Utilisation de la structure de mots de passe personnalisés pour la reconnaissance de locuteurs embarquée

AnthonyLarcher[1,2], *Jean-François Bonastre*[1], *JohnS.D.Mason*[2]

1. Laboratoire d'Informatique d'Avignon (LIA), UAPV, France
2. Speech and Image group, Swansea University, Wales, UK
{anthony.larcher, jean-francois.bonastre} @univ-avignon.fr
J.S.D.Mason@swansea.ac.uk

ABSTRACT

Embedded speaker recognition in mobile devices involves a limited amount of computing resource. Both the enrolment and the test have to be done using short audio sequences. Even if they proved their efficiency in more classical situations, GMM/UBM based systems show their limits in this context. This paper deals with this problem and proposes to take into account the linguistic nature of the speech material inside the GMM/UBM framework. The proposed solution mixes the text-independent aspects of the GMM/UBM with a semi-continuous like approach in order to deal with the text-dependent information. This system respects both the resource and the ergonomic constraints of the considered application field. The preliminary experiments are done on the MyIdea database and show the potential of the proposed approach.

Keywords: Speaker Recognition, Embedded System, Password Recognition

1. Introduction

L'utilisation d'un système de reconnaissance du locuteur sur un appareil embarqué est soumise à plusieurs contraintes. En terme de ressources, la mémoire et le temps de calcul sont limités. Au niveau de l'ergonomie, l'utilisation spécifique à ce type d'applications impose d'apprendre les modèles avec peu de données et d'effectuer les tests sur de courtes séquences audio. Les systèmes classiques de reconnaissance du locuteur sont indépendants du texte et fondés sur le paradigme GMM/UBM [1]. Ces systèmes ont montré leur haut niveau de performance, notamment lors des évaluations NIST [7]. Cependant, le peu de données d'apprentissage et les courtes séquences de test dus au contexte embarqué sont peu adaptés aux systèmes UBM/GMM dont les performances dépendent fortement de la quantité de données disponible. Une solution à ce problème consiste à exploiter la structure temporelle de la séquence prononcée en utilisant un mot de passe spécifique au client. L'information structurelle du mot de passe permet alors de compenser la courte durée des séquences de test. L'ajout de cette information temporelle peut être obtenue par l'utilisation d'un système de reconnaissance de mots isolés en parallèle d'un système classique de reconnaissance de locuteur [6]. Le type d'application visée nécessite que le système accepte n'importe quel mot de passe dans n'importe quelle langue. Un système de décodage de la parole utilisant des modèles de phonèmes peut être adapté à une application multilingue en choisissant un jeu de phonèmes adéquat, ce type de modèle sera cependant plus coûteux en terme de mémoire qu'un système modélisant chaque mot de passe par une structure HMM. De plus un système embarqué est confronté à des environnements extrêmement variables. Le modèle acoustique utilisé doit donc être facilement adaptable à l'environnement. Une modélisation utilisant des HMMs ne permet pas de répondre à ces contraintes car leur adaptation nécessite une importante quantité de données.
La solution proposée dans cet article associe les modèles statistiques que sont les GMMs à une architecture originale s'inspirant de la reconnaissance de la parole. Elle utilise le paradigme du GMM/UBM pour modéliser l'espace acoustique et effectuer la reconnaissance du locuteur et une approche HMM/Viterbi qui permet de tirer partie de l'aspect dépendant du texte de l'application. Une architecture semblable a été proposée dans [2] pour la reconnaissance du locuteur et étendue à la reconnaissance de mots isolés dans [5]
L'architecture à trois niveaux est décrite dans la section 2. Les méthodes employées afin de réduire les ressources mémoire et processeur ainsi que les algorithmes d'apprentissage sont également décrits dans cette section. Le protocole expérimental et les résultats obtenus sont décrits dans la section 3 ainsi que la base de données utilisée, MyIdea. La section 4 présente une analyse des résultats et les travaux à venir.

