2] Receiver Operating Characteristic
[3] National Institute of Standards and Technologies

- Le Développement et l'implémentation des techniques qui illustrent ces idées
- Mesurer la performance de ces techniques.

Chaque évaluation a un plan officiel qui contient un ensemble de règles et de tâches qu'il faut respecter afin de valider la participation. Depuis la création des évaluations NIST, l'ENST a participé à ces évaluations dans le cadre du consortium ELISA [25][40][58]. Ce consortium est le fruit d'un effort commun de développement des laboratoires suivants :

- ENST : Ecole Nationale Supérieure des Télécommunications.
- IRISA : Institut de Recherche en Informatiques et Sciences Avancées.
- LIA : Laboratoire d'Informatique d'Avignon.
- EPFL : Ecole Polytechnique Fédérale de Lausanne.
- IDIAP : Institut Dalle Molle d'Intelligence Artificielle Perceptive
- VT-BRNO : BRNO University of Technology Faculty of Electrical engineering and computer science
- RMA : Royal Military Academy of Belgium
- Rice University-FPMs : Rice University avec la Faculté Polytechnique de Mons

3.6 Domaines d'applications

Dans ce paragraphe on donne quelques exemples d'applications en RAL et que l'on peut regrouper en trois catégories principales : applications sur sites géographiques, juridiques, téléphoniques.

3.6.1 Applications sur sites géographiques

Cette catégorie concerne les applications qui se trouvent sur un site géographique particulier, elles sont utilisées principalement pour limiter l'accès à des lieux privés . Voici quelques exemples de ce type d'applications :

- Verouillage automatique : ces applications sont utilisées comme une sorte de verrous électroniques comme par exemple la protection de domicile, garage, bâtiment, etc.
- Validation des transactions sur site (comme contrôle supplémentaire au niveau des distributeurs bancaires) .

- Accès aux lieux de production des usines : qui sont en général réservés aux employés, ouvriers et inspecteurs afin de protéger le secret de la production et du matériel.

L'intérêt de ce type d'application est :

- D'abord l'environnement est facilement contrôlable.
- La vérification du locuteur a un rôle dissuasif.
- La reconnaissance vocale peut être associée à d'autre techniques de reconnaissance d'identité (ex : analyse du visage, des empreintes digitales, etc).
- L'utilisateur peut avoir son modèle sur lui (ex : sur la puce d'une carte)

3.6.2 Applications téléphoniques

Ce type d'applications utilise le téléphone comme un moyen matériel de communication entre l'homme et la machine. C'est la catégorie la plus importante parce qu'elle permet de vérifier ou identifier le locuteur à longue distance. Il existe plusieurs applications dans cette catégorie et parmi elles :

- Validation de transactions bancaires par téléphone (pour améliorer le service bancaire, ainsi que pour valider légalement la transaction effectuée)
- Accès à des bases de données pour plus de sécurité et pour plus de protection (ex : consultation d'émail, consultation de répondeur, etc).
- Accès à des services téléphoniques (ex : téléphoner sur son compte de facturation personnelle de n'importe quelle ligne téléphonique comme par exemple le service de la carte FanceTélécom).

Les inconvénients de ce type d'applications sont principalement :

- L'environnement est difficilement contrôlable parce que la qualité des lignes téléphoniques peut varier considérablement d'un appel à un autre, ainsi que le bruit de fond produit par le lieu d'appel (bar, restaurant, bureau, etc).
- Les applications exigent le stockage des données de manière centralisée .
- Il est impossible d'utiliser d'autres techniques de reconnaissances (excepté un code numérique tapé sur des touches à fréquences vocales).

3.6.3 Applications juridiques

Enfin on trouve le domaine d'applications qui pose actuellement le plus de problèmes, c'est le domaine juridique. La reconnaissance de locuteur est utilisée par exemple pour :

- L'orientation des enquêtes.
- La constitution des éléments de preuves au cours d'un procès.

Dans ces applications on trouve beaucoup plus d'inconvénients que d'avantages :

- La quantité de la parole à disposition est en général très limitée .
- Les conditions d'environnement sont très mauvaises .
- Les locuteurs impliqués sont très rarement coopératifs .
- Il existe souvent d'autres techniques plus sûres pour effectuer une reconnaissance (empreintes digitales, enregistrement vidéo, etc) .

Chapitre 4

SVM pour la vérification du locuteur en mode dépendant du texte

Dans ce chapitre nous allons présenter notre approche concernant l'utilisation des SVM en vérification du locuteur en mode dépendant du texte. Ce travail a été une des nos premières tentatives d'appliquer les SVM en vérification de locuteur. Il rentre dans le cadre du projet Européen PICASSO. Dans cette approche, nous proposons une nouvelle modélisation basée sur une reconnaissance phonétique de la parole qui nous permet de construire des vecteurs de taille fixe que nous utiliserons comme vecteur d'entrée pour les SVM.

4.1 Projet PICASSO

Le project PICASSO "Pioneering Caller Authentication for Secure Service Operation" est un projet Européen qui répond au besoin de sécurisation dans le domaine des télécommunications (exp : utilisation de cartes téléphoniques), des institutions financières (exp : transactions bancaires) ou toutes autres entreprises exploitant le télé-commerce [13]. Elle représente la suite d'un précédent projet Européen nommé CAVE "CAller VErification" [69] qui a mis en place le système de base utilisé dans le projet PICASSO. L'objectif du projet PICASSO est de développer et d'intégrer de nouvelles technologies combinant la reconnaissance automatique de la parole et la vérification automatique du locuteur pour des services utilisant le téléphone comme moyen de communication. Ce projet a

réuni un certain nombre de partenaires[1] industriels et universitaires qui ont pour tâche de valider les systèmes développés. Le projet PICASSO a été soutenu par la commission Européenne et rattaché au secteur des nouvelles technologies en communications et plus particulièrement de l'usage des technologies du traitement du langage appliquées à la télématique.

4.2 Protocole expérimental

4.2.1 Base de données

Dans le cadre du Projet PICASSO, nous avons utilisé la base de données POLYVAR. C'est une base fournie par l'IDIAP (Institut Dalle Molle d'Intelligence Artificielle Perceptive), un des partenaires au projet PICASSO [22]. Elle contient 144 locuteurs (85 hommes et 58 femmes), chaque locuteur a réalisé entre 1 et 255 sessions. Une session est un enregistrement qui contient 17 mots de passes de longueur différente qui varie de 3 phonèmes pour le mot "guide" à 13 phonèmes pour le mot "galerie du manoir", trois répétitions de nombres et 10 phrases phonétiquement équilibrées de la langues française. Dans ce projet, nous utilisons seulement les 17 mots de passes.
En ce qui concerne notre approche basée sur les SVM, nous avons utilisé un sous ensemble de la base de données POLYVAR dans une application de *mots de passe publics*. C'est à dire que nous supposons que le système connaît les mots de passe ainsi que leur transcription phonétique. Dans ce sous ensemble, nous avons limité nos tests sur cinq mots de passe de longueur variable (quitter, cinéma, annulation, exposition et manifestation). En ce qui concerne les locuteurs, notre sous ensemble est composé de 20 locuteurs (12 hommes, 8 femmes) considérés comme clients et qui ont été choisis parmi ceux qui ont enregistré plus de 21 sessions. 38 locuteurs dont 19 hommes et 19 femmes ont été utilisés pour réaliser des accès pseudo-imposteurs dans la phase d'apprentissage pour apprendre les modèles SVM de chaque client. Pour chaque client nous avons utilisé les cinq premières répétitions de chaque mot de passe pour l'apprentissage et les 16 dernières répé-

[1]PTT telecom BV, Katholic University of Nijmegen (KUN), KPN Research, Kungel Tekniska Högskolan (KTH), Ecole National Supérieure des Télécommunications (ENST), Institut National de Recherche en Informatique et en Automatique (INRIA), Vocalis Limited, Fortis Nederland NV, Institut Dalle Molle d'Intelligence Artificielle Perceptive (IDIAP), Union Bank of Switzerland (UBILAB)

titions pour le test.
Dans la phase de test nous avons réalisé 320 accès clients et 7500 accès imposteurs pour chaque mot de passe, ce qui nous donne 1600 accès clients et 37500 accès imposteurs tous mots de passe confondus.

Apprentissage Incrémental Dans nos expériences, nous avons voulu tester l'approche d'apprentissage incrémental. Pour cela, nous avons mené 4 expériences avec 4 ensembles d'apprentissage. L'ensemble de test est le même dans toutes ces expériences. Dans le premier ensemble d'apprentissage nous avons utilisé seulement les deux premières répétitions des mots de passes et les 19 répétitions des pseudo-imposteurs (les pseudo-imposteurs sont de même sexe que le client) pour construire le modèle SVM du client. Ensuite, on ajoute à l'ensemble d'apprentissage une répétition des mots de passe du client et en apprend un nouveau modèle SVM jusqu'à l'utilisation des 5 répétitions de chaque mot de passe tout en gardant les 19 répétitions des pseudo-imposteurs. Le tableau suivant présente la répartition des données sur les 4 ensembles d'apprentissage utilisés pour les 4 expériences :

L'ensemble d'apprentissage	1	2	3	4
Le nombre de répétitions d'un mot de passe	21	22	23	24
Le nombre de répétitions d'un mot de passe par client	2	3	4	5
Le nombre de répétitions d'un mot de passe par les 19 pseudo-imposteurs	1 par locuteur	1 par locuteur	1 par locuteur	1 par locuteur

NB : Dans nos expériences, on a décidé de se mettre dans la situation la plus délicate qu'on puisse avoir dans une application réelle. C'est la cas où l'imposteur connaît le mot de passe du client. *C'est une application de mots de passe publics.*

4.2.2 Paramétrisation

La paramétrisation MFCC "Mel Frequency Cepstral Coefficients" a été utilisée. Des vecteurs de 39 coefficients sont calculés sur des fenêtres de 25ms tout les 10ms. Les 13 premiers coefficients représentent 12 coefficients cepstreaux plus le logarithme de l'énergie, les 26 derniers coefficients sont une dérivation du premier et second ordre des 12 premiers coefficients.

Pour la phase de modélisation et décision nous avons proposé une nouvelle technique de construction des vecteurs d'entrée pour les SVM basée sur la transcription phonétique des mots de passe. Une description détaillée de cette modélisation est présentée dans la prochaine section.

Pour évaluer notre nouveau système basé sur les SVM, nous avons utilisé un système de référence basé sur les techniques classiques représentant l'état de l'art en VAL en mode dépendant du texte (les HMM en modélisation et LLR en décision).

Description du système de référence : La même paramétrisation est utilisée pour tous les systèmes. En ce qui concerne la modélisation, la technique HMM a été utilisée. Chaque locuteur est représenté par l'ensemble de modèles de phones correspondant à un mot de passe et qui sont des HMM à 3 états et une gaussienne par état. Ces modèles HMM sont obtenus par adaptation MAP des modèles de phones indépendant du locuteur. La modélisation de chaque client est réalisée en deux étapes :

- Segmentation des mots de passe : cette phase consiste à segmenter le signal de parole de chaque répétition du mot de passe utilisée en apprentissage. Cette segmentation est réalisation par un alignement forcé en utilisant la vraie transcription des mots de passe et les modèles de phones indépendant du locuteur. Les modèles HMM utilisés dans cette étape sont des modèles HMM à 3 états et 3 gaussiennes par états.
- Modélisation des clients : Pour modéliser un client, on reprend la même segmentation obtenue dans la première étape. Le modèle HMM indépendant du locuteur à 3 états et 1 gaussienne de chaque phone est adapté en utilisant toute les occurrences de ce phone apparaissant dans les répétions dédiés pour l'apprentissage.

En phase de décision, la technique LLR a été utilisée.

4.3 Approche proposée

4.3.1 Nouvelle modélisation pour les SVM

Dans cette section, nous allons décrire la nouvelle modélisation que nous proposons pour utiliser les SVM. Cette modélisation est constituée de deux étapes :

- Reconnaissance de la parole : Cette étape consiste a reconnaître le mot de passe prononcé en utilisant les phonèmes comme unité acoustique. L'objectif de cette étape est d'aligner le segment de parole de chaque mot de passe à reconnaître avec sa vraie transcription phonétique. Les modèles génériques des phones qui sont au nombre de 35 plus un modèle du silence ont été appris sur un sous ensemble la base de données POLYPHONE [22]. Ce sous ensemble contient 10000 phrases phonétiquement équilibrées de la langue française, enregistrées à travers les lignes téléphoniques (comme c'est le cas de la base POLYVAR) par 1000 locuteurs. Le tableau suivant donne un exemple de la segmentation obtenue suite à la reconnaissance du mot de passe "cinéma" :

trame de début	trame de fin	le phonème reconnu
2700000	3700000	ss
3700000	4400000	ii
4400000	5200000	nn
5200000	5600000	ai
5600000	6700000	mm
6700000	8800000	aa

Les modèles des phones sont des HMM de 3 états à 3 gaussiennes par état. Le logiciel HTK (Hiden Markov Models ToolKit) version 1.4 a été utilisé pour la paramétrisation, la modélisation HMM, la reconnaissance des mots de passe et le calcul des vraisemblances nécessaire pour la construction des vecteurs d'entrée pour les SVM.

- Modélisation SVM des clients : Elle consiste à construire les modèles SVM des clients. Pour construire ces modèles nous aurons besoin de quelques répétitions clients de chaque mot de passe pour construire les vecteurs d'entrée SVM de la classe client et de quelques répétitions pseudo-imposteurs du même mot de passe pour construire les vecteurs d'entrée SVM représentant la classe imposteurs.

– Construction des vecteurs d'entrée SVM : Rappelons qu'une répétition d'un mot de passe nous permet de construire un vecteur SVM. Quel que soit le locuteur derrière cette répétition (client ou pseudo-imposteur en phase d'apprentissage, client ou imposteur en phase de test), la construction du vecteur d'entrée est réalisée de la même façon. Comme nous l'avons déjà mentionné, cette phase se base principalement sur la segmentation du mot de passe obtenue lors de la reconnaissance. Ainsi pour chaque phonème reconnu, on calcule le vecteur de vraisemblance entre ce phonème et les 35 modèles génériques des phones de la langue française. Le vecteur d'entrée des SVM n'est autre que la concaténation des vecteurs des vraisemblances obtenus en traitant tout les phones du mot de passe. Ce vecteur est de dimension (35 $\times$ le nombre de phonèmes de son mot de passe). La figure 4.1 représente un exemple de la construction du vecteur d'entrée pour une répétition du mot de passe "cinéma".

Après avoir traité toutes les répétitions client et pseudo-imposteurs d'un mot de passe, le logiciel "SVM software[2]" développé par Royal Holloway University of London a été utilisé pour apprendre les modèles SVM pour chaque client et chaque mot de passe.

Décision

Après la modélisation des locuteurs par la technique expliquée plus haut, on procède à l'application des SVM sur les données de test. Pour chaque mot de passe prononcé, on construit un vecteur suivant la technique expliquée dans la section précédente. Ainsi la décision est prise suivant le score obtenu par *class(x)*, la fonction de classification des SVM donnée par l'équation (2.8). Si le score est supérieur à zéro on considère que c'est un accès client sinon c'est un accès imposteur. Pour nos expériences on a testé les 4 noyaux suivants :
Le noyau linéaire :

$$K(x, y) = x \cdot y$$

Le noyau polynomial de degré 2 :

$$K(x, y) = [(x \cdot y) + 1]^2$$

Le noyau RBF (Radial Basis Fonction) avec $\sigma^2 = 50$:

$$K(x, y) = exp\left(-0.01 \ \ |x - y|^2\right)$$

[2] http ://svm.dcs.rhbnc.as.uk

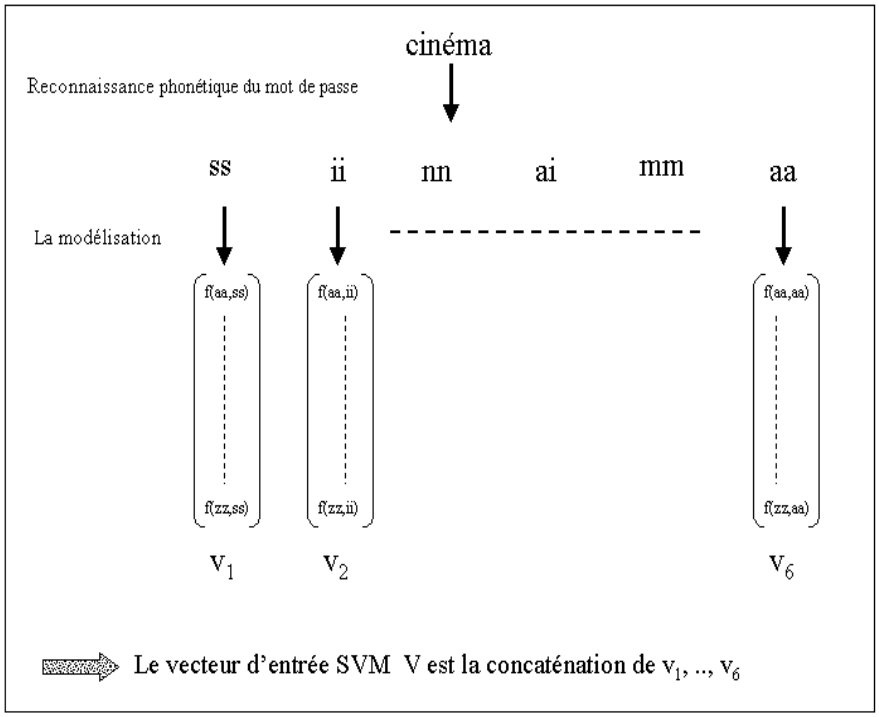


FIG. 4.1 – *Exemple de la construction du vecteur d'entrée pour une répétition du mot de passe "cinéma"*

4.3.2 Résultats :

Les résultats sont présentés sous formes de 4 tableaux correspondant chacun à un de nos 4 ensembles d'apprentissages. Chaque tableau présente les HTER (Half Total Error Rate) et leurs intervalles de confiance[3] obtenus sur le même ensemble de test pour chaque mot de passe et pour chaque noyau utilisé pour les SVM ainsi que le système de référence (cf. section 4.2.2).

