ACI Locale Floue pour la Reconnaissance du Locuteur en Combinant la Voix et les Mouvements des Lèvres

Abdenebi Rouigueb et Salim Chitroub

Laboratoire du Traitement du Signal et d'Image
Faculté d'Electronique et d'Informatique
U.S.T.H.B, Alger
Algerie

Ahmed Bouridane

School of Computing and Engineering and Information Sciences
Northumbria University
Pandon Building, Newcastle upon Tyne
Royaume Uni

RÉSUMÉ. ACI est une méthode statistique émergeante d'ordre supérieur qui peut être employée pour des applications diverses telles que la séparation de sources, la réduction de dimension et la représentation de données. Dans cet article, nous proposons un système de classification qui se déroule en deux étapes : (i) l'estimation de densité en utilisant la méthode ACI locale à base de clustering flou et (ii) l'ajustement du biais d'estimation par la classification SVM. Les tests de classification ont été effectués sur une base biométrique bimodale (audio et vidéo) où les vecteurs sont obtenus par la fusion des paramètres des mouvements des lèvres et de la voix. Les résultats de classification obtenus montrent une amélioration par rapport au classificateur SVM standard.

ABSTRACT. ICA is an emergent method which uses high-order statistics and it can be used in many applications such as source separation, dimension reduction and data representation. This work makes use of ICA and proposes a new classification scheme spreading over two stages: (i) density estimation using local ICA based on fuzzy clustering and (ii) correction of the estimation bias using SVM classification. The classification experiments are carried out over a bimodal biometric dataset where vectors are obtained by the fusion of lip movement and acoustic features. The results show an improvement in classification rate than the standard SVM classifier.

MOTS