2. Description du Système

Le système proposé combine une représentation statistique de l'espace acoustique concentré dans un unique modèle GMM avec une modélisation de la structure temporelle des mots de passe. Basé sur les Modèles de Markov Cachés Semi-Continus (SCHMM) [8] il comporte trois niveaux.

2.1. EBD, une architecture hiérarchique

La figure 1 montre l'architecture du système proposé, appelé EBD pour *"Embedded LIA_SpkDET"* [3]. Les noeuds de cette architecture sont des modèles GMM. Le premier niveau est le moins spécialisé, il s'agit d'un modèle du monde classique (UBM). Il modélise l'ensemble de l'espace acoustique.
Le niveau intermédiaire contient des modèles de locuteurs indépendants du texte. Ces modèles sont obtenus par adaptation du modèle UBM, classique en RAL ; chaque modèle de locuteur est dérivé de l'UBM en utilisant l'algorithme EM et un critère de *Maximum A Posteriori* (MAP) [4]. Seules les moyennes des distributions sont adaptées. Les autres paramètres sont ceux du modèle UBM.
Le dernier niveau utilise les propriétés des modèles SCHMM pour modéliser l'information dépendante du texte. Ces modèles prennent en compte la structure

temporelle des mots de passe des utilisateurs. Chacun des états des SCHMMs est obtenu en dérivant le modèle du client du niveau intermédiaire. Seuls les poids de certaines gaussiennes du modèle sont adaptés, les autres paramètres du modèle sont directement issus du modèle de locuteur indépendant du texte.

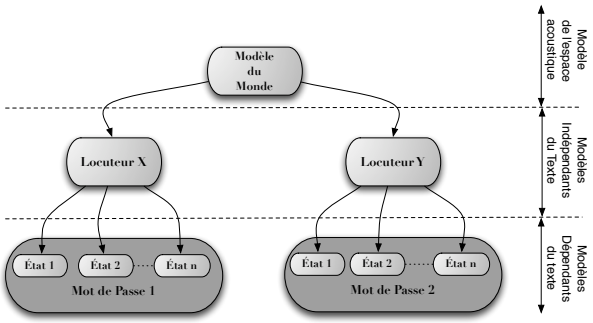

Fig. 1: Architecture du système EBD.

2.2. Apprentissage

L'apprentissage du modèle EBD est effectué en trois étapes, correspondant à chaque niveau de l'architecture. L'UBM du premier niveau est entraîné avec une grande quantité de données grâce à l'algorithme EM/ML pour modéliser au mieux l'espace acoustique. Les modèles indépendants du texte du deuxième niveau sont appris en adaptant les moyennes de l'UBM avec un algorithme EM/MAP et l'ensemble des données disponibles pour le locuteur, indépendant du contenu linguistique. Les SCHMMs du troisième niveau sont eux appris état par état. La séquence "mot de passe" est décodé. Chacun des segments est utilisé pour adapter un état du SCHMM. Le nombre d'états par modèle SCHMM est déterminé expérimentalement.

2.3. Phase de tests

Durant la phase de tests, un algorithme de Viterbi permet de confronter le signal d'entrée au modèle de mot de passe SCHMM. Le décodage Viterbi utilise l'ensemble des trames acoustiques du signal -parole et non-parole- mais le score final correspond à la somme des log-vraisemblances des seules trames de parole. Ce score est normalisé par celui obtenu avec le modèle UBM et les mêmes trames de parole.

2.4. Respect de la contrainte de ressources

Afin d'évaluer les ressources mémoire et le coût de calcul dus au système EBD, nous comparons celuici à des systèmes de référence. L'occupation mémoire est évaluée en considérant le nombre de paramètres stockés. En effet une telle application nécessite une compression et un codage spécifique des données stockées qui n'est pas pris en compte dans cet article. Dans ce contexte une estimation de la mémoire utilisée en terme de méga-octets n'est pas significative. Le coût de calcul est estimé d'après le nombre de calculs de log-vraisemblance pour une trame acoustique et une gaussienne monodimensionnelle. Comme le système EBD se situe à la frontière entre la reconnaissance du locuteur et de mots isolés, il convient de comparer les ressources utilisées à celles d'un système de reconnaissance du locuteur couplé à un système de reconnaissance de mots isolés fonctionnant en parallèle.