	Linéaire	Poly degré 2	RBF	Sys de réf (LLR)
Manifestation	29% ± 1.00	29% ± 1.00	30% ± 1.01	19.5% ± 0.87
Exposition	27% ± 0.98	27% ± 0.98	28% ± 0.99	19.5% ± 0.87
Annulation	28% ± 0.99	29% ± 1.00	29% ± 1.00	21% ± 0.90
Cinéma	31% ± 1.02	31% ± 1.02	33% ± 1.03	22.5% ± 0.92
Quitter	35% ± 1.05	36% ± 1.06	37% ± 1.07	24% ± 0.94

Tableau 1 : Les HTER et leurs intervalles de confiance obtenus en utilisant l'ensemble d'apprentissage 1

	Linéaire	Poly degré 2	RBF	Sys de réf (LLR)
Manifestation	23% ± 0.93	24% ± 0.94	22% ± 0.92	14% ± 0.77
Exposition	23% ± 0.93	25% ± 0.96	21% ± 0.90	16% ± 0.81
Annulation	22% ± 0.92	22% ± 0.92	21% ± 0.90	16.5% ± 0.82
Cinéma	27% ± 0.98	28% ± 0.99	26% ± 0.97	18% ± 0.85
Quitter	32% ± 1.03	32% ± 1.03	32% ± 1.03	20% ± 0.89

Tableau 2 : Les HTER et leurs intervalles de confiance obtenus en utilisant l'ensemble d'apprentissage 2

	Linéaire	Poly degré 2	RBF	Sys de réf (LLR)
Manifestation	19% ± 0.86	20% ± 0.89	19% ± 0.86	11.5% ± 0.71
Exposition	18% ± 0.85	18% ± 0.85	18% ± 0.85	13% ± 0.74
Annulation	20% ± 0.89	20% ± 0.89	19% ± 0.86	15% ± 0.79
Cinéma	24% ± 0.94	24% ± 0.94	26% ± 0.97	16% ± 0.81
Quitter	30% ± 1.01	29% ± 1.00	29% ± 1.00	16.5% ± 0.82

Tableau 3 : Les HTER et leurs intervalles de confiance obtenus en utilisant l'ensemble d'apprentissage 3

[3] L'intervalle de confiance de niveau 0.95 d'une proportion P = P ± $1.96\sqrt{\frac{P(1-P)}{N}}$ où N est la taille de l'échantillon

	Linéaire	Poly degré 2	RBF	Sys de réf (LLR)
Manifestation	19% ± 0.86	19% ± 0.86	18% ± 0.85	10% ± 0.66
Exposition	17% ± 0.83	17% ± 0.83	17% ± 0.83	12% ± 0.72
Annulation	17% ± 0.83	18% ± 0.85	17% ± 0.83	13% ± 0.74
Cinéma	22% ± 0.92	22% ± 0.92	22% ± 0.92	14.5% ± 0.78
Quitter	26% ± 0.97	26% ± 0.97	26% ± 0.97	15% ± 0.79

Tableau 4 : Les HTER et leurs intervalles de confiance obtenus en utilisant l'ensemble d'apprentissage 4

4.3.3 Interprétations :

En comparant les résultats des 4 tableaux représentant les HTER obtenus suite à l'utilisation de nos 4 ensembles d'apprentissage, on constate que les performances de notre système SVM s'améliorent à fur et à mesure que le nombre de données d'apprentissage augmente (une répétition de client de mot de passe est ajoutée à chaque fois qu'on change d'ensemble d'apprentissage en allant de l'ensemble d'apprentissage 1 qui contient 2 répétitions de client de mots de passe et en arrivant à l'ensemble d'apprentissage 4 qui contient 5 répétitions de client de mots de passe). Ainsi, le HTER est autour de 30% ± 0.45 sur le premier tableau, 25% ± 0.42 sur le tableau 2 puis 22% ± 0.41 sur le tableau 3 est 20% ± 0.39 sur le tableau 4. Notons qu'il n'y a aucun recouvrement entre les intervalles de confiance des HTER. Cette amélioration des performances a été attendue et tout à fait explicable puisque à chaque fois on ajoute une répétition des mots de passe du client à l'ensemble d'apprentissage ce qui nous permet d'avoir plus de données pour une meilleure discrimination entre la classe client et la classe imposteurs. Ces résultats confirment et valident d'autres travaux menés dans le cadre du projet PICASSO sur l'efficacité de l'apprentissage incrémental [34][49][51].
Les résultats montrent également que le noyau RBF est le plus adapté à notre application puisque c'est le noyau qui a donné les meilleures performances (même si la différence n'est pas très significative), sauf sur le tableau 1 où les meilleurs résultats sont obtenus par le noyau linéaire à cause du peu de données dans l'ensemble d'apprentissage 1 qui ne permettent pas une bonne estimation des paramètres du noyau RBF. En s'intéressant aux résultats obtenus par chaque mot de passe pour étudier l'influence de la longueur des mots de passe, on constate que les meilleures performances sont obtenues par les deux mots "exposition et

annulation", alors que l'on s'attendait à ce que le mot de passe "manifestation" donne les meilleures performances. En plus nous avons remarqué que les performances obtenues par le mot de passe "quitter" sont étrangement très élevées. Nos résultats obtenus au niveau de chaque mot de passe ne nous permettent pas de mettre une relation entre la longueur du mot de passe et les performances obtenues sauf si on écarte les résultats obtenus sur le mot de passe "manifestation". Dans ce cas, on peut conclure que plus le mot de passe est long plus les performances sont bonnes. Cela peut être expliqué par le fait que plus le mot de passe est long plus il y a des données qui permettent de mieux discriminer les clients des imposteurs. Dans le cadre du projet PICASSO nous avons mené une étude sur l'influence de la longueur des mots de passe sur les performances d'un systèmes de VAL basé sur les techniques classiques HMM et LLR. Les résultats de cette étude montrent qu'effectivement plus le mot de passe est long plus les performances sont bonnes [51].
Les résultats obtenus par notre système SVM s'étant avérés moins bons que ce que nous prévoyions, nous avons décidé d'étudier la base de données POLYVAR en détails. Malheureusement, nous avons relevé plusieurs anomalies concernant l'enregistrement de plusieurs fichiers de notre protocole. Ce problème d'enregistrement a touché pratiquement tous les mots de passe et principalement les deux mots de passe "manifestation et quitter". Nous avons signalé ce problème à l'IDIAP, le distributeur de la base POLYVAR qui a confirmé l'existence de ce problème et qui nous a envoyé une nouvelle version plus propre.
Les résultats du système de référence (présentés à titre indicatif), ainsi que tous les travaux dont nous faisons référence dans ce paragraphe sont obtenus sur la version propre de la base de données. Ce qui explique la grande différence ($\sim 8\%$ en moyenne) constaté entre les performances de notre système SVM et le système de référence.

4.4 Bilan

Dans ce chapitre, nous avons présenté une nouvelle approche qui permet d'utiliser les SVM pour la vérification du locuteur en mode dépendant du texte. Cette approche est basé sur une reconnaissance de la parole utilisant des modèles génériques HMM des phones pour construire les vecteurs d'entrée pour les SVM. Nous avons expérimenté cette approche dans le cadre du projet Européen PICASSO

dans une application utilisant des mots de passe publics. Les résultats obtenus peuvent être considérés comme très encourageants compte tenu des problèmes d'enregistrement sur la première version de la base de données POLYVAR sur laquelle nous avons expérimenté notre approche. Suite aux engagements de l'ENST dans le cadre du projet PICASSO, nous n'avons pas eu la possibilité de refaire ces expériences pour pouvoir valider notre approche. Dernièrement, dans le cadre d'un autre projet nommé MAJORDOME, nous avons repris ces expériences, mais malheureusement nous n'avons pas encore les résultats pour pouvoir les présenter dans ce rapport de thèse.

Chapitre 5

SVM pour la vérification du locuteur en mode indépendant du texte

Dans ce chapitre nous allons décrire notre approche concernant l'utilisation des SVM en vérification du locuteur en mode indépendant du texte. Mais avant d'aborder cette partie qui constitue le travail principal de cette thèse, nous commençons par un historique sur l'utilisation des SVM en reconnaissance automatique des locuteurs.

5.1 Historique

Depuis que les SVM ont vu le jour en 1995 [91], plusieurs chercheurs du domaine de la reconnaissance de formes ont commencé à s'y intéresser. Principalement, en traitement d'image, on peut citer les brillants travaux réalisés sur la reconnaissance de lettres et de chiffres manuscrits [29][66] et sur la détection du visage [67], ainsi que le travail de B. Schôlkopf en reconnaissance d'objet en 3D [84]. S. Benyacob s'est intéressé aux SVM pour faire de la fusion des données de différents experts pour l'identification biométrique [8]. Les résultats intéressants obtenus par ces applications ont incité les chercheurs d'autres disciplines comme la reconnaissance de locuteur à s'intéresser aux SVM. Sachant que les SVM exigent des vecteurs d'entrée de taille fixe, leur adaptation au RAL est moins évidente que dans le cas du traitement d'image. Si une image peut être facilement représentée par un vecteur fixe que ce soit en 2D et 3D, le signal de

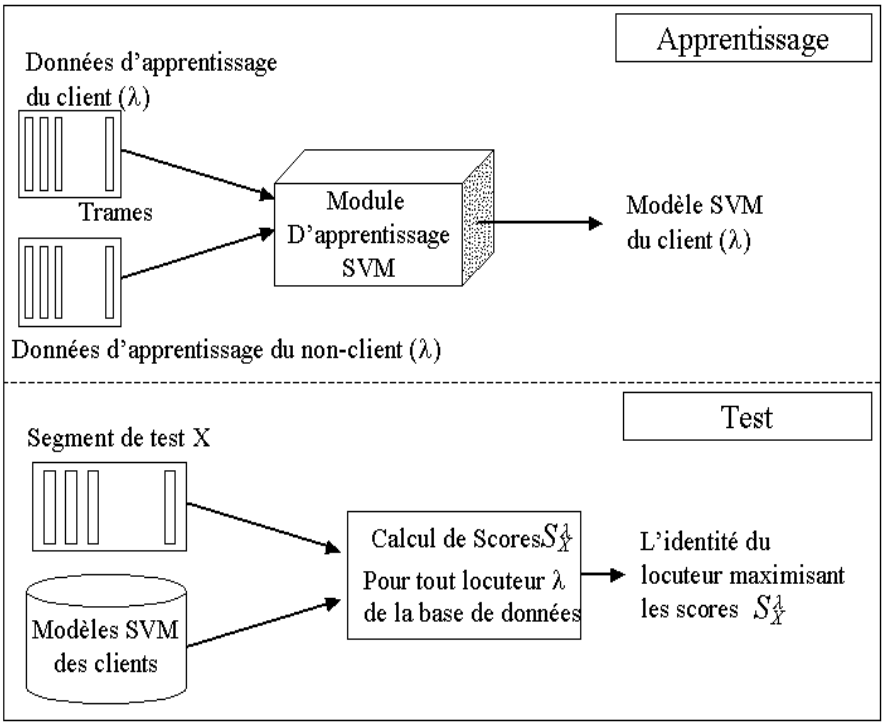


FIG. 5.1 – *La structure du premier système utilisant les SVM pour IAL proposé par M.Schmidt et H. Gish*

parole est difficilement représentable par un vecteur fixe puisqu'il est non délimité dans le temps. La durée d'un signal de parole varie de quelques secondes à plusieurs minutes. Ainsi pour adapter les SVM à toute application utilisant le signal de parole, il faudrait trouver une nouvelle représentation de données qui permette de fournir un vecteur de taille fixe quelle que soit la longueur du signal de parole à traiter. La première idée qu'on peut avoir est d'utiliser les trames obtenues suite à la paramétrisation du signal. Cette idée a été mise en place par M. Schmidt et H. Gish du laboratoire de BBN Systems and Technologies en 1996 pour une application d'identification du locuteur en mode indépendant du texte [82][83]. La figure 5.1 représente le système proposé dans cette application dans les deux phases apprentissage et test. Dans le système proposé par M. Schmidt et H. Gish un modèle SVM est construit pour chaque client λ de la base de données en utilisant toutes les trames calculées à partir des segments de signal de parole destinés pour l'apprentissage contre toutes les trames des segments de signal de parole d'apprentissage de tous les autres locuteurs de la base de données. Dans la

phase de test et pour chaque client λ, un score S_λ est calculé grâce à l'équation suivante :

$$S_\lambda^X = \sum_{t \in X} f_\lambda(t)$$

où X est le segment de test, t une trame du segment X, λ est le client et f_λ est le classifieur SVM du client λ. Ainsi le locuteur qui maximise les scores S_λ^X est retenu comme le bon locuteur. Tenant en compte que c'était la première tentative d'adapter la technique SVM pour la reconnaissance du locuteur, les résultats obtenus par ce système s'avèrent être intéressants puisqu'ils égalent les performances obtenues par un système classique basé sur les techniques GMM pour la modélisation et LLR pour la décision. Pour plus de détails, consulter les références [82][83]. Suite à ce travail, d'autres groupes de recherche travaillant sur la reconnaissance de locuteur dont l'ENST fait partie se sont intéressés à ces techniques. Récemment et en parallèle à notre travail, l'équipe d'IBM a publié à Eurospeech2001 un travail intéressant sur l'adaptation des SVM en identification du locuteur [29]. Le système qu'ils ont proposé utilise les SVM comme système supplémentaire d'aide à la decision qui entre en action seulement quand le score obtenu par le système de base utilisant les GMM et LLR n'est pas fiable. Dans ce système, IBM a utilisé un nouveau noyau nommé le noyau de Fisher. Pour plus de détails sur cet intéressant travail, vous pouvez consulter la référence [30]. Dans le reste de ce chapitre nous allons décrire notre travail sur l'adaptation de la technique SVM pour la vérification de locuteur en mode indépendant du texte.

5.2 Approches hybrides GMM-SVM proposées

Dans cette section, nous allons décrire les différents systèmes que nous avons développé. Tous ces systèmes sont des systèmes hybrides GMM-SVM utilisant *des nouvelles représentations des données* basées sur une modélisation des clients par des modèles GMM. Ces nouvelles représentations des données vont nous permettre d'utiliser les SVM dans la phase de décision.

5.2.1 Protocole expérimental

Base de données

Dans nos expériences, nous avons utilisé la base de données NIST'1999 qui est une partie de la base SWITCHBOARD. Cette base est constituée de locu-

teurs qui ont enregistré chacun une session de deux minutes de parole et une dizaine de segments qui vont servir pour effectuer des tests. La durée des segments de test varie de 3 secondes à 60 secondes. Deux combinés téléphoniques (électret, carbone) ont été utilisés pour enregistrer cette base. La base est divisée en 4 ensembles complètement disjoints. Les deux premiers ensembles, pour le développement et l'évaluation, contiennent chacun 100 locuteurs appelés clients dont 50 femmes et 50 hommes. Le troisième ensemble est utilisé pour apprendre 4 modèles indépendants du locuteur et dépendants du sexe et du combiné téléphonique utilisant environ 50 locuteurs pour chaque modèle. Ces modèles sont appelés modèles du monde. Le dernier ensemble a été utilisé pour réaliser des accès pseudo-imposteurs. Cet ensemble est composé de 4 sous-ensembles d'environs 50 locuteurs chacun qui sont de même sexe et de même combiné téléphonique. Sur l'ensemble de développement, nous avons 5609 tests dont 519 sont des accès clients et 5190 sont des accès imposteurs. L'ensemble d'évaluation contient 4980 tests dont 490 sont des accès clients et 4490 sont des accès imposteurs.

Paramétrisation

Pour la paramétrisation, *le module standard Spro du consortium ELISA* est utilisé (le logiciel et le manuel d'utilisation sont dans le lien suivant :
http ://www.irisa.fr/metiss/guig/spro).
Le signal de parole est représenté toutes les 10ms par des trames de 32 composantes calculées sur des fenêtres de 20ms, 16 coefficients ceptraux plus 16 delta de coefficients calculés sur 5 trames. Une normalisation basée sur le retrait de la moyenne (Cepstral Mean Substraction) permet de minimiser les perturbations dues aux différents canaux de transmission de la voix. Enfin, un algorithme de suppression de trames basé sur une modélisation bi-gaussienne de l'énergie, est utilisé pour supprimer les trames qui correspondent au silence et qui appartiennent à la gaussienne de plus faible moyenne [58].

Modélisation

Dans nos expériences, nous avons utilisé deux algorithmes différents de modélisation MLLR et MAP. Chaque client est modélisé par 128 gaussiennes de matrices de covariances diagonales. Les modèles des clients sont obtenus par adaptation du modèle du monde du même sexe et du même combiné téléphonique. Dans la première approche que nous avons proposée, l'algorithme MLLR est utilisé pour

créer les modèles GMM en utilisant un module de modélisation de *la plate-forme ENST* développée pour les évaluations NIST de 1997 à 1999 [19][55][96]. L'algorithme MAP a été testé à partir de la deuxième approche que nous avons proposé en utilisant *le module standard de modélisation de la plate-forme ELISA* [37]. Ce module a été développé par le groupe de recherche sur la vérification du locuteur du laboratoire d'informatique de l'Université d'Avignon.

Décision

Dans cette phase, nous avons utilisé les techniques SVM en proposant des nouvelles représentations des données basées sur une modélisation GMM des clients. Pour bien évaluer nos approches, nous avons comparé les performances des nos différents systèmes avec un système de référence basé sur la technique LLR.
Le logiciel SvmFu disponible sur le site http ://svm.first.gmd.de a été utilisé pour apprendre les modèles SVM.

Description du système de référence : le système de référence que nous utilisons est un système basé sur les techniques classiques représentant l'état de l'art en VAL en mode indépendant du texte. Ainsi ce système utilise les GMM comme technique de modélisation et LLR comme technique de décision.
Notons que la différence entre le système de référence et nos systèmes hybrides GMM-SVM est le module de décision. Les précédents modules (paramétrisation et modélisation) sont les mêmes pour tous les systèmes.

5.2.2 Première approche

Comme nous l'avons déjà mentionné, cette approche est basé sur une modélisation GMM des clients. Une nouvelle représentation des données est proposée permettant l'utilisation des SVM en phase de décision.
La première étape de ce système consiste à diviser les données d'apprentissage de chaque client en deux parties. La première partie va être utilisée pour construire le modèle GMM du client et la deuxième partie sera utilisée pour construire le modèle SVM de chaque client. Les modèles GMM des clients sont construits par adaptation du modèle du monde du même sexe et du même combiné téléphonique en utilisant l'algorithme MLLR.
Pour apprendre le modèle SVM de chaque client, nous avons besoin du modèle GMM du client, du modèle du monde du même sexe et de même combiné,

quelques segments de parole du client pour construire les vecteurs SVM de la classe client et de quelques segments de parole provenant des pseudo-imposteurs pour construire les vecteurs SVM de la classe non-client.

Construction de vecteurs d'entrée des SVM

Supposons que le modèle du client et le modèle du monde ont n gaussiennes chacun, la taille des vecteurs d'entrée des SVM est de $2 * n$. Les n premières composantes vont correspondre à des mesures de vraisemblances par rapport aux n gaussiennes du modèle de client et les n dernières composantes vont correspondre à des mesures de vraisemblances par rapport aux n gaussiennes du modèle du monde. Soit X un segment de parole (client où pseudo-imposteur), λ le modèle de l'identité proclamée et $\overline{\lambda}$ le modèle du monde. Pour construire un vecteur d'entrée des SVM V_X^{λ}, nous allons construire deux vecteurs $V_X^{\lambda}(\lambda)$ et $V_X^{\lambda}(\overline{\lambda})$ de dimension n chacun dont la concaténation fournira le vecteur final. La construction des vecteurs d'entrée des SVM est réalisée comme suit :
D'abord toutes les composantes des deux vecteurs sont initialisées à 0. Ensuite pour chaque trame t du segment X, le score S_t est calculé par l'équation suivante :

$$S_t = \max_{g_i \in \lambda, \overline{\lambda}} Log[P(t|g_i)]$$

Si le score S_t est maximisé par la iéme gaussienne du modèle du client, la ième composante de notre vecteur $V_X^{\lambda}(\lambda)$ sera incrémentée par le score S_t. Si ce score est maximisé par la ième gaussienne du modèle du monde, c'est la ième composante du vecteur $V_X^{\lambda}(\overline{\lambda})$ qui sera incrémentée par le score S_t. Enfin et après avoir traité toutes les trames du segment X, les composantes des deux vecteurs $V_X^{\lambda}(\lambda)$ et $V_X^{\lambda}(\overline{\lambda})$ sont normalisées par le nombre de trames du segment X. Ainsi le vecteur d'entrée des SVM est obtenu par concaténation des deux vecteurs $V_X^{\lambda}(\lambda)$ et $V_X^{\lambda}(\overline{\lambda})$.
La figure 5.2 représente un schéma modulaire de la construction des vecteurs d'entrée pour les SVM.
Dans la phase de test, pour chaque segment de test X' et une identité β, un vecteur $V_{X'}^{\beta}$ est construit de la même façon que dans l'apprentissage. Un score de décision est obtenu par la fonction de classement des SVM suivante :

$$class(X') = \sum_{z_i \in VS(\beta)} \alpha_i^o y_i K(z_i, V_{X'}^{\beta}) + b_o \tag{5.1}$$

où z_i est un vecteur support du modèle SVM du client β, α_i^o est le coefficient de Lagrange correspondant au vecteur support z_i, y_i est la classe de z_i,$K(z_i, V_{X'}^{\beta})$

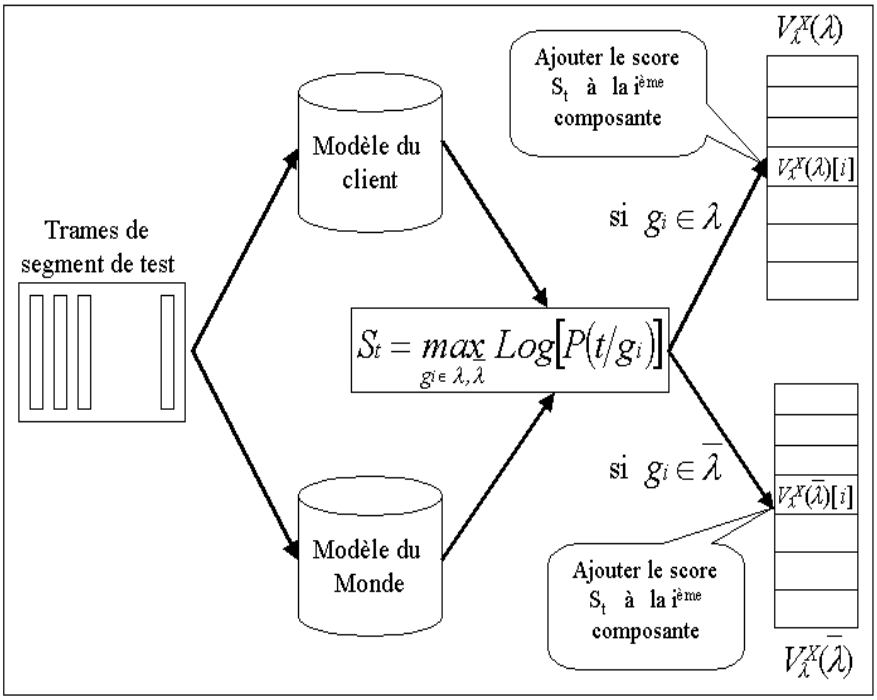


FIG. 5.2 – *Construction des vecteurs d'entrée pour les SVM de la première approche*

représente le noyau utilisé pour apprendre le modèle SVM du client β appliqué au couple $(z_i, V_{X'}^{\beta})$ et b_o est le biais du modèle SVM du client β.
Supposons que dans l'apprentissage des modèles SVM des clients, les vecteurs d'entrée représentant la classe client ont été étiquetés par 1 et les vecteurs représentant la classe non-clients ont été étiquetés par -1. Si $class(X')$ est positif alors le système décide que le segment X' est prononcé par le client β sinon le système décide que le segment X' provient d'un imposteur.
Dans cette première expérience que nous avons effectuée sur le corpus de développement des données NIST'1999, nous avons utilisé pour chaque client une des deux sessions d'une minute de parole chacune pour apprendre le modèle GMM alors que l'autre session a été divisée en 5 segments de 12 secondes chacune pour construire 5 vecteurs d'entrée représentant la classe client. La classe non-client a été représentée par 6 vecteurs construits en utilisant 6 segments de test de 10 secondes chacun provenant de 6 pseudo-imposteurs (choisis au hasard) qui sont du même sexe et qui ont utilisé le même combiné téléphonique que les données d'apprentissage du client.

Résultats et interprétations : Les résultats obtenus sont présentés sous forme de courbes DET [61]. La figure 5.3 représente les résultats obtenus par notre système hybride GMM-SVM comparés aux résultats obtenus par le système de référence utilisant la technique LLR sur le corpus de développement et sans aucune normalisation de scores.
Pour cette expérience nous avons utilisé une machine SVM linéaire. La courbe labellisée SVM-linear-s12 représente les résultats obtenus par notre système hybride GMM-SVM en utilisant la première session des données d'apprentissage pour construire les modèles GMM des clients et la deuxième session pour construire les modèles SVM. La courbe labellisée SVM-linear-s21 est obtenue en utilisant notre système hybride GMM-SVM où la deuxième session des données d'apprentissage a été utilisée pour construire les modèles GMM des clients et la première session pour construire les modèles SVM. Pour le système de référence représenté par la courbe labellisée par LLR, nous avons utilisé les deux sessions pour apprendre les modèles GMM des clients.
La figure 5.5 montre que les performances du système de référence basé sur la technique LLR sont meilleures que les performances obtenues par notre système hybride GMM-SVM. Le TEE (Taux d'Égale Erreur) est de 18.3% pour le système de référence contre 21.5% obtenu par le système GMM-SVM. Cette dif-

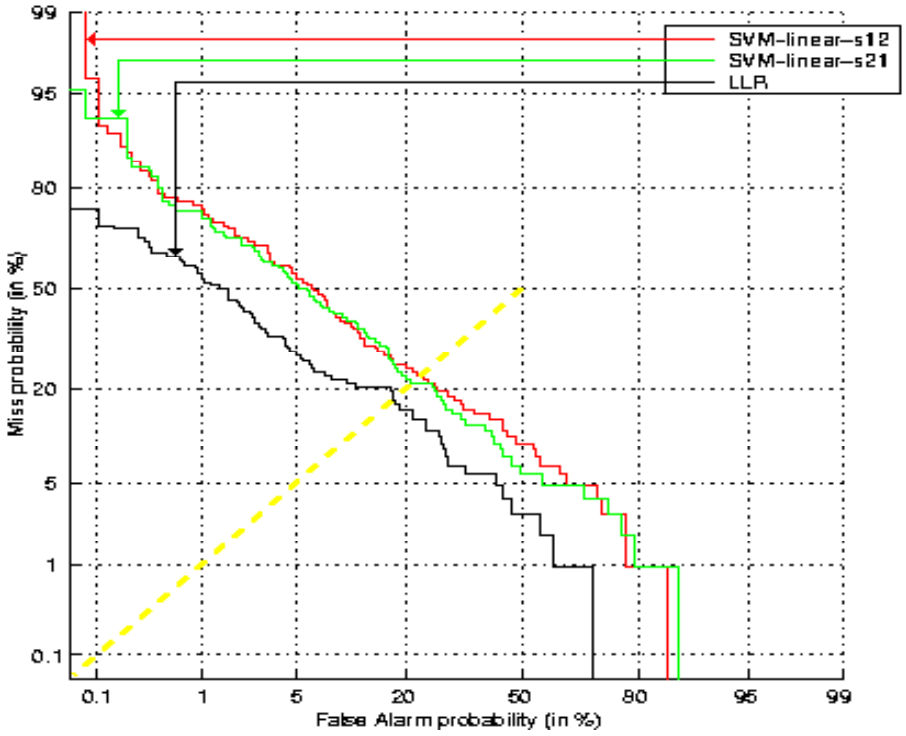


FIG. 5.3 – *Courbes DET représentant les performances de notre premier système GMM-SVM comparés au système de référence sur le corpus de développement NIST'1999. Aucune normalisation des scores n'est utilisée. Les GMM des clients sont obtenus par adaptation MLLR (les courbes sont présentées par TEE croissant).*

férence entre les performances peut être expliquée par le faite que les modèles GMM des clients utilisés dans le système GMM-SVM sont appris par la moitié des données utilisé pour apprendre les modèles GMM des clients dans le système de référence et par le petit nombre de vecteurs utilisés pour apprendre le modèle SVM de chaque client (11 vecteurs en tout dont 5 représentant la classe client et 6 représentant la classe non-client). Cette proposition a fait l'objet d'un article présenté au forum des étudiants à ICASSP'2001 [52].
Pour une première tentative d'utilisation des SVM dans un système de VAL en mode indépendant du texte, les résultats sont encourageants.
Comme nous l'avons déjà mentionné, l'inconvénient majeur de cette première approche est le très peu de données utilisées que ce soit pour l'apprentissage des modèles GMM où pour les modèles SVM. Pour surmonter ces problèmes, nous avons proposé une deuxième approche où nous construisons un seul modèle SVM plus général séparant la classe des accès clients et la classe des accès imposteurs au lieu de construire un modèle SVM pour chaque client.

5.2.3 Deuxième approche

Cette approche représente une généralisation de celle présentée dans la section précédente. Ainsi la technique de construction des vecteurs d'entrée pour les SVM reste exactement la même.
Tandis que dans la première approche nous avons construit un modèle SVM pour chaque client, dans cette approche nous proposons de construire un seul modèle SVM plus général permettant de séparer les accès clients des accès imposteurs.
Ce modèle a été construit en utilisant les accès réalisés sur l'ensemble de développement et testé sur les accès réalisés sur l'ensemble d'évaluation.
Cette nouvelle approche nous a permis de vaincre les deux inconvénients majeurs de la première approche :

- le modèle GMM de chaque client est de meilleure qualité puisque qu'il est construit en utilisant les deux sessions d'une minute chacune dédié pour l'apprentissage. Ce qui n'est pas le cas dans la première approche où nous utilisons une seule session pour apprendre le modèle du client.
- beaucoup plus de données pour apprendre le modèle SVM. Le modèle SVM a été appris en utilisant 5709 vecteurs dont 519 vecteurs représentant la classe des accès clients et 5190 vecteurs représentant la classe des accès imposteurs.

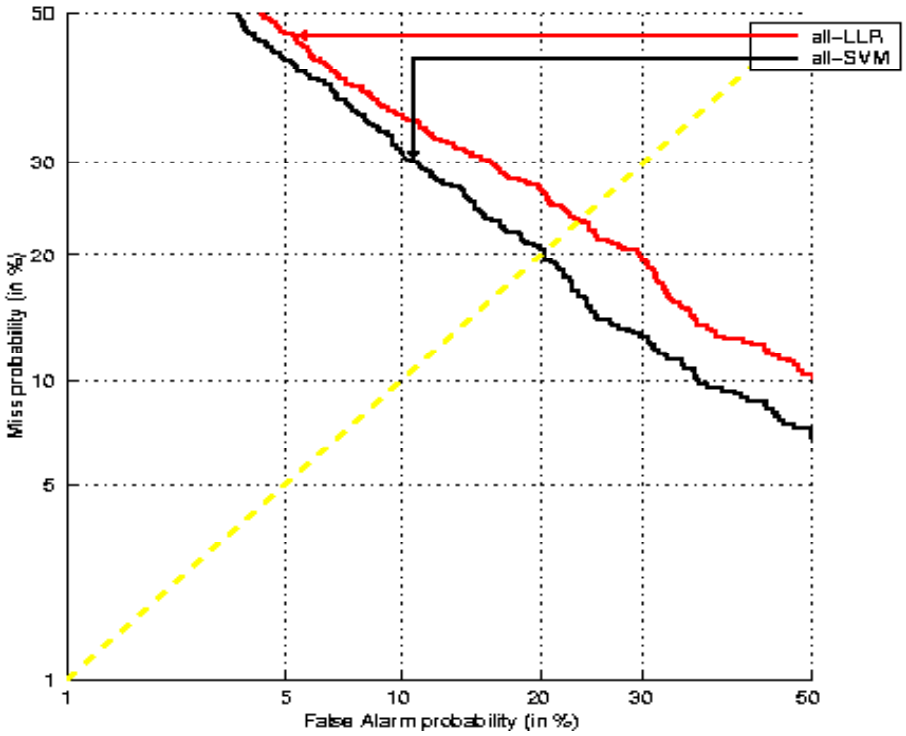


FIG. 5.4 – *Courbes DET représentant les performances de notre deuxième système GMM-SVM comparés au système de référence sur le corpus d'évaluation NIST'1999 utilisant deux sessions d'une minute chacune pour l'apprentissage. Aucune normalisation des scores n'est utilisée. Les GMM des clients sont obtenus par adaptation MLLR (les courbes sont présentées par TEE croissant).*

Résultats et interprétations : La figure 5.4 représente les résultats obtenus par cette approche sous forme de courbe DET [61].

La figure 5.4 montre que le système hybride a de meilleures performances que le système de référence basé sur la technique LLR. Ce qui confirme notre analyse des résultats obtenus par la première approche.

Suite à ces résultats, nous avons décidé de participer aux évaluations NIST'2001 avec le système que nous avons mis en place dans cette deuxième approche.

Sachant que dans les évaluations NIST'2001 nous disposons d'une session de deux minutes pour apprendre les modèles des clients, nous avons décidé de refaire la même expérience décrite précédemment sur les données de NIST'1999 en changeant les deux sessions d'une minute chacune par une seule session de deux minutes afin de se mettre dans les même conditions que les évaluations NIST'2001.

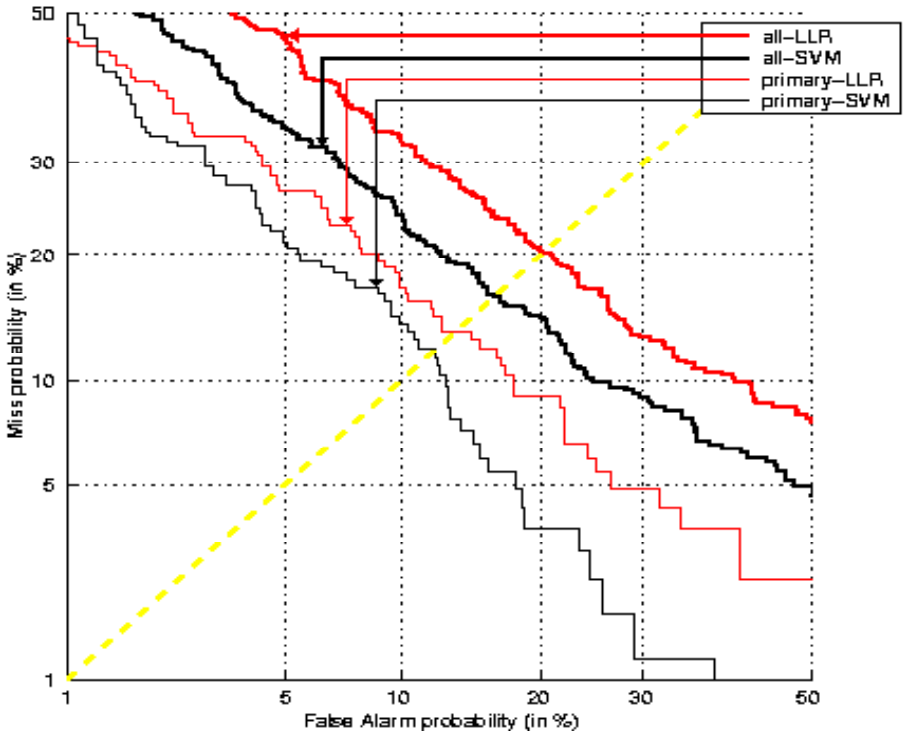


FIG. 5.5 – *Courbes DET représentant les performances de notre deuxième système GMM-SVM comparées au système de référence sur le corpus d'évaluation NIST'1999 utilisant une session de deux minutes pour l'apprentissage. Aucune normalisation des scores n'est utilisée. Les GMM des clients sont obtenus par adaptation MLLR (les courbes sont présentées par TEE croissant).*

Ces données ont été fournis par NIST comme données de développement pour la compagne 2001. Les résultats sont présentés dans la figures 5.5

Les résultats présentés dans la figure 5.5 concernent deux conditions "all" et "primary". La condition "all" prend en compte les décisions prises sur tous les tests réalisés, tandis que la condition "primary" ne prend en compte que les décisions prises sur les tests utilisant des segments d'une durée qui varie entre 15 secondes et 45 secondes et qui sont enregistrés avec le même combiné téléphonique "électret" et où le client a également utilisé le combiné "électret" pour enregistrer ses données d'apprentissage. On peut constater que les performances de notre système hybride GMM-SVM sont largement meilleures que ceux du système du référence sur les deux conditions "all" et "primary" et sans normalisation.