Reconnaissance du locuteur : Les deux premiers niveaux de l'architecture du système EBD présentent la même structure qu'un système de reconnaissance du locuteur classique. Cette partie du système a donc les mêmes caractéristiques en termes d'occupation mémoire et de coût de calcul qu'un système GMM/UBM standard. Il est alors possible d'économiser des ressources en partageant certains paramètres entre le modèle UBM et les différents GMM, ainsi qu'en utilisant les n-meilleures gaussiennes pour le calcul des scores. Pour un tel système à 128 gaussiennes par modèle, dont les vecteurs acoustiques ont 32 coefficients et en considérant 5 modèles de locuteurs, le nombre de paramètres stockés est de 28800. Pour chaque trame acoustique on calcule 24 576 log-vraisemblances. Un gain additionnel peut être obtenu en n'adaptant qu'une partie des moyennes de chaque modèle de locuteur.

Reconnaissance de la parole : Les systèmes état-de-l'art de reconnaissance de mots isolés utilisent des méthodes statistiques comme les Modèles de Markov cachés et les modèles de phonèmes. Deux approches sont donc comparées au système EBD.
La première utilise des modèles de phonèmes non-contextuels. Pour un système comprenant 108 états émetteurs, 128 gaussiennes par état et des vecteurs acoustiques de dimension 32, le nombre de paramètres stocké est approximé par :

$$nbem \times nbg \times \underbrace{(2 \times vectdim + 1)}_{unegaussienne} \qquad (1)$$

où *nbem* est le nombre d'états émetteurs, *nbg* est le nombre de distributions gaussiennes et *vectdim* est la dimension des vecteurs acoustiques. La log-vraisemblance toutes les distributions doit être calculée pour chaque trame, ce qui représente 13 824 log-vraisemblances.
La seconde approche consiste à modéliser chaque mot de passe par un HMM complet.La taille d'un modèle de mot de passe spécifique à un locuteur est estimée par :

$$nbet \times nbg \times \underbrace{(2 \times vectdim + 1)}_{unegaussienne} \qquad (2)$$

où nbet est le nombre d'états du HMM. Le coût de calcul est estimé par :

$$nbet \times nbg \times vectdim \qquad (3)$$

	Nombre de paramètres	Nombre de log-vraisemblances à calculer
Reconnaissance du locuteur + modèle à base de phonèmes	927, 360	442, 368
Reconnaissance du locuteur + HMM mot de passe	1, 276, 800	614, 400
EBD	**33, 600**	**24, 576**
Reconnaissance du locuteur	28, 800	24, 576[a]

[a]Le coût des somme pondéré est négligé.

Tab. 1: Estimation de la mémoire (en terme de paramètres) et du coût de calcul par vecteur acoustique (en terme de log-vraisemblances) du système EBD et de 3 systèmes de référence.

En ce qui concerne le système EBD, le nombre de paramètres supplémentaire stockés (juste quelques poids

pour chaque état des SCHMMs) est :

$$nbet \times nbg \times 1 \quad (4)$$

Le supplément de calcul du à la reconnaissance de parole dan le système EBD est uniquement du à des sommes pondérées, que l'on néglige devant le coût de calcul des log-vraisemblances. Le tableau 1 présente les ressources nécessaires aux 4 systèmes pour le stockage et le test de 5 locuteurs ayant enregistré chacun 2 mots de passe. Pour le systèmes HMM et EBD, les modèles de mot de passe ont 15 états. Les GMMs ont 128 gaussiennes et les vecteurs acoustiques 32 paramètres.