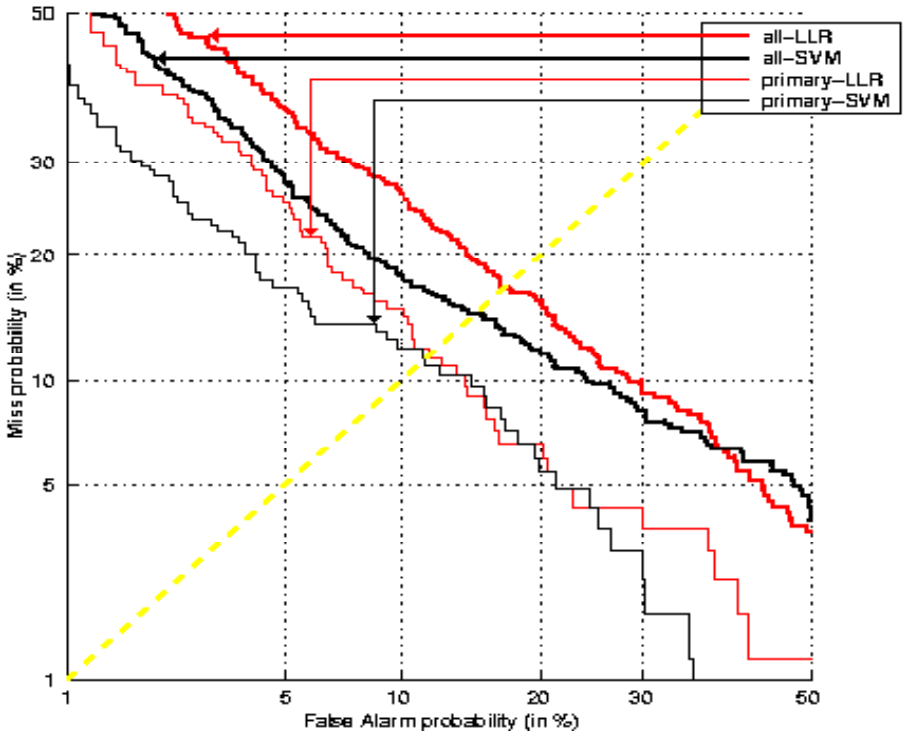


FIG. 5.6 – *Courbes DET représentant les performances avec la normalisation Hnorm du deuxième système GMM-SVM comparées au système de référence sur le corpus d'évaluation NIST'1999 utilisant une session de deux minutes pour l'apprentissage. La normalisation Hnorm est utilisée. Les GMM des clients sont obtenus par adaptation MLLR (les courbes sont présentées par TEE croissant).*

Pour améliorer les performances de notre système pour ces évaluations, nous avons décidé d'appliquer la technique de normalisation de scores Hnorm [43]. Les résultats obtenus sur l'ensemble l'ensemble d'évaluation NIST'1999 sont présentés sous forme de courbe DET dans la figure 5.6.

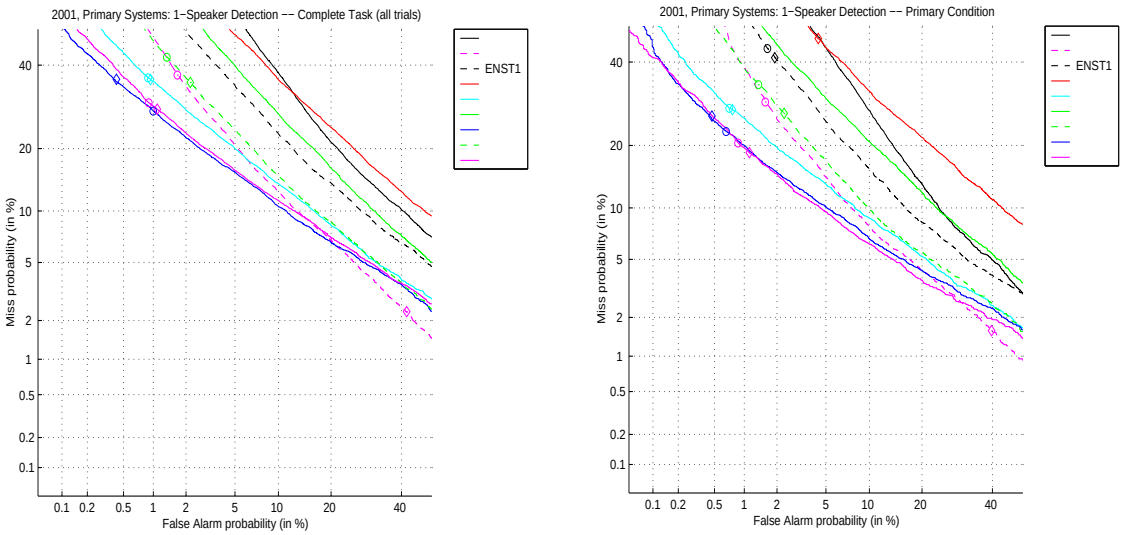
La figure 5.6 montre que les performances de notre système hybride GMM-SVM restent toujours meilleures que celles du système de référence. Les résultats montrent également une amélioration de performances des deux systèmes. Sur la condition "all", on peut noter une amélioration de ∼3%. Par contre sur la condition "primary" l'amélioration est de ∼2.5% pour le système de référence et de ∼1.5% pour notre système hybride GMM-SVM.

On peut dire que l'application des techniques de normalisation de scores a la même influence puisqu'une amélioration équivalente des performances est consta-

tée sur les deux systèmes (système de référence et le système hybride GMM-SVM). C'est avec ce système que nous avons participé aux évaluations NIST'2001.

Description de la base de données NIST'2001 : La base de données NIST'2001 est composée de 1003 locuteurs dont 546 femmes et 557 hommes. Chaque locuteur a enregistré une session de deux minutes de parole dédiée pour l'apprentissage et une dizaine de segments d'une durée variant entre 3s et 45s dédiés pour le test. Le nombre de tests à réaliser s'élève à 94517 tests dont 45862 accès femmes et 48655 accès hommes.

Les courbes DET présentées dans la figure 5.7 sont les résultats officiels de cette participation pour les deux conditions "all" et "primary" sachant que les systèmes participants ne sont évalués que par leur performances sur la condition "primary". Tenant compte que c'est notre première participation aux évaluations NIST avec ce nouveau système hybride GMM-SVM, on peut dire que notre participation est assez satisfaisante et que les résultats sont très encourageants.



Condition "all" Condition "primary"

FIG. 5.7 – *Courbes DET obtenus par les différents systèmes participants aux évaluations NIST'2001*

Au cours de la compagne d'évaluation NIST'2001, nous avons remarqué que la plupart des systèmes utilisent l'algorithme MAP pour apprendre les modèles GMM. Sachant aussi que plusieurs travaux ont montré qu'une modélisation utilisant l'algorithme MAP est plus performante qu'une modélisation utilisant l'algorithme MLLR [37], nous avons décidé de mener une expérience sur le corpus d'évaluation NIST'1999 en changeant l'algorithme MLLR par l'algorithme MAP pour apprendre les modèles GMM.
Les résultats obtenus sans aucune normalisation des scores sont présentés dans le tableau suivant :

	primary	All
Sys de réf (LLR)	10.5% [9.7, 11.3]	19.5% [18.4, 20.6]
GMM-SVM (RBF)	11% [10.2, 11.8]	21% [19.9, 22.1]
GMM-SVM (Polynomial d = 2)	12% [11.1, 12.9]	20% [18.9, 21.1]
GMM-SVM (linéaire)	12.5% [11.6, 13.4]	21.5% [20.4, 22.6]

Tableau 5.1 : Les Taux d'Égale Erreur et leurs intervalles de confiance (entre "[]") obtenus par le système de référence et notre système hybride GMM-SVM avec différents noyaux sur les données d'évaluation NIST'1999. Aucune normalisation des scores n'est utilisée. Les GMM des clients sont obtenus par adaptation MAP.

Comme nous l'avons prévu, L'algorithme MAP a amélioré les résultats du système de référence basé sur la technique LLR. Une amélioration des TEE de 3% est obtenue sur la condition "primary" et de 0.5% sur la condition "all". Par contre, sur notre système hybride GMM-SVM (au noyau linéaire puisque c'est le seul noyau que nous avons utilisé dans nos précédentes expériences) nous avons relevé une baisse de performances sur la condition "all" où le TEE est passé de 17.5% en utilisant l'algorithme MLLR à 21.5% en utilisant l'algorithme MAP alors que sur la condition "primary", nous avons obtenu un TEE égale à 12.5% avec les deux algorithmes (MLLR et MAP).
Sachant que la différence principale entre la condition "all" et la condition "primary" est le 10% de tests ajoutés à la condition "all" où les segments de test ont été enregistrés avec un combiné téléphonique différent de celui utilisé pour enregistrer les données d'apprentissage, on peut déduire que notre système est très fragile par rapport au changement d'environnement (le combiné téléphonique dans le cas présent).

Les modèles GMM obtenus par l'algorithme MAP sont de meilleure qualité que ceux obtenus par l'algorithme MLLR (une amélioration très nette est constatée sur le système de référence utilisant la technique LLR). Ceci dit, la chute de performances de notre système hybride ne peut provenir que de la représentation des données que nous utilisons. En effet, dans cette représentation des données, le vecteur d'entrée au SVM est obtenu par concaténation des deux vecteurs intermédiaires où le premier représente les scores maximisant sur le modèle du client et le deuxième vecteur correspond aux scores maximisant sur le modèle du monde. Cette représentation est sans doute discriminante mais ne présente aucune normalisation directe par rapport au modèle du monde pour éliminer l'influence de l'environnement comme c'est le cas dans la technique LLR.
Pour résoudre ce problème, nous avons proposé une deuxième forme de représentation de données qui permet de mettre en place une normalisation par le modèle du monde. La construction des vecteurs d'entrée correspondant à cette nouvelle représentation est basée sur la construction des vecteurs intermédiaires que nous avons présentée précédemment (figure 5.2). Si dans la première représentation des données le vecteur d'entrée est obtenu par concaténation des deux vecteurs intermédiaires, dans cette nouvelle représentation de données le vecteur d'entrée des SVM est obtenu en calculant le rapport terme à terme du premier vecteur correspondant aux scores client et le deuxième vecteur représentant les scores obtenus par le modèle du monde. L'idée de cette normalisation terme par terme est le fait que chaque gaussienne g_i du modèle de client est adaptée à partir de la gaussienne gi du modèle du monde. La figure 5.8 peut être considérée comme une continuité de la figure 5.3 qui présente cette nouvelle représentation des données. Les résultats obtenus suite à l'utilisation de cette normalisation par le modèle du monde sont présentés dans la figure 5.9. La figure 5.9 montre une nette amélioration des performances de notre système hybride GMM-SVM. Le TEE est passé de 21.5% à 19.5% sur la condition "all" et de 12.5% à 11% sur la condition "primary". Bien que l'application de cette normalisation a amélioré les résultats de notre système hybride, les résultats obtenus par le système de référence restent meilleurs surtout sur la condition "primary" où le TEE est de 10.5% pour le système de référence et de 11% pour notre système hybride.
En analysant la normalisation que nous avons appliqué, nous avons constaté qu'elle est très générale et qu'il y a possibilité de la rendre beaucoup plus significative. C'est pourquoi nous avons proposé une nouvelle représentation des données qui permet d'utiliser une normalisation par modèle du monde au niveau

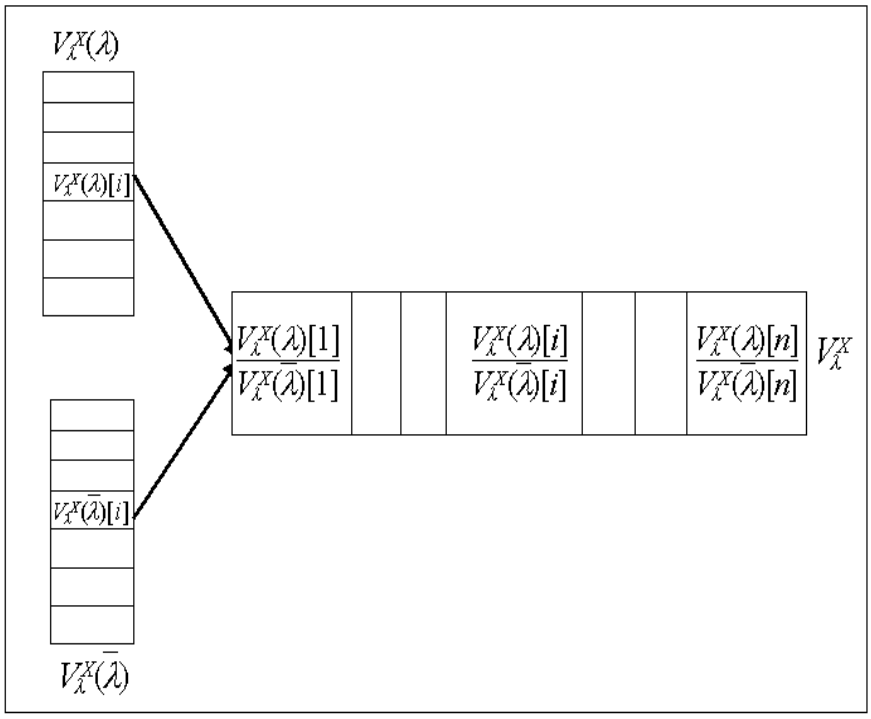


FIG. 5.8 – *Construction des vecteurs d'entrée pour les SVM utilisant une normalisation par le modèle du monde*

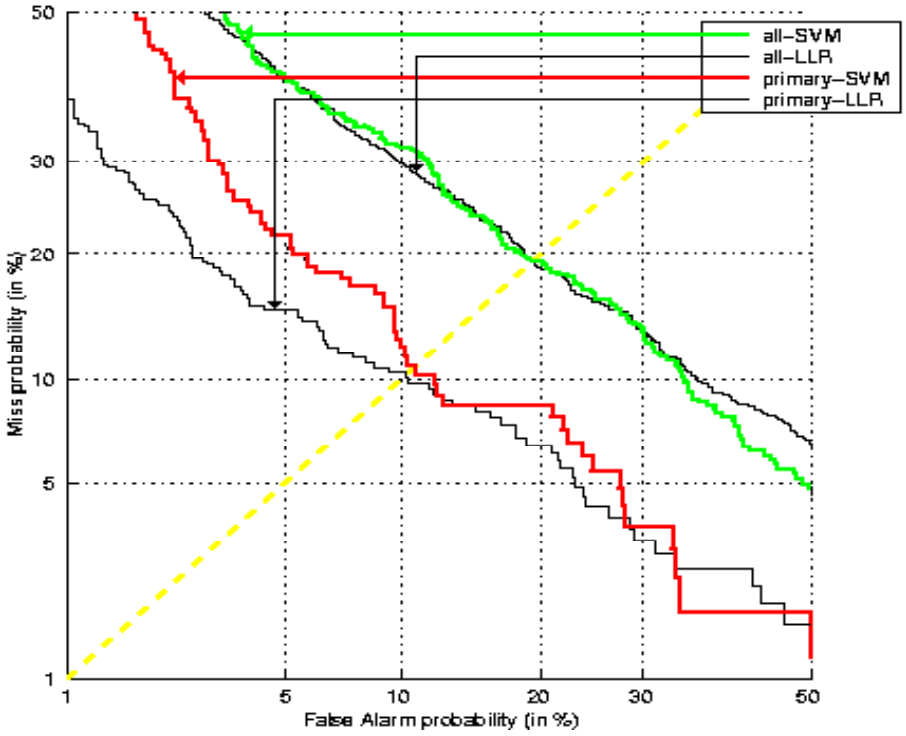


FIG. 5.9 – *Courbes DET du système GMM-SVM utilisant la nouvelle représentation des données comparés au système LLR obtenues sur le corpus d'évaluation NIST'1999. Aucune normalisation des scores n'est utilisée. Les GMM des clients sont obtenus par adaptation MAP (les courbes sont présentées par TEE croissant).*

du calcul du score de chaque trame. Cette proposition fait l'objet de la troisième approche que nous décrivons en détail dans la section suivante.

5.2.4 Troisième approche

Dans le dernier système que nous proposons, nous avons utilisé une nouvelle représentation des données qui permet d'appliquer la normalisation par le modèle du monde au calcul du score de chaque trame.

Construction des vecteurs d'entrée des SVM

Pour décrire cette nouvelle représentation des données, soit λ le modèle du client et $\overline{\lambda}$ le modèle du monde de n gaussiennes chacun et soit X un segment de parole. La construction du vecteur d'entrée pour les SVM V_X^λ de dimension n correspondant à cet accès est réalisé de la manière suivante : d'abord on initialise toutes les composantes de notre vecteur V_X^λ par zéro. Ensuite pour chaque trame t du segment X, on détermine l'indice i de la gaussienne $g_j \in \lambda$, $\overline{\lambda}$ qui maximise la probabilité $P(t|g_j)$ suivant l'équation :

$$i = \arg \max_{g_j \in \lambda, \overline{\lambda}} P(t|g_j)$$

Un score S_t est alors calculé par l'équation suivante :

$$S_t = Log[\frac{P(t|g_i^\lambda)}{P(t|g_i^{\overline{\lambda}})}]$$

Ce score S_t est ensuite ajouté à la iéme composante du vecteur V_X^λ. Enfin et après avoir traité toutes les trames du segment X, les composantes de notre vecteur V_X^λ sont normalisées par le nombre de trames du segment X.
La figure 5.10 représente un schéma modulaire de ce système. La première expérience que nous avons menée était sur les données NIST'1999, données de développement de l'évaluation NIST'2001.
Nous avons testé plusieurs machines pour apprendre le modèle SVM. Le noyau linéaire a été testé avec différentes valeurs du paramètre de régularisation ainsi que le noyau RBF avec différentes valeurs de la variance σ^2. Nous avons testé également le noyau polynômial avec différents degrés, mais les résultats n'étaient pas intéressants pour les mettre dans ce rapport. La seule conclusion qu'on a pu en tirer est que le noyau polynômial n'est pas adapté à notre application.

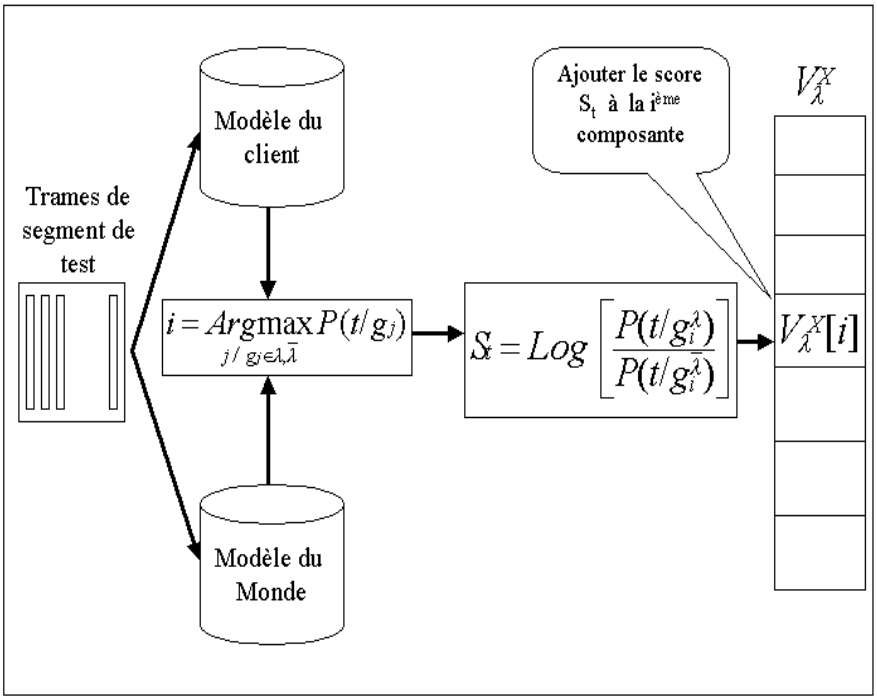


FIG. 5.10 – *Nouvelle représentation des données de la troisième approche*

Résultats et interprétations : Les résultats sont présentés dans le tableau suivant :

	primary	All
Sys de réf (LLR)	10.5% [9.7, 11.3]	19.5% [18.4, 20.6]
GMM-SVM (RBF $\frac{1}{2\sigma^2} = 50$)	**8.5% [7.7, 9.2]**	19.25% [18.1, 20.3]
GMM-SVM (RBF $\frac{1}{2\sigma^2} = 40$)	**8.5% [7.7, 9.2]**	19.5% [18.4, 20.6]
GMM-SVM (RBF $\frac{1}{2\sigma^2} = 30$)	9% [8.2, 9.8]	19.5% [18.4, 20.6]
GMM-SVM (RBF $\frac{1}{2\sigma^2} = 20$)	9% [8.2, 9.8]	19.5% [18.4, 20.6]
GMM-SVM (RBF $\frac{1}{2\sigma^2} = 10$)	9.25% [8.4, 10]	20% [18.9, 21.1]
GMM-SVM (linéaire C = 0.01)	9% [8.2, 9.8]	19.75% [18.6, 20.8]
GMM-SVM (linéaire C = 0.1)	9.25% [8.4, 10]	19.75% [18.6, 20.8]
GMM-SVM (linéaire C = 1)	9.25% [8.4, 10]	19.75% [18.6, 20.8]

Tableau 5.2 : Les Taux d'Égale Erreur et leurs intervalles de confiance (entre "[]") obtenus par le système de référence et notre système hybride GMM-SVM avec différents noyaux sur le corpus d'évaluation NIST'1999. Aucune normalisation des scores n'est utilisée. Les GMM des clients sont obtenus par adaptation MAP.

Le tableau 5.2 montre que les meilleures résultats sont obtenus par le système GMM-SVM utilisant le noyau RBF avec $\sigma^2 = 0.01$ ($\frac{1}{2\sigma^2} = 50$) sans aucun recouvrement des intervalles de confiance sur la condition "primary" et un fort recouvrement sur la condition "all" . On constate aussi que les systèmes GMM-SVM ont obtenus un TEE sur la condition "primary" meilleure que le TEE obtenu par le système de référence basé sur la technique LLR. Par contre, sur la condition all, il n'y a que le système GMM-SVM utilisant un noyau RBF avec $\sigma^2 = 0.01$ qui a obtenu un TEE légèrement meilleur que celui du système LLR, 19.25% contre 19.5%.
Pour bien évaluer cette approche que nous venons de décrire, nous avons décidé de lancer notre système sur toutes les données NIST'2001. Les résultats obtenus sont présentés dans le tableau suivant :

	primary	All
Sys de réf (LLR)	11.5% [11.3, 11.7]	16% [15.77, 16.23]
GMM-SVM (RBF $\frac{1}{2\sigma^2} = 50$)	**10.75% [10.55, 10.95]**	**15% [14.78, 15.22]**
GMM-SVM (RBF $\frac{1}{2\sigma^2} = 40$)	11.25% [11.04, 11.46]	16% [15.77, 16.23]
GMM-SVM (RBF $\frac{1}{2\sigma^2} = 30$)	11.75%[11.53, 11.97]	16.25% [16.02, 16.48]
GMM-SVM (RBF $\frac{1}{2\sigma^2} = 20$)	12% [11.88, 12.22]	17% [16.85, 17.25]
GMM-SVM (RBF $\frac{1}{2\sigma^2} = 10$)	12.5% [12.27, 12.73]	17.5% [17.22, 17.78]
GMM-SVM (linéaire C = 0.01)	11.5% [11.3, 11.7]	16.25% [16.02, 16.48]
GMM-SVM (linéaire C = 0.1)	12.5% [12.27, 12.73]	17% [16.85, 17.25]
GMM-SVM (linéaire C = 1)	13.5% [13.2, 13.8]	17.5% [17.22, 17.78]

Tableau 5.3 : Les Taux d'Égale Erreur et leurs intervalles de confiance (entre "[]") obtenus par le système de référence et notre système hybride GMM-SVM avec différents noyaux sur le corpus d'évaluation NIST'2001. Aucune normalisation des scores n'est utilisée. Les GMM des clients sont obtenus par adaptation MAP.

Les résultats présentés dans ce tableau confirment les conclusions de l'expérience menée sur les données d'évaluation NIST'1999 dans le sens où les meilleures performances reviennent toujours au système GMM-SVM utilisant le noyau RBF avec $\sigma^2 = 0.01$ sans aucun recouvrement des intervalles de confiance sur les deux conditions "all" et "primary". Mais on a noté une légère baisse des performances des autres systèmes GMM-SVM surtout sur la condition "primary" où on s'attendait à avoir des performances meilleures que le système de référence. La figure 5.11 représente les courbes DET obtenus par le système GMM-SVM avec le noyau RBF avec $\sigma^2 = 0.01$ et le système de référence LLR.

Pour des raisons de temps de calcul, tous les résultats que nous avons présentés sont obtenus sans l'utilisation d'une normalisation des scores du genre Znorm, Hnorm ou Tnorm [43][78].

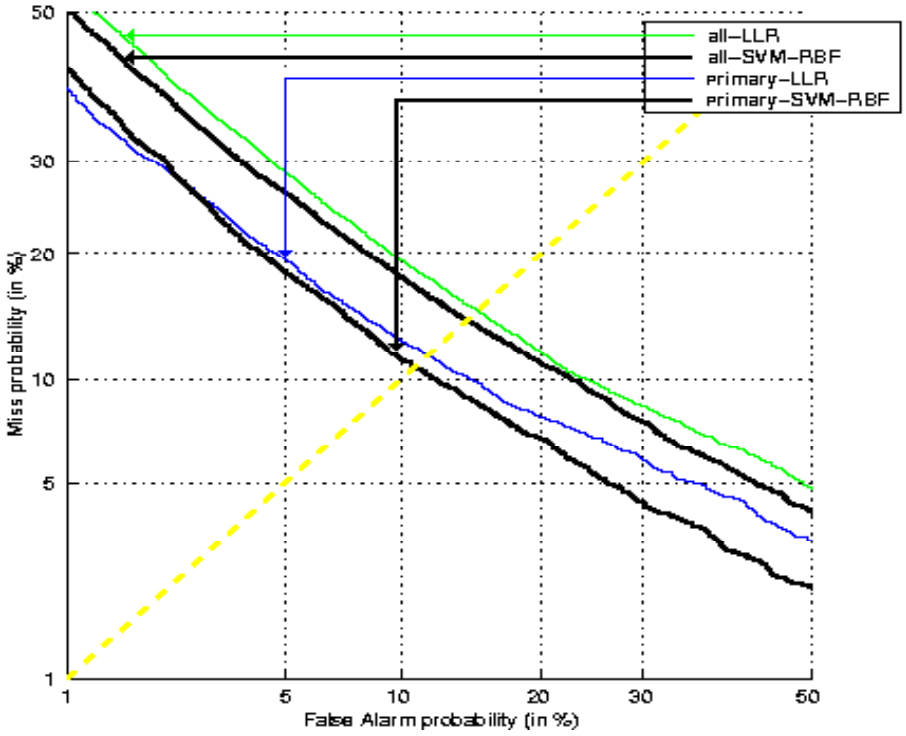


FIG. 5.11 – *Courbes DET du meilleurs systèmes GMM-SVM utilisant le noyau RBF avec $\sigma^2 = 0.01$ comparé au système de référence LLR obtenus sur les données NIST'2001. Aucune normalisation des scores n'est utilisée. Les GMM des clients sont obtenus par adaptation MAP (les courbes sont présentées par TEE croissant).*

5.3 Bilan

Dans ce chapitre nous avons présenté trois approches différentes utilisant les SVM pour la vérification du locuteur. Dans ces approches, nous avons proposé trois *nouvelles représentations des données* basées sur la modélisation GMM des locuteurs. Ces nouvelles représentations des données nous ont permis de mettre en place des systèmes hybrides GMM-SVM réunissant l'efficacité des GMM en modélisation et la performance des SVM en classement.
Dans la première approche, une première représentation des données a été présenté permettant de construire un modèle SVM pour chaque client permettant de le discriminer par rapport à un ensemble de locuteur, représentant les imposteurs. Cette représentation consiste à construire des vecteurs d'entrée pour les SVM en concaténant deux vecteurs, le premier représente les scores des trames maximisant sur le modèle du client et le deuxième représente les scores maximisant sur le modèle du monde. Les résultats obtenus sont très encourageants pour un premier système utilisant les SVM en VAL bien que les performances du système de référence utilisant la technique LLR soient meilleures.
L'analyse des résultats obtenus par ce premier système nous a permis de proposer une deuxième approche qui représente une généralisation de la première. Dans cette deuxième approche, un seul modèle SVM a été construit séparant la classe des accès clients de la classe des accès imposteurs en utilisant la même représentation des données proposée dans la première approche. Les résultats obtenus sont très intéressants. Les performances de notre système hybride GMM-SVM sont meilleures que celles du système de référence.
Une étude sur l'influence de la normalisation de scores a été réalisée sur cette approche. Cette étude a montré que notre système hybride a un comportement similaire que celui du système de référence en marquant une amélioration équivalente de l'ordre de $\sim 2\%$ suite à l'application de la technique Hnorm.
Une autre étude concernant les algorithmes d'adaptation a été menée en comparant des deux algorithmes MLLR et MAP. Les résultats obtenus montrent que l'adaptation MAP est plus intéressante (puisqu'elle améliore les performances du système utilisant MLLR), mais moins robuste au changement d'environnement que le système classique basé sur la technique LLR. Cette fragilité envers le changement d'environnement est dû à la représentation des données utilisée et qui ne présente aucune normalisation des scores par rapport au modèle du monde qui est une technique classique pour éliminer l'influence d'environnement dans le

système de VAL. Pour surmonter ce problème, nous avons proposé une nouvelle représentation des données. Dans celle-ci, au lieu de concaténer le vecteur représentant le client et le vecteur représentant le monde comme c'est le cas dans la première représentation des données proposée, nous avons calculé un vecteur en normalisant le vecteur du représentant le client par celui représentant le monde. Certes, cette normalisation a amélioré les résultats, mais cette amélioration n'est pas suffisante en comparant au système de référence.
Ainsi, une troisième approche a été proposée utilisant une troisième représentation des données. Cette représentation des données permet d'introduire la normalisation par le modèle du monde au niveau du calcul du score de chaque trame. Les résultats obtenus sur l'ensemble d'évaluation NIST'1999 sont très intéressants. Pour valider ces résultats nous avons fait tourner notre système sur la base de données NIST'2001 qui contient beaucoup plus de données où nous avons réalisé 94517 tests. Les résultats que nous avons obtenus confirment ceux obtenus sur l'ensemble de d'évaluation NIST'1999. Le système hybride que nous avons proposé a de meilleures performances que le système de référence basé sur la technique LLR considérée comme état de l'art en VAL.
A cause du temps considérable que prend chaque expérience et qui est dû aux grandes tailles des bases de données utilisées, nous nous sommes limités à tester les noyaux les plus répandus : le noyau linéaire, le noyau polynômial et le noyau RBF avec différents paramètres.

Chapitre 6

SVM pour la fusion des données

La fusion est une procédure qui permet d'intégrer plusieurs informations provenant de (plusieurs) sources différentes dans un même processus de classement. On peut distinguer trois types de fusion :

- Fusion d'éléments : elle consiste à fusionner des informations suite à des traitements partiels de la même entité.
- Fusion de méthodes : elle permet de combiner des informations provenant de différents traitement sur la même entité.
- Fusion de modes : elle consiste à fusionner des informations obtenues par traitement de différentes entités

6.1 État de l'art

le problème de la fusion des données est souvent présenté sous la forme suivante :
Soit N experts distincts et V le vecteur des scores obtenus par les différents experts :

$$V = \begin{bmatrix} v_1 \\ v_2 \\ . \\ . \\ \\ v_N \end{bmatrix}$$

où v_i est le score obtenu par le ième expert.
A partir du vecteur V, le système de fusion appelé aussi superviseur a la tâche de

prendre la decision finale qui consiste à classer ce vecteur (ex : accepter/rejeter un locuteur dans le cas de la VAL ou individu dans le cas de la vérification multimodale de l'identité).
Dans la littérature, on distingue deux grandes familles de superviseurs qui dépendent principalement de la nature des informations transmises par les différents experts.
Dans la première famille, chaque expert remet au système de fusion une décision plutôt qu'un score. Les décisions sont présentées sous forme de valeurs binaires ($v_i \in \{0,1\}$ où 1 dénote l'acceptation et 0 dénote le rejet). Ces données binaires sont traitées par le système de fusion, qui a une vue de l'ensemble des différents opinions, pour prendre la décision finale. A cause du type (binaire) des données transmises par les experts, la décision s'effectue en générale par l'intermédiaire des opérateurs logiques (ex : ET/OU). Si le nombre d'expert le permet, d'autres techniques de fusion peuvent être utilisées (ex : le choix de la décision majoritaire). De tel schéma de fusion est appelé *fusion de décisions* ou encore *fusion dure*, en regard des données "brutes" binaires traitées [71][93].
Dans la deuxième famille, les systèmes de fusion font usage direct des scores obtenus par les différents experts et les combine pour prendre la décision finale. Cette combinaison peut être faite par n'importe quelle fonction des scores issus des experts : de la plus simple combinaison linéaire (ex : le calcul de la moyenne arithmétique) jusqu'aux règles de la logique floue en passant par des techniques de décision optimisant un critère statistique particulier, tel le critère de Bayes. D'autres techniques de classement peuvent être utilisées comme les KPV et RN. Les termes *fusion de scores* ou *fusion douce* sont souvent utilisés pour désigner les systèmes de fusion appartenant à cette deuxième famille [71] [93].

Dans le reste de ce chapitre, nous allons décrire nos expériences utilisant les techniques de classement SVM pour des tâches de fusion de scores. La première expérience consiste à fusionner les scores obtenus par les différents participants à la tâche "one speaker detection" aux évaluations NIST'2001. C'est une fusion de méthodes. La deuxième expérience consiste à fusionner les scores sur les données M2VTS[1] suite au traitement de 4 modalités différentes (image de face, image profil contour, image profile luminance et la parole). Pour plus de détails sur les différentes modalités de la base M2VTS, les lecteurs peuvent consulter la thèse de Stéphane Pigeon [71] et la thèse de P. Verlinde [93]

[1]Multi Modal Verification for Teleservices and Security applications

Ce travail entre dans le cadre du projet BIOMET sur la vérification biométrique multimodale de l'identité.

6.2 Projet BIOMET

L'objectif de BIOMET est de mettre en synergie les compétences des équipes des Écoles du GET (Groupe des Écoles de Télécommunications) impliquées dans le domaine de la vérification et de l'authentification pour accéder à un système sécurisé au moyen de diverses modalités biométriques : vérification de signatures, analyse du visage, des empreintes digitales et de la forme de la main, authentification du locuteur, ainsi que leur fusion pour la vérification multimodale de l'identité. Les différentes tâches de ce projet sont :

- Créer une base de données multimodale, impératif préalable à tout travail en commun sur la fusion des différentes modalités mentionnées ;
- Mettre au point des systèmes de vérification opérationnels pour chacune des modalités ;
- Chercher une ou plusieurs stratégies de fusion adéquates, et les tester sur cette plate-forme.

Le travail que nous avons réalisé dans cette thèse concerne la troisième tâche en menant des expériences de fusion utilisant la technique SVM.