3. Évaluation sur MyIdea

3.1. La base de données MyIdea

Le système EBD a été testé sur les 30 hommes de la partie BIOMET de la base de données audio/vidéo MyIdea. Dans cette sous-partie de MyIdea, les locuteurs prononcent plusieurs groupes de phrases choisies pour leur contenu linguistique varié, lors de 3 sessions espacées dans le temps. Chaque locuteur prononce 25 phrases de longueurs variables. Parmi celles-ci, 10 phrase courtes et 2 phrases longues sont communes à tous les locuteurs. Ces trois sessions sont enregistrées dans un environnement à l'éclairage et au bruit contrôlé. La base MyIdea a été choisie dans le but d'intégrer à terme des informations vidéo dans le système.

3.2. Protocole expérimental

Étant donné le peu de locuteurs disponibles, les 30 hommes de la partie BIOMET sont séparés en deux groupes de 15, A et B qui servent tour à tour de données d'entraînement du modèle du monde et de clients/imposteurs.
Lorsque l'ensemble des enregistrements du groupe A est utilisé pour entraîner le modèle du monde, le groupe B est utilisé pour l'enrôlement et le test des locuteurs. Chacun des 15 locuteurs du groupe B est successivement considéré comme un client pour lequel les 14 autres locuteurs du groupe B sont des imposteurs. Deux conditions sont définies :

- *1-occ* :pour cette condition, trente modèles indépendants du texte sont entraînés pour chaque locuteur. Chacun de ces modèle est adapté à partir du modèle du monde en utilisant les deux phrases longues communes à tous les locuteurs (modèles indépendants du texte) ainsi que l'une des 10 phrases courtes communes de l'une des 3 sessions d'enregistrement (un seul exemple du mot de passe). Pour chaque modèle du locuteur X, la phrase courte utilisée pour l'adaptation de ce modèle est utilisée pour l'entraînement du modèle SCHMM d'un mot de passe. Cette procédure permet d'obtenir 900 modèles de mot de passe pour les deux groupes (10 phrases, 3 sessions, 30 clients).
- *2-occ* :pour cette condition, trente modèles indépendants du texte sont entraînés pour chaque locuteur. Chacun de ces modèle est adapté à partir du modèle du monde en utilisant les deux phrases longues communes à tous les locuteurs ainsi que 2 occurrences de l'une des 10 phrases courtes communes. Pour chaque modèle du locuteur X, les deux phrases courtes utilisées pour l'adaptation de ce modèle sont utilisées pour l'entraînement du modèle SCHMM d'un mot de passe. Cette procédure permet d'obtenir 900 modèles de mot de passe pour les deux groupes (10 phrases, 3 sessions, 30 clients).

Le nombre d'accès clients est dépendant de la condition utilisée. Pour chaque modèle de mot de passe, les tests clients sont réalisés en utilisant la ou les occurrences de la phrase courte modélisée qui n'ont pas servi à l'apprentissage du modèle. Le nombre d'accès client testés dans la condition *1-occ* est 1800 (900 mots de passe et 2 occurrences de test) ; et dans la condition *2-occ*, 900 tests clients sont effectués (450 mots de passe et 1 occurrence de test).
Trois configurations de tests imposteurs sont proposées. Le nombre de ces tests est dépendant de la condition utilisée et pour chaque locuteur les imposteurs utilisés sont les 14 autres locuteurs du groupe dont il est issu.

- MDP : dans cette configuration les imposteurs prononcent le contenu linguistique de la séquence utilisée pour l'entraînement du mot de passe testé. Chaque modèle de mot de passe est comparé à aux trois enregistrements de chacun des 14 autres locuteur du groupe dont est issu le locuteur.
- FAUX : dans cette configuration les imposteurs prononcent une phrase différente de celle utilisée pour l'apprentissage du mot de passe testé. Pour chaque session, une phrase est tirée au sort parmi les 9 phrases courtes restantes.
- TOUS : l'ensemble des tests imposteurs des configurations MDP et FAUX sont utilisés dans cette configuration.