6.3 Fusion des scores NIST'2001

Dans ces expériences, nous avons demandé à NIST (National International Standard of technologies) de nous fournir les scores de tous les participants à la tâche "One Speaker detection" aux évaluations NIST'2001 pour pouvoir tourner des expériences de fusion sur ces données. Après l'accord de tous les participants, nous avons reçu les résultats de 10 systèmes différents sur lesquels nous avons fait tourner nos expériences.

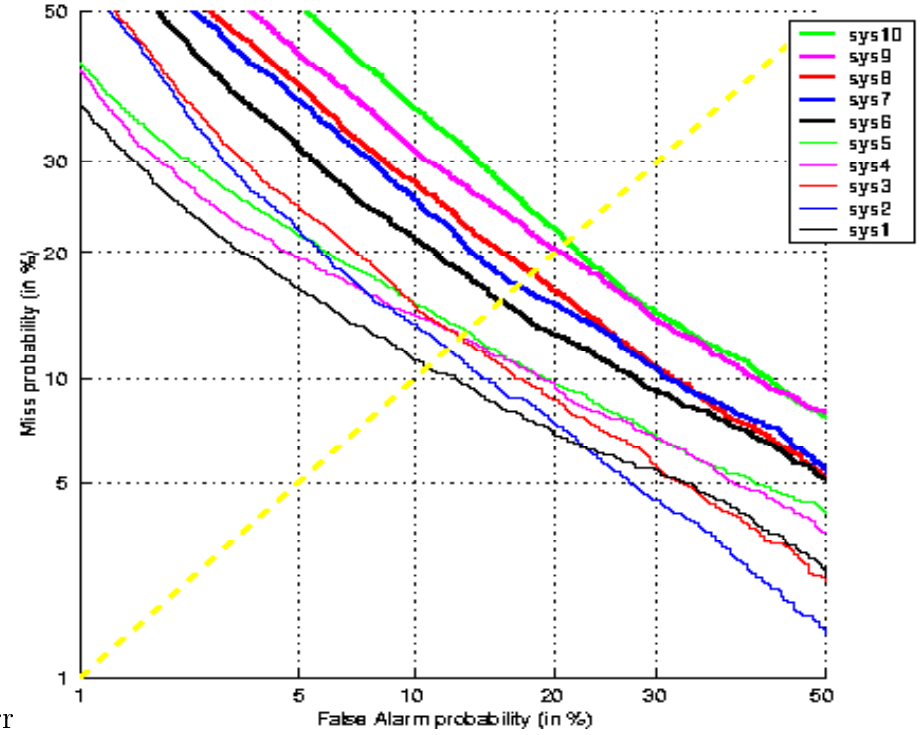
6.3.1 Protocole expérimental

Dans les évaluations NIST'2001, tous les participants à la tâche "one speaker detection" ont pu faire tourner leur systèmes sur un sous ensemble du corpus SWITCHBOARD suivant un protocole bien défini par NIST. Ce sous ensemble

est constitué de 1003 locuteurs (546 femmes et 557 hommes). Le nombre de tests réalisé est de 94517 tests dont 45862 accès femmes (3010 accès clients et 42852 accès imposteurs) et 48655 accès hommes (2817 accès clients et 45838 accès imposteurs).
Pour réaliser ces expériences de fusion, nous avons mis en place un protocole expérimental qui consiste à utiliser l'ensemble des accès femmes en apprentissage et l'ensemble des accès hommes pour le test. Ainsi les modèles SVM séparant la classe des accès clients de la classe des accès imposteurs ont été appris sur les accès femmes et testés sur les accès hommes. Le choix d'utiliser l'ensemble des accès femmes pour l'apprentissage est basé sur le fait que cet ensemble contient moins de données (moins de temps de calcul pour apprendre les modèles SVM) et surtout parce que le rapport des accès clients sur accès imposteurs est plus important dans l'ensemble des accès femmes que dans l'ensemble des accès hommes. Pour apprendre ces modèles SVM, tous les accès femmes ont été étiquetés à 1 si c'est un accès client et à -1 c'est un accès imposteur. Les vecteurs utilisés sont de dimension 10 dont chaque composante correspond au score obtenu par un système bien défini. Dans cette expérience, nous avons testé trois noyaux différents : linéaire, polynomial et RBF avec différents paramètres (cf. chapitre 2).
Pour évaluer nos systèmes de fusion utilisant les techniques SVM, nous les comparons avec un système de référence représentant le système de fusion le plus simple et qui consiste à calculer le score moyen de scores obtenus par les différents systèmes.

6.3.2 Résultats

Les résultats sont présentés sous formes de courbes DET [61]. La figure 6.1 représente les performances des différents systèmes utilisés dans cette expérience de fusion sur les accès hommes seulement. Les performances de ces systèmes en terme de TEE (Taux d'Égale Erreur) varient de 12% à 21%. La figure 6.2 représente les courbes DET obtenues par le système de référence, et les trois systèmes de fusion utilisant les techniques SVM avec différents noyaux. Ces résultats montrent que les systèmes de fusion utilisant les techniques SVM avec les noyaux (linéaire, polynomial de degré 2) sont largement meilleurs que les résultats obtenus par le système de référence, par contre de très faibles résultats sont obtenus par le système de fusion SVM avec le noyau RBF. Le système de fusion de référence marque une amélioration du TEE de 2% tandis qu'une amélioration de 3% et 4%



rr

FIG. 6.1 – *Courbes DET de tous les systèmes participants sur les accès hommes de l'évaluation NIST'2001 (les courbes sont numérotées par TEE croissant).*

est obtenue respectivement par la fusion SVM au noyau polynomial de degré 2 et par le noyau linéaire.
Étant donné que ce travail est purement expérimental, nous n'avons aucune explication concernant ces performances étranges obtenues par le système SVM avec le noyau RBF. Le seul commentaire qu'on peut donner suite à ces résultats est que le noyau RBF n'est pas adapté à cette application.
Notons qu'aucune technique de normalisation des scores à fusionner ni aucune technique de normalisation de score final n'a été utilisée.
D'une manière générale, ces expériences prouvent l'efficacité des SVM en tant que technique de fusion. Ce qui rejoint la conclusion de P. Verlinde qui a mené des expériences similaires sur les scores obtenus par les participants à la tâche "One Speaker Detection" aux évaluations NIST'1999 [93]. Dans ces expériences, P. Verlinde a testé la fusion des scores en utilisant un classifieur bayésien basé sur une technique de regression logistique. Les résultats obtenus dans ces expériences montrent une amélioration du TEE de 3% par la fusion de tous les systèmes utilisés. Pour plus de détails sur ce travail, le lecteur peut se référer au [91]

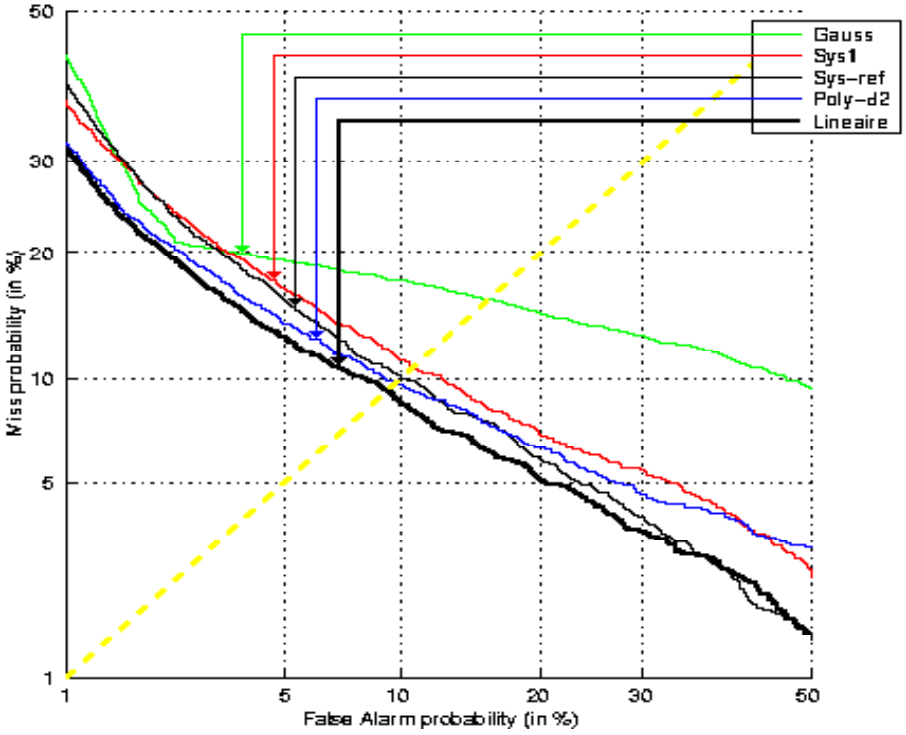
6.4 Fusion des scores M2VTS

6.4.1 Les différentes modalités de la base M2VTS

Dans le cadre de projet M2VTS, l'authentification multimodale est réalisée suivant quatre modalités différentes. La figure 6.3 empruntée à S. Pigeon représente la structure générale d'un système d'authentification multimodale utilisant les quatre modalités utilisées dans le projet M2VTS.

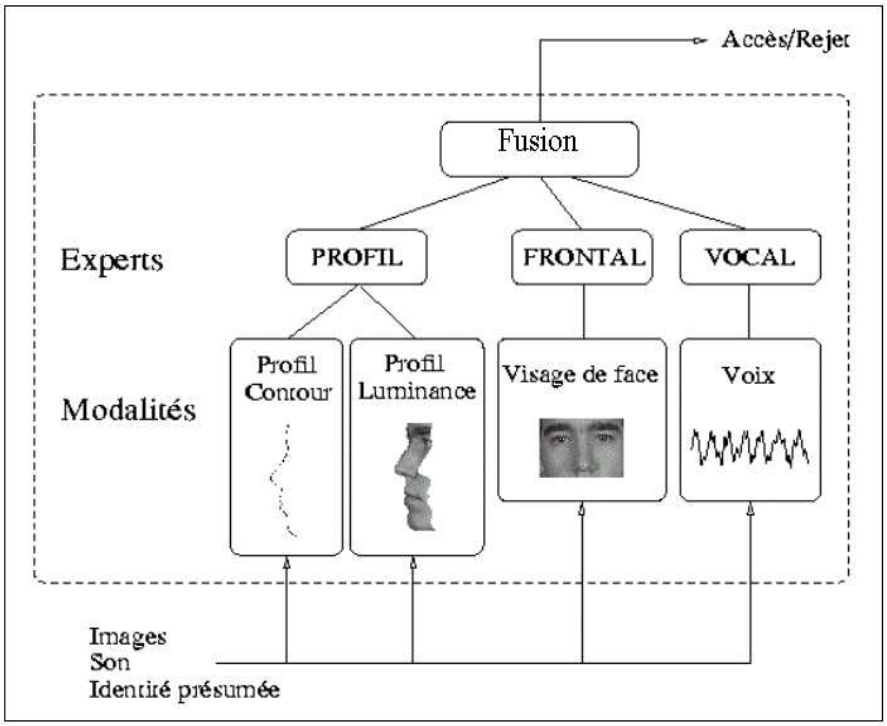
6.4.2 Base de données (S. PIGEON)

La base de données M2VTS est constituée de 37 personnes et de 5 prises de vues pour chacune. La cinquième prise de vue reprend une collection de cas plus difficiles à traiter : défauts de mise au point, mauvais rapports signal à bruit dans l'image ou le son, yeux clos, visages voilés par une écharpe, présence d'un chapeau, etc. Chaque enregistrement fut espacé d'une semaine au minimum, à moins que des modifications majeures du visage soient intervenues entre-temps. Lors de chaque prise de vue, il a été demandé au sujet de compter de "0" jusqu'à "9" dans sa langue maternelle (36 francophones, 1 catalan), puis de tourner la



rr

FIG. 6.2 – *Courbes DET de nos systèmes de fusion SVM et le système de référence sur les accès hommes (les courbes sont présentées par TEE croissant)*



rr

FIG. 6.3 – *La structure générale d'un système d'authentification multimodale utilisant les quatre modalités utilisées dans le projet M2VTS*

tête de façon continue en passant par les positions angulaires suivantes : 0°, -90°, 0°, +90° puis retour à 0 degré. Si la personne porte des lunettes, il lui est alors demandé de les retirer et de faire ce même mouvement une seconde fois, A partir de cet enregistrement, trois séquences d'images sont extraites : la séquence *voix*, la séquence *mouvement* et la séquence *mouvement sans lunettes* dans le cas où une telle séquence a été filmée. La première séquence peut être utilisée à des fins d'authentification de la parole, de reconnaissance dynamique du visage vu de face (en choisissant automatiquement la ou les images les plus appropriées dans la séquence) ou encore pour l'étude de la corrélation entre la voix et le mouvement des lèvres. Les deux autre séquences sont destinées à des fins d'authentification du visage uniquement et donnent accès à la typologie tridimensionnelle de celui-ci grâce au mouvement de rotation. Ces séquences peuvent être utilisées pour implémenter et comparer des techniques telles que la reconnaissance du visage de face, de profil, de vue intermédiaires ou multiples.
Un matériel offrant une bonne résolution [70] a été utilisé lors de l'enregistrement de la base de données, laissant le choix à l'utilisateur de dégrader la qualité des images par la suite pour simuler un système d'acquisition bon marché. Après divers conversations de format, la résolution finale des séquences d'images est de 286×350, en 25Hz progressif. Le son, quant à lui, a été échantillonné à 48kHz sur 16 bits.
Mis à part le cas de la cinquième prise de vue, cette base de données peut être considérée comme ayant été produite dans des conditions quasi idéales : bonne qualité d'images, enregistrement intérieur, illumination presque constante, fond gris uniforme, etc. Aussi (et surtout) les personnes filmées ont fait de leur mieux pour suivre les instructions qui leur étaient données. Malgré tout, certains écarts par rapport à l'idéal théorique peuvent être remarquées :

- certains personnes ne parviennent pas à tourner leur tête convenablement et l'on peut noter une translation horizontale du visage dans la direction de rotation, une inclinaison vertical du visage variable selon l'angle de rotation ou encore une couverture incomplète des 180 degrés ;
- centaines personnes peuvent avoir la bouche ouverte lors d'une prise de vue et fermé dans une, débouchant ainsi sur différents contours du profil ;
- la direction de départ du mouvement de rotation de la tête peut différer d'une prise de vue à l'autre ;
- la focale de la caméra n'a pas été fixée (différents facteurs d'échelle) ;
- certaines personnes parlent très faiblement, ce qui diminue significativement

le rapport signal sur bruit ;
- certaines ne peuvent pas s'empêcher de sourire ou rire pendant l'enregistrement ;
- la vitesse de rotation de la tête peut varier d'une façon considérable entre les différentes prises de vues, mais aussi au sein de la même séquence ;
- un temps d'exposition réduit peut engendrer un flou de "bougé" lors de mouvements de rotation rapides.

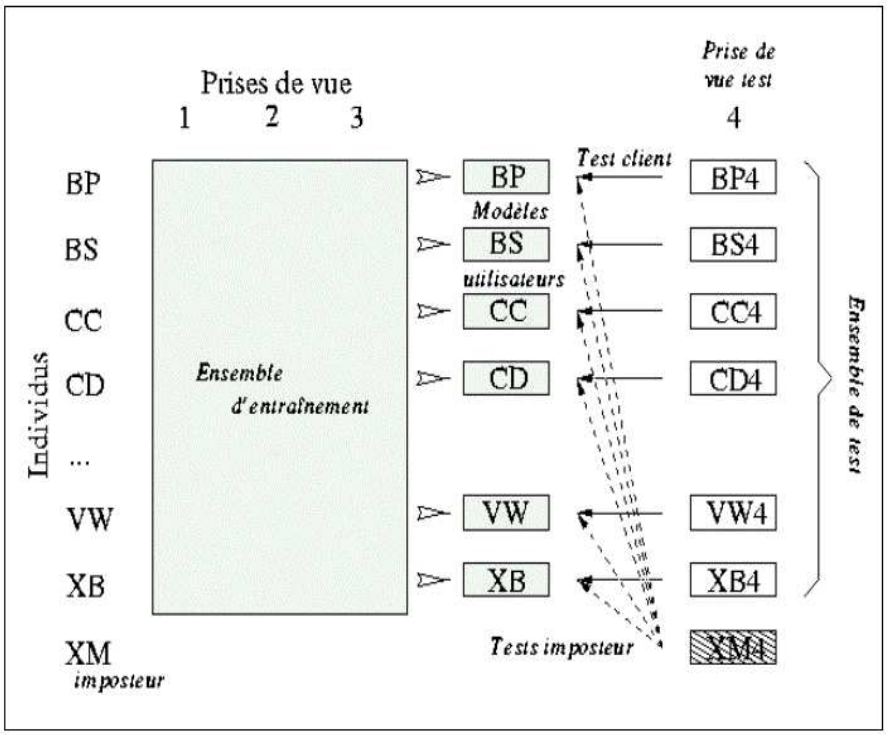
Néanmoins, de telles imperfections apparaîtront également en pratique et l'on peut raisonnablement penser qu'un système qui ne parviendrait pas à donner des résultats performants sur cette base de données, ne pourrait pas non plus faire face à des conditions opératoires réelles.

6.4.3 Protocole expérimental

Dans ce protocole expérimental, seules les 4 premières prise de vues ou sessions sont utilisées. L'ensemble des données a été divisé en deux sous-ensembles disjoints, apprentissage et test. La procédure pour construire ces deux sous-ensembles est la suivante : une session est mise à l'écart pour former le sous-ensemble de test, les 3 sessions restantes seront utilisées en phase d'apprentissage. Ensuite, à chaque fois qu'on fixe la session de test et les sessions d'apprentissage, un individu parmi les 37 personnes participants à cette base de données est mis à l'écart et considéré comme imposteur. Ainsi, on obtient 4×37 groupes élémentaires de test, soit $4\times37\times36 = 5328$ tests clients et autant de test imposteurs. En fixant un individu comme imposteur et une session comme session de test, nous réalisons 72 accès test dont 36 accès clients et 36 accès imposteur. En même temps sur les données d'apprentissage où les sessions de l'individu considéré comme imposteur ne sont pas utilisées, on réalise 3888 accès dont 108 des accès clients et 3780 accès imposteurs. La figure 6.4 empruntée à Stéphane Pigeon présente la procédure de test dans le cas où la personne XM et la session 4 sont mises à l'écart.