Le nombre de tests des configurations MDP et FAUX sont identiques. Le nombre total de tests imposteurs effectués par configuration après rotation des groupes A et B, et rotation des 15 locuteurs au sein de ces groupes est :

- pour MDP et FAUX : 37 800 tests imposteurs ;
- pour la configuration TOUS : 75 600 tests imposteurs.

Pour chacune des trois configuration, le nombre de tests clients est 1800.

3.3. Paramétrisation

Des coefficients cepstraux (MFCC) sont calculés à une fréquence de 10ms. Un seuil d'énergie est appliqué aux trames pour séparer les trames *parole* des trames *non-parole*. Les trames sont des vecteurs de 32 dimensions : 15 coefficients cepstraux, la log-énergie et les coefficients Δ correspondants.

3.4. Resultats

Tous les résultats exposés dans cette partie sont comparés aux résultats d'un système GMM/UBM. Les paramètres utilisés pour ces expériences ont été fixés de façon expérimentale. La configuration retenue dans les expériences est la suivante :

- dimension des GMMs fixée à 128 gaussiennes ;
- 128 moyennes adaptées pour les modèles indépendants du texte ;
- 32 poids adaptés par état des modèles SCHMM.

Imposteurs	système GMM	Nombre d'états de l'EBD		
		5	10	15
TOUS	3,67	**3,55**	3,77	3,67
FAUX	2,78	2,28	2,00	**1,89**

Tab. 2: EER des systèmes GMM et EBD (avec différents nombres d'états) - condition d'apprentissage *1-occ* et tests imposteurs FAUX

Le principal avantage du système EBD comparé au GMM/UBM classique est d'intégrer une information provenant du mot de passe, sa structure temporelle.

Imposteurs	système GMM	Nombre d'états de l'EBD		
		5	10	15
TOUS	3,67	**3,55**	3,77	3,67
MDP	**4,39**	4,78	5,06	5,17

Tab. 3: EER des systèmes GMM et EBD (avec différents nombres d'états) - condition d'apprentissage *1-occ* et tests imposteurs MDP

Cet aspect est évalué par les expériences présentées dans les tableaux 2 et 3. Ces tableaux présentent les performances du système EBD en fonction du nombre d'états des modèles SCHMM et de la nature des tests imposteurs. Les performances, dans les mêmes conditions, du système GMM de référence ainsi que les résultats pour la configuration imposteurs TOUS sont présentées pour comparaison. Notons que le système GMM de référence est en fait équivalent à un système EBD dont les mots de passe auraient un seul état. Le tableau 2 (tests imposteurs FAUX) montre que lorsque les imposteurs ne connaissent pas le mot de passe du client, le système EBD obtient de meilleurs résultats que le GMM classique. La comparaison avec la configuration imposteurs TOUS montre également ce phénomène. L'augmentation du nombre d'états de 1 à 15 améliore les performances en termes d'EER.
Ce résultat n'est pas confirmé par le tableau 3 où les

Configuration	système GMM	Nombre d'états de l'EBD		
		5	10	15
1occ-TOUS	3,67	3,55	3,77	3,67
2occ-TOUS	**1,77**	**2,11**	2,33	2,33
1occ-MDP	4,39	4,78	5,06	5,16
2occ-MDP	**2,24**	2,89	3,22	3,33
1occ-FAUX	2,78	2,28	2,00	**1,89**
2occ-FAUX	0,91	0,57	0,56	**0,56**

Tab. 4: Comparaisons des EER obtenus avec une ou deux répétitions du mot de passe pour l'apprentissage dans différentes configurations de tests imposteurs

imposteurs connaissent le mot de passe des clients. On observe une perte de performances lorsque le nombre d'états augmente. Il semblerait que l'information structurelle du mot de passe masque l'information spécifique du locuteur ; *i.e.* on reconnaît le mot de passe et non le locuteur. Une augmentation de la quantité de données spécifique du locuteur durant la phase d'enrôlement, par exemple du nombre d'occurrences du mot de passe utilisées pour l'adaptation des modèles du locuteur pourrait permettre de résoudre ce problème. Afin de tester cette possibilité, nous proposons une expérience semblable aux précédentes mais utilisant deux occurrences du mot de passe pendant la phase d'apprentissage. Le tableau 4 présente les résultats des expériences utilisant une ou deux répétitions des mots de passe en fonction du nombre d'états des modèles et dans les différentes configuration de test. Comme espéré, l'augmentation des données d'entraînement améliore les résultats dans toutes les configurations. De plus, dans la configuration imposteurs FAUX, l'accroissement des données d'apprentissage profite plus au système EBD qu'au GMM, même si celui-ci s'avère plus efficace dans les autres configurations. Ce résultat semble indiquer que l'ajout de données d'apprentissage peut résoudre le problème de conflit entre information spécifique au locuteur et information liée à la structure du mot de passe.

4. Conclusions et travaux futurs

Nous avons présenté dans ce papier une nouvelle architecture acoustique de reconnaissance du locuteur basé sur l'utilisation de mots de passe pour des systèmes embarqués. L'approche proposée associe les avantages des systèmes indépendants du texte de type GMM/UBM et des systèmes HMM/Viterbi dépendants du texte. Elle répond aux contraintes des systèmes embarqués en minimisant l'espace mémoire et le nombre de calculs nécessaires. Les expériences menées ont montré que ce système est capable de reconnaître les mots de passe en utilisant très peu de données d'apprentissage. Ce système présente des performances égales aux systèmes GMM/UBM et dans certains cas supérieures à celui-ci.
Les travaux futurs se focaliseront sur la modélisation de la structure temporelle des mots de passe par la structure des SCHMMs. Il s'agira également d'incorporer à ce système une autre modalité, par exemple un flux vidéo afin d'améliorer sa capacité à résister aux impostures.

Références

[1] Frederic Bimbot, Jean-François Bonastre, Corinne Fredouille, Guillaume Gravier, Ivan Magrin-Chagnolleau, Sylvain Meigner, Teva Merlin, Javier Ortega-Garcia, Dijana Petrovska-Delacretaz, and Douglas A. Reynolds. A tutorial on text-independent speaker verification. *EURASIP Journal on Applied Signal Processing*, 4 :430–451, April 2004.

[2] Jean-François Bonastre, Philippe Morin, and Jean-Claude Junqua. Gaussian dynamic warping (gdw) method applied to text-dependent speaker detection and verification. In *European Conference on Speech Communication and Technology (Eurospeech)*, Geneva (Switzerland), 2003.

[3] Jean-François Bonastre, Nicolas Scheffer, Driss Matrouf, Corinne Fredouille, Anthony Larcher, Alexandre Preti Gilles Pouchoulin, Nicholas Evans, Benoît Fauve, and John S. Mason. Alize/spkdet : a state-of-the-art open source software for speaker recognition. In *Odyssey Conference*, 2008. http ://mistral.univ-avignon.fr/.

[4] Jean-Luc Gauvain and Chin-Hui Lee. Maximum a posteriori estimation for multivariate gaussian mixture observations of markov chains. In *IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP*, volume 2, pages 291–298, Adelaide (Australia), April 1994.

[5] Christophe Lévy, Georges Linares, Pascal Nocera, and Jean-François Bonastre. *Mobile Phone Embedded Digit-Recognition*, chapter 7 in Digital Signal Processing for In-Vehicle and Mobile Systems 2. Springer Sciences, 2006.

[6] Jiri Navratil, Upendra V. Chaudhari, and Stephane H. Maes. A speech biometrics system with multigrained speaker modeling. In *Conference for Natural Speech Processing*, 2000.

[7] Mark A. Przybocki, Alvin F. Martin, and Audrey N. Le. NIST speaker recognition evaluations utilizing the mixer corpora - 2004, 2005, 2006. *IEEE Transactions on Audio, Speech and Language Processing*, 15(7) :1951–1959, 2007.

[8] Steve J. Young. The general use of tying in phoneme-based hmm speech recognisers. In *IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP*, volume 1, pages 569–572, San Francisco (USA), 1992.