6.4.4 Résultats

Les résultats sont présentés sous forme de courbes de DET. La figure 6.5 présente 4 courbes DET représentant chacune les performances d'une des 4 modalités de la base de données M2VTS.



rr

FIG. 6.4 – *La procédure de test dans le cas où la personne XM et la quatrième session sont mises à l'écart*

Les résultats montrent que le TEE est de 9.75% pour la modalité "frontal", 11%

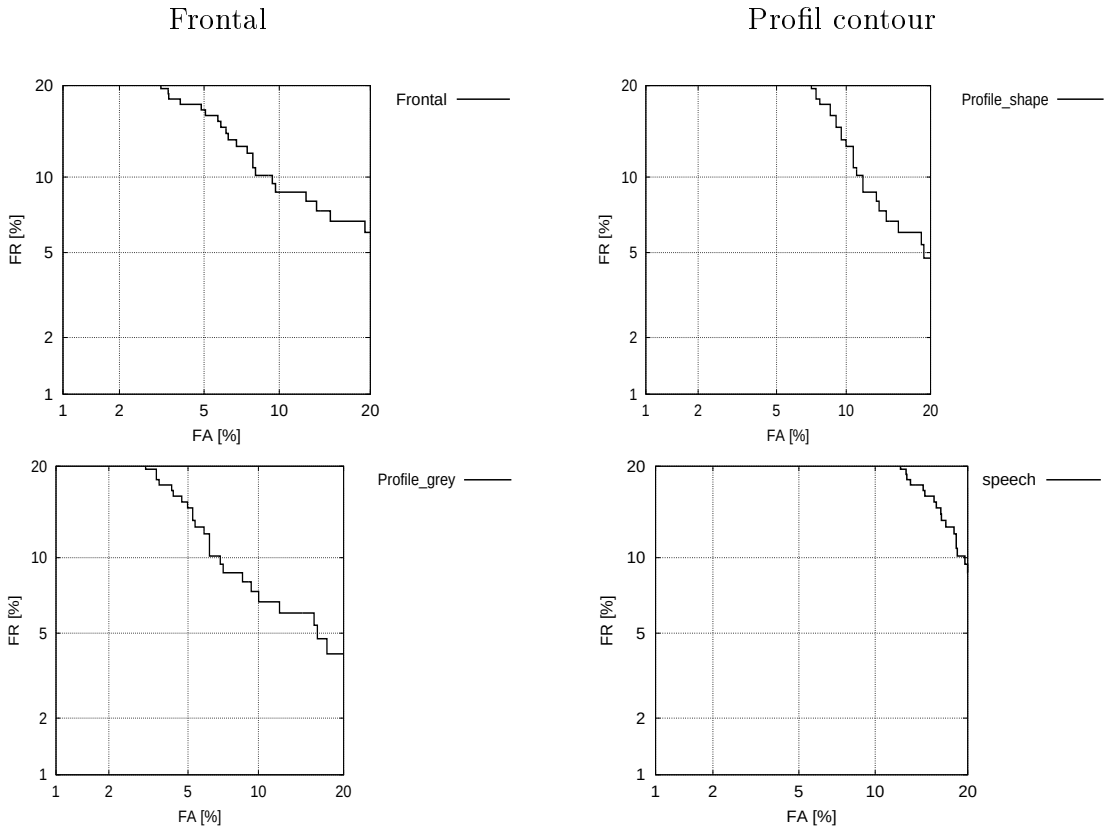


FIG. 6.5 – *Courbes DET obtenues sur chacune des quatre modalités*

pour la modalité "profil contour labellisé profil_shape", 8.5% pour la modalité "profil luminance labellisé profil_grey" et 15% pour la modalité "parole".

La figure 6.6 présente les courbes DET obtenues suite à la fusion des 4 modalités de la base M2VTS. Ces résultats concernent 4 systèmes de fusion : un système de référence labellisé " sys-ref " où le score de décision est la moyenne arithmétique de tous les scores obtenus par toutes les modalités et trois systèmes de fusion utilisant les SVM avec différents noyaux (linéaire, polynomial de degré 3 et le noyau RBF avec $\sigma^2 = 0.01$). Les résultats montrent une grande amélioration des performances d'authentification de tout les systèmes de fusion par rapport aux performances de chaque modalité. Ce qui justifie l'utilisation des techniques de fusion dans de telles applications. Les résultats montrent également que tous les systèmes de fusion utilisant les SVM sont plus performants que le système de

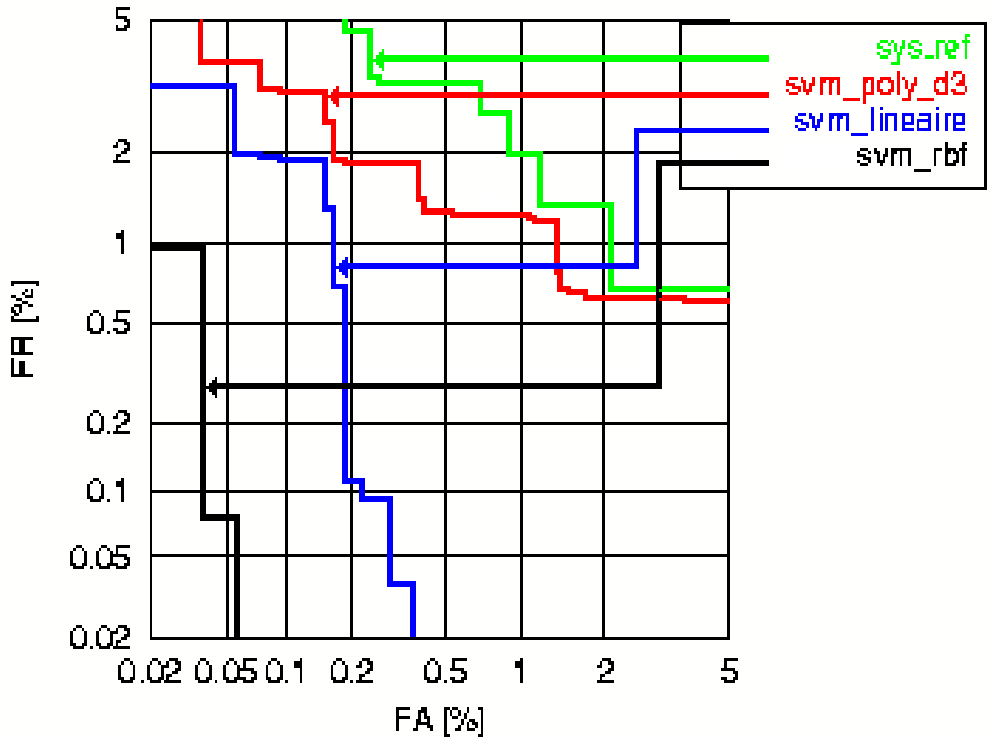


FIG. 6.6 – *Courbes DET obtenus avec le système de fusion de référence et trois systèmes de fusion utilisant les SVM (les courbes sont présentées par TEE croissant)*

référence. La meilleure performance revient au SVM au noyau RBF où le TEE est de 0.06% suivie par le système SVM au noyau linéaire qui a un TEE de 0.19%. Par contre le TEE du système SVM au noyau polynomial de degré 3 est 1.3% contre 1.5% pour le système de référence.
Plusieurs travaux semblables ont été réalisés sur la base de données M2VTS. Dans les travaux de thèses de P. Verlinde [45][93][94] et S. Pigeon [70][71], des études comparatives de plusieurs techniques de fusion ont été réalisées. Les résultats présentés sont très intéressants et atteignent un TEE de 0.1% dans les travaux de P. Verlinde et 0.5% dans les travaux de S. Pigeon. Une autre étude a été présentée par S. Ben yacoub utilisant les SVM pour la fusion des scores sur M2VTS [8]. Cette étude montre que les SVM sont des techniques très performantes en comparant avec un système calculant la moyenne des scores et un système bayésien. Une des dernière études utilisant les SVM pour la fusion des scores M2VTS a été réalisée par G. Gutschoven [45]. Les résultats présentés dans cette étude montrent que les SVM associés à certaines méthodes de normalisation des scores (pondération des scores de chaque expert par son taux de réussite) sont des techniques très performantes qui peuvent atteindre un TEE de 0%.

6.5 Bilan

Dans ce chapitre, nous avons décrit deux expériences de fusion utilisant les techniques SVM. Ces expériences concernent deux types de fusion : la fusion de méthode, c'est l'expérience de la fusion sur les scores des participants NIST'2001 et la fusion de mode, c'est l'expérience de fusion sur les données M2VTS.
Dans ces expériences qui entrent dans le cadre du projet BIOMET, nous avons montré que les SVM peuvent être très efficaces pour des tâches de fusion avec un TEE de 0.06%. Ce qui confirme les conclusion des travaux précédents principalement ceux réalisés par P. Verlinde et B. Gutschoven [45] et S. ben Yacob [8] sur la fusion des scores des données M2VTS et l'étude de P. Verlinde sur la fusion des scores NIST [93].
Les résultats que nous avons obtenus ne sont pas comparables avec ceux obtenus par les travaux précédents pour des raisons de différence de protocoles expérimentaux (le nombre de données d'apprentissage et de test n'est pas le même d'un protocole à l'autre), mais confirment tous que les SVM sont des techniques très intéressantes et surtout prometteuses pour des tâches de fusion de scores.

Conclusions et perspectives

6.6 Conclusions

Ce travail thèse s'inscrit dans le domaine de la Vérification Automatique du Locuteur (VAL). Un système de VAL consiste à vérifier l'identité d'une personne à partir de son signal de parole. Utilisés principalement pour des raisons de sécurité et/ou confidentialité, les systèmes de VAL sont souvent développés dans des applications téléphoniques où le signal de parole reste le moyen le plus fiable pour la vérification de l'identité. On peut également utiliser la VAL associée à d'autre modalités (ex : vérification de signatures, analyse du visage, des empreintes digitales et de la forme de la main) dans des systèmes de vérification et d'authentification multimodale de l'identité.
Bien que jusqu'à maintenant on n'a pas pu déterminer des caractéristiques physiques discriminant un locuteur d'un autre, le domaine de la VAL a connu les dix dernières années une progression considérable. Ainsi, certaines systèmes dans des environnements contrôlés ont pu avoir des performances très intéressantes. On peut distinguer deux type de VAL : la VAL en mode dépendant du texte où le locuteur doit prononcer un message connu a priori par le système et la VAL en mode indépendant du texte où il n'y a aucune contrainte sur le message que le locuteur doit prononcer.

L'objectif de ce travail de thèse est d'introduire les techniques d'apprentissage statistique SVM "Support Vector Machines" dans le domaine de la VAL. Ainsi nous avons proposé plusieurs approches permettant d'utiliser ces techniques en mode dépendant et indépendant du texte. Nous avons également étudié ces techniques pour des tâches de fusion de méthodes et de fusion de modes.

Une des premières applications que nous avons proposé concerne l'utilisation des SVM en VAL en mode dépendant du texte. Cette étude a été menée dans le cadre du projet Européen PICASSO [13] dans une application utilisant des

mots de passe publics. Dans cette approche, nous avons proposé une nouvelle modélisation de locuteur basée sur la transcription phonétique des mots de passe pour construire des vecteurs d'entrée pour les SVM. Cette modélisation consiste à calculer des vecteurs basés sur la vraisemblance de chaque phonème constituant le mot de passe et les modèles génériques des phonèmes (cf chapitre 5). Les résultats obtenus sont intéressants, mais malheureusement un problème d'enregistrement de la première version de la base de données POLYVAR utilisée dans nos expériences nous a empêché de valider cette proposition en comparant avec d'autre techniques classiques de ce domaine. Dernièrement, nous avons repris ces expériences dans le cadre d'un autre projet nommé MAJORDOME.

Concernant la VAL en mode indépendant du texte, nous avons proposé des systèmes hybrides GMM-SVM où les GMM sont utilisés en phase de modélisation et les SVM en phase de décision. Dans ces systèmes, nous avons proposé trois nouvelles représentations de données permettant de combiner l'efficacité des GMM en modélisation et la performance des SVM en décision. Grâce aux résultats obtenus sur les données NIST'1999 et NIST'2001 (un sous ensemble de la base SWITCHBOARD) nous avons pu montrer que les techniques SVM sont plus performantes que la technique classique LLR considérée comme étant l'état de l'art dans le domaine de la VAL.

Dans le cadre du projet BIOMET mené par le GET (Groupe des Écoles de Télécommunications) sur l'authentification biométrique, nous avons testé les techniques SVM pour deux tâches de fusion. Les premiers tests concernent la fusion de méthodes où nous avons fusionné les scores obtenus par les participants aux évaluations NIST'2001. Les seconds tests concernent la fusion de modes menés sur les scores obtenus pour quatre modalités différentes de la base de données M2VTS. Les résultats obtenus par notre fusion SVM ont été comparés à un système de référence basique qui prend la décision en se basant sur la moyenne des scores à fusionner. Ces résultats montrent une grande efficacité des SVM pour les tâches de fusion ce qui rejoint les conclusions des travaux précédents réalisés par P. Verlinde sur la fusion de méthode [93][94] et S. Ben Yacob [8], B. Gutschoven et P. Verlinde en fusion de mode [45].

6.7 Perspectives

Les SVM sont des techniques très récentes (proposées par V. Vapnik en 1995). Plusieurs travaux utilisant ces techniques dans différentes applications ont montré qu'elles sont très efficaces, très intéressantes et surtout très prometteuses.
Ce travail de thèse représente une des premières tentatives d'introduire les SVM dans le domaine de la VAL. Les résultats que nous avons obtenu par nos systèmes sont très intéressants. Cependant, tous les problèmes liés aux approches que nous avons proposés sont loins d'être résolus. De nombreuses études peuvent être envisagées afin de valider ces approches et beaucoup de travail reste à faire pour améliorer les performances des systèmes proposés.
En VAL en mode indépendant de texte : Dans les systèmes que nous avons présentés, l'utilisation des SVM n'aurait pu être possible sans les nouvelles représentations des données que nous avons proposées. L'étude que nous avons menée a montré que ces systèmes ont de meilleures performances qu'un système classique (représentant l'état de de l'art) basé sur la technique LLR. Une question importante s'impose à ce niveau-là. Est ce que ces meilleurs résultats sont dûs à l'utilisations des SVM ou aux nouvelles représentations des données que nous avons proposées ?
Pour répondre à cette question, on peut envisager une étude qui consiste à utiliser ces même représentations de données avec d'autres techniques discriminates comme les K-Plus Proches Voisins (KPV), les Réseaux de Neurones (RN) ..etc.
Dans le cadre de ce travail de thèse nous nous sommes basés sur l'expérience de notre équipe pour régler certains paramètres de notre système, notamment le nombre de gaussiennes à utiliser pour la modélisation des clients. Dans tous les systèmes que nous avons proposés, nous avons utilisé des modèles GMM de 128 gaussiennes. Ce choix est basé sur plusieurs études que nous avons menées à l'ENST et dans le cadre du consortium ELISA. Ces études ont montré que les performances d'un système classique sont très satisfaisantes avec 128 gaussiennes et que l'amélioration obtenue en augmentant le nombre de gaussiennes n'est pas intéressante si on prend en considération le temps de calcul. Une étude semblable sur nos systèmes GMM-SVM peut être très intéressante surtout que les représentations des données que nous avons proposées dépendent principalement de la modélisation GMM utilisée.
Enfin, l'utilisation des techniques de pré-traitement de données (ex : ACP Analyse en Composantes Principales) peut être très efficace pour améliorer les per-

formances de nos systèmes hybrides GMM-SVM.

En VAL en mode dépendant de texte : Une première étude envisageable en VAL en mode dépendant du text consiste à reprendre l'approche que nous avons proposée et la tester sur la bonne version de la base POLYVAR et sur un nombre de données plus significatif (ex : le protocole expérimental du projet PICASSO).
Des représentations de données semblables à celles proposées en mode indépendant du texte peuvent être très intéressantes. Par exemple : en utilisant une modélisation HMM, on peut imaginer des vecteurs construits à partir des scores obtenus au niveau de chaque états du modèle HMM...etc

En fusion des scores : Plusieurs travaux de fusion ont montré que les SVM sont des techniques très efficaces. Dans le projet BIOMET, une des expériences qu'on peut envisager consiste à faire de la fusion sur sous ensembles disjoints de modalités et pourquoi pas fusionner les scores de chaque système de fusion...

Dans l'absence d'une étude théorique sur le choix des noyaux à utiliser, la reprise de toutes nos expériences avec des noyaux autres que ceux utilisés dans ce travail de thèse (noyau linéaire, noyau polynomial et noyau RBF) peut être très intéressante. Ces expériences vont nous permettre de déterminer le noyau le plus adapté à chacune des tâches traitées.

Enfin une étude qui consiste à chercher de nouveaux noyaux ne peut être qu'enrichissante pour les techniques SVM elles même et leurs domaines d'applications.

Bibliographie

[1] B. S. Atal " Automatic recognition of speakers from their voice " Proceeding of the IEEE, vol. 64(4), pages 460-475, 1976.

[2] R. Auckenthaler, J. S. Mason " Score normalisation for text-independent speaker verification systems " Digital Signal Processing Journal, Vol. 10, N 1-3, pages 42-54, 2000.

[3] R. Auckenthaler, J. S. Mason " Score normalisation in a multi-band speaker verification system " Workshop on Speaker Recognition and its Commercial and Forensic Applications (RLA2C), pages 102-105, 1998.

[4] Y. Bennani " Approches connexionnistes pour la reconnaissance automatique du locuteur : Modélisation et Identification " Thèse de l'Université de Paris-Sud, 1992.

[5] Y. Bennani, P. Gallinari " Connectionist approaches for automatic speaker recognition " ESCA, Workshop on Automatic Speaker Recognition identification verification, Martigny, pages 95-102, 1994.

[6] Y. Bennani " A connectionist approach for speaker verification " ICASSP'90, pages 265-268, 1990.

[7] K. Bennet, A. Demiriz " Semi-supervised support vector machines " Proceedings NISP'98, 1998.

[8] S. Ben Yacoub, " Multi-Modal data for person authentication using support vector machines " In Proceedings of the second International

Conference on Audio- and Video-based Biometric person Authentication (AVBPA'99), pages 25-30, 1999.

[9] L. Besacier " Un modèle parallèle pour la reconnaissance automatique du locuteur " Thèse de l'Université d'Avignon, 1998.

[10] L. Besacier, J. F. Bonastre " Subband approach for automatic speaker recognition : optimal division of the frequency domain " Audio- and Video-based Biometric Person Authentication (AVBPA), Bigun, et.al, Eds, Springer LNCS 1206, 1997.

[11] F. Bimbot, " Synthèse de la parole : du segments au règles, avec utilisation de la décomposition temporelle " Thèse de l'Ecole Nationale Supérieure des Télécommunications, 1988.

[12] F. Bimbot, G. Chollet "Assessment of speaker verification systems," In : EAGLES Handbook on Spoken Language Systems, Chapter 11, Mouton de Gruyter, 1997.

[13] F. Bimbot, M. Blomberg, L. Boves, G. Chollet, C. Jaboulet, B. Jacob, J. Kharroubi, J. Koolwaaij, J. Lindberg, J. Mariéthoz, C. Mokbel, H. Mokbel " An overview of the PICASSO project research activities in speaker verification for telephone application" Eurospeech'99, Budapest (Hongrie) , 1999.

[14] C. M. Bishop " Neural networks for pattern recognition" Clarendon Press, Oxford 1995.

[15] J. F. Bonastre " Stratégie analytique orientée connaissance pour la caractérisation de la reconnaissance du locuteur " Thèse de l'Université d'Avignon, 1994.

[16] B. E. Boser, I. Guyon, V. Vapnik " A training algorithm for optimal margin classifiers " Proceedings of the 5th Annual ACM Workshop on Computer Learning Theory, pages 144-152, 1992.

[17] C. Burges " A tutorial on support vector machines for pattern recognition " Data Mining and Knowledge Discovery, 2-2, 1998.

[18] M. J. Carey, E. S. Parris " Speaker verification using connected words " Proceedings of Institute of Acoustics, Vol. 14, pages 95-100, 1998.

[19] G. Celeux, J. Diebolt " L'algorithme SEM : un algorithme d'apprentissage probabiliste pour la reconnaissance de mélange de densités " Revue de Statistique Appliquée, 34(2), 1986.

[20] D. Charlet " Authentification vocale par téléphone en mode dépendant du texte " Thèse de l'Ecole Nationale Supérieure des Télécommunications, 1997.

[21] G. Chollet, J. Cernocky, G. Gravier, J. Hennebert, D. Petrovska-Delacretaz, F. Yvon " Towards fully automatic speech processing techniques for interactive voice servers " G. Chollet, M. Di Benedetto, A. Esposito, M. Marinaro, Editors, Speech Processing, Recognition and Artificial neural network, proceedings of the 3rd International School On Neural Nets, Eduardo R. Caianiello, Spring Verlag, 1998.

[22] G. Chollet, J. L. Cochard, A. Constantinescu, C. Jaboulet, P. Langlais " Swiss French polyphone and PolyVar : telephone speech databases to model inter- and intra-speaker variability " Technical Report, RR 96-01, IDIAP, 1996.

[23] P. Ciarlet " Introduction à l'analyse numérique matricielle et à l'optimisation " Masson, 1994, nouvelle édition.

[24] A. P. Dempster, N. M. Laid, D. B. Durbin " Maximum Likelihood from incomplete data via the EM algorithm " J. Royal Statistical Soc, volume 39(39), pages 1-38, 1977.

[25] ELISA Consortium " The ELISA systems for NIST'99 evaluation in speaker detection and tracking " Digital Signal Processing Journal, Vol. 10 N 1-3, pages 143-153, 2000.

[26] ESCA " Workshop on Automatic Speaker Recognition Identifica-

tion and Verification " G. Chollet, F. Bimbot, A. Paoloni, Martigny April 1994.

[27] R. Fernandez " Analyse et implémentation du classifieur support vector machines " Mémoire de DEA de l'Université Pierre et marie curie, 1996.

[28] R. Fernandez, E. Viennet " Face identification with support vector machines " Proceedings ESANN'99, 1999.

[29] R. Fernandez, " Machines de support pour la reconnaissance de formes : propriétés et applications " Thèse de l'Université Paris 13, Juin 1999.

[30] S. Fine, J. Navratil, R. A. Gopinath " Enhancing GMM Score Using SVM 'Hints' ", Eurospeech 2001, Vol 3, pp 1757- 1760, 2001 .

[31] R. Fletcher " Practical methods of optimization " John Wiley, 1987. (2nd edition).

[32] C. Fredouille " Reconnaissance du locuteur et approche statistique : Information dynamiques et normalisation bayésienne des vraisemblances " Thèse de l'Université d'Avignon, 2000.

[33] C. Fredouille, J. F. Bonastre, T. Merlin " Segmental normalization for robust speaker verification " Proceeding Workshop on robust methods for speech recognition in adverse conditions COST 249, 1998.

[34] C. Fredouille, J. Hennebert, C. Jaboulet " Unsupervised incremental enrolment experiments" Rapport technique, projet Européen PICASSO, Ubilab/UBS, Zurich, Octobre 1999.

[35] C. Fredouille, J. F. Bonastre, T. Merlin " AMIRAL : a block-segmental multi-recognizer approach for automatic speaker recognition " Digital Signal Processing Journal, Vol. 10 N 1-3, pages 172-197, 2000.

[36] S. Furui " An overview of speaker recognition technology " Work-

shop on Automatic Speaker Recognition Identification and Verification, Martigny April 1994.

[37] J. L. Gauvain " Maximum a posteriori estimation for multivariate mixture observations of Markov Chains " IEEE Trans, on Speech and audio processing, 2(2), 1994.

[38] D. Genoud, G. Gravier, F. Bimbot, G. Chollet " Combining methods to improve speaker verification decision " ICSLP, vol. 3, pages 1756-1759, 1996.

[39] D. Genoud " Reconnaissance et transformation de locuteurs " Thèse de l'Ecole Polytechnique Fédérale de Lausanne, 1999.

[40] G. Gravier, J. Kharroubi, G. Chollet, F. Bimbot, R. Blouet, M. Seck, J. F. Bonastre, C. Ferdouille, T. Merlin, S. Pigeon, J. Cernocky, D. Petrovska, B. Nedic, I. Magrin-Chagnolleau, G. Durou " The ELISA 99 speaker recognition and tracking system " Workshop on Automatic identification Advanced technologies, AutoId'99, 1999.

[41] G. Gravier " Analyse statistique à deux dimensions pour la modélisation segmentale de parole : Application à la reconnaissance " Thèse de l'Ecole Nationale Supérieure des Télécommunications, 2000.

[42] G. Gravier, G. Chollet, " Comparison of normalisation techniques for speaker recognition " Workshop on Speaker Recognition and its Commercial and Forensic Applications (RLA2C), pages 97-100, Avignon, 1998.

[43] G. Gravier, J. Kharroubi, G. Chollet, " On the use of prior knowledge in normalization scheme for speaker verification " Digital Signal Processing Journal, Vol. 10, N 1-3, pages 213-225, 2000.

[44] Y. Guermeur " Théorie de l'apprentissage de Vapnik et support vector machines " rapport technique, LIP URA CNRS 1398, Ecole Supérieure de Lyon, 1998.

[45] B. Gutschoven, P. Verlinde "Multimodal identity verification using

support vector machines" Fusion 2000, session ThB3, 2000.

[46] J. P. Haton " Neural network for automatic speech recognition : a review " G. Chollet, M. Di Benedetto, A. Esposito, M. Marinaro, Editors, Speech Processing, Recognition and Artificial neural network, proceedings of the 3rd International School On Neural Nets, Eduardo R. Caianiello, Spring Verlag, pages 259-280, 1998.

[47] A. Higgins, L. Bahler, J. Porter " Speaker verification using randomized phase prompting " Digital Signal Processing, vol. 1, pages 89-106, 1991.

[48] M. M. Homayounpour " Vérification vocale d'identité : dépendante et indépendante du texte " Thèse de l'Université Paris-sud centre d'Orsay, 1995.

[49] B. Jacob, J. Mariéthoz, G. Gravier, F. Bimbot " Robustesse de la vérification du locuteur par un mot de passe personnalisé " Jep2000, pages 357-360, 2000.

[50] D. Jouvet " Reconnaissance des mots connectés indépendamment du locuteur par des méthodes statistiques " Thèse de l'Ecole Nationale Supérieure des Télécommunications, 1988.

[51] J. Kharroubi, G. Chollet " Utilisation de mots de passe personnalisés pour la vérification du locuteur " Jep2000, pages 361-364, 2000.

[52] J. Kharroubi, G. Chollet " Text-independent speaker verification using support vector machines ", Résumé pour Student Forum ICASSP'2001.

[53] J. Kharroubi, D. Petrovska-Delacretaz, G. Chollet, " Text-Independant speaker verification using support vector machines " Odyssey, pages 51-54, 2001.

[54] J. Kharroubi, D. Petrovska-Delacretaz, G. Chollet, " Combining GMM's with support vector machines for speaker verification " Eurospeech, pages 1761-1764, 2001.

[55] C. J. Leggetter, P. C. Woodland, " Maximum Likelihood Linear Regression for speaker adaptation of continuous Hidden Markov Models " Computer Speech and Language, Vol. 9, pages 171-185, 1995.

[56] K. P. Li, E. H. Wrench Jr " An approach to text-independent speaker recognition with short utterances " IEEE trans, on ASSP, pages 555-558, 1983.

[57] I. Magrin-Chagnolleau " Approches statistiques et filtrages vectoriel de trajectoires spectrales pour l'identification du locuteur indépendante du texte " Thèse de l'Ecole Nationale Supérieure des Télécommunication, 1997.

[58] I. Magrin-Chagnolleau, G. Gravier, R. Blouet, " Overview of the Elisa consortium research activities " Odyssey, pages 67-72, 2001.

[59] T. Matsui, S. Furui " Similarity normalisation method for speaker verification based on a posteriori probability " ESCA, Workshop on Automatic Speaker Recognition identification verification, Martigny, pages 59-62, 1994,.

[60] A. Martin, M. Przybocki " The NIST 1999 Speaker recognition evaluation -An Overview " Digital Signal Processing Journal, Vol. 10 N 1-3, pages 1-18, 2000.

[61] A. Martin et al, " The DET curves in assessment of detection task performance " Eurospeech, Vol. 4, pages 1895-1898, 1997.

[62] M. Minoux, " Programmation mathématique : théorie des algorithmes (tome1) " Dunod, 1983.

[63] NIST " Speaker Recognition Workshop " MITAGS, March 27-28, 1996.

[64] J. Oglesby " What's in a number ? : moving beyond the Equal Error Rate " Speech communication, vol. 17(1-2), pages 193-209, 1995.

[65] E. Osuna, R. Freund, F. Girosi " Improved training algorithm for support vector machines " NNSP'97, VII - Proceeding of the IEEE Workshop, 1997.

[66] E. Osuna, R. Freund, F. Girosi " Support vector machines : Training and applications " Technical Report, AIM-1602, MIT, 1997.

[67] E. Osuna, R. Freund, F. Girosi " Training support vector machines : An application to face detection " Proceedings CVRP, Puerto Rico, 1997.

[68] T. Pham, D. Tran, M. Wagner " Speaker verification using relaxation labelling " Workshop on Speaker Recognition and its Commercial and Forensic Applications (RLA2C), pages 102-105, 1998.

[69] J. B. Pierrot " Elaboration et validation d'approches en vérification du locuteur " Thèse de l'Ecole Nationale Supérieure des Télécommunications, 1998.

[70] S. Pigeon " The M2VTS multimodal face database (release 1.00) " European ACTS Deliverable AC102/UCL/WP1/DS/P/161, 1996.

[71] S. Pigeon " Authentification multimodale d'identité " Thèse de l'Université Catholique de Louvain, 1999,.

[72] L. R. Rabiner, B. H. Juang, " An introduction to Hidden Markov Models " IEEE ASSP magazine, 1986.

[73] L. R. Rabiner, B. H. Juang " Fundamentals of speech recognition " Signal Processing, Prentice Hall, Englewood Cliffs, NJ, 1993.

[74] D. A. Reynolds " A gaussian mixture modelling approach to text-independent speaker identification " Thesis of Georgia Institute to Technology, 1992.

[75] D. A. Reynolds " Experimental evaluation of features for robust speaker identification " IEEE transaction Speech Audio Processing, vol. 2,

pages 639-643, 1994.

[76] D. A. Reynolds " Speaker identification and verification using gaussian mixture speaker models " Workshop on Automatic Speaker Recognition Identification Verification, Martigny, pages 27-30, 1994.

[77] D. A. Reynolds " Comparison of background methods for text-independent speaker verification " Proceeding Eurospeech'97, pages 963-966, 1997.

[78] D. A. Reynolds, T.F. Quatieri, R. B. Dunn " Speaker verification using adapted gaussian mixture models ", Digital Signal Processing Journal, Vol. 10 N 1-3, pages 19-41, 2000.

[79] A. E. Rosenberg, J. Delong, C. H. Lee, B. H. Juang, F. K. Soong " The use of cohort normalized scores for speaker verification " ICSLP'92, pages 599-602, 1992.

[80] G. Saporta " Probabilité, analyse des données statistique " Edition Technique, 1990.

[81] C. Saunders, M. O. Stitson, J. Weston, L. Bottou, B. Scholkopf, A. Smola " Support vector machines : Reference Manual " technical report CSD-Tr-98-03, 1998.

[82] M. Schmidt, H. Gish, " Speaker identification via support vector classifier " Proceedings ICASSP'96, pages 105-108, 1996.

[83] M. Schmidt, " Identifying with Support Vector Networks ", Proceedings of Interface'96, 1996.

[84] B. Scholkôpf " Support vector learning " Thesis, R. Oldenbourg verlag, Munich, 1997.

[85] B. Scholkôpf, C. Burges, V. Vapnik " Incorporating invariances in support vector learning machines " Proceedings ICANN'96, Menlo Park,

AAAI Press, 1996.

[86] B. Scholkôpf, K. Sung, C. Burges, F. C. Girosi, T. Poggio, V. Vapnik " Comparing support vector machines with gaussian kernel to radial basis functions classifiers " Technical Report AIM-1599, MIT, 1996.

[87] M. O. Stitson, J. Weston, A. Gammerman, V. Vovj, V. Vapnik " Theory of vector machines " technical report CSD-TR-96-18, Royal Holloway University of London, 1996.

[88] F. K. Soong, A. E. Rosenberg, L. R. Rabiner, B. H. Juang " A vector quantization approach to speaker recognition " ICASSP'98, pp 387-390, 1985.

[89] V. Vapnik " Estimation of dependencies based on empirical data " Springer-Verlag, New York, 1982.

[90] V. Vapnik, A. Chervonenkis " The necessary and sufficient conditions for consistency of the method of empirical risk minimisation " Pattern recognition and Image Analysis, 1(3), pages 284-305, 1991.

[91] V. Vapnik " The nature of statistical learning theory " Spring-Verlag, New York, 1995.

[92] V. Vapnik " Statistical learning theory " J. Willey, 1998.

[93] P. Verlinde " Contribution à la vérification multimodale d'identité en utilisant la fusion de décisions " Thèse de l'Ecole Nationale Supérieure des Télécommunications, 1999.

[94] P. Verlinde, G. Chollet, M. Acheroy " Multi-modal identity verification using expert fusion " Information Fusion, 1(1), pages 17-33, 2000.

[95] E. Viennet, R. Fernandez " Machines à vecteurs de support et réseaux de neurones : comparaisons expérimentales pour l'identification de visages " Proceedings CAP'99, 1999.

[96] J. Zhang " The mean field theory in EM algorithm procedure for Markov random fields " IEEE Trans. Signal Processing, 40(10), 1992.

Annexe A :
Bibliographie Personnelle

M. Fuentes, D. Mostefa, J. Kharroubi, S. Garcia-Salicetti, B. Dorizzi, G. Chollet " Vérification de l'identité par fusion de données biométriques : signatures en ligne et parole " accepté au CIFED'02, Tunisie, 2002.

J. Kharroubi, G. Chollet " Nouveau système hybride GMM-SVM pour la vérification du locuteur " Jep'2002, pages 101-104, Nancy, France, 2002.

J. Kharroubi, D. Petrovska, G. Chollet " Combining GMM's with support vector machines for text-independent speaker verification" Eurospeech'2001, pages 1761-1764, Aalborg, Denmark, 2001.

J. Kharroubi, D. Petrovska, G. Chollet " Text-independent speaker verification using support vector machines " Odyssey'2001, pages 51-54, Chania (Crete), Greece, 2001.

Consortium ELISA " Overview of the 2000-2001 ELISA consortium Research Activities" Odyssey'2001, pages 67-72, Chania (Crete), Greece, 2001.

J. Kharroubi, G. Chollet " Text-independent speaker verification using support vector machines " Résumé pour le Student Forum ICASSP'2001, Salt lake City, USA, 2001.

J. Kharroubi, G. Chollet " Utilisation de mots de passe personnalisés pour la vérification du locuteur " Jep'2000, pages 361-364, Aussois, France, 2000.

G. Gravier, J. Kharroubi, G. Chollet " One the use of prior in normalization schemes for speaker verification " Digital Signal Processing Journal, Vol. 10, N 1-3, pages 213-225, 2000.

Consortium ELISA " The ELISA systems for NIST'99 evaluation in speaker detection and tracking " Digital Signal Processing Journal, Vol. 10 N 1-3, pages 143-153, 2000.

Consortium ELISA "The ELISA 99 Speaker Recognition and Tracking Systems " AUTOID'99, Summit, Morristown, New Jersey, USA, 1999.

F. Bimbot and al " An Overview of the PICASSO Project Research Activities in speaker verification for telephone Applications " Eurospeech'1999, Vol. 5, pages 1963-1966, Budapest, Hungary, 1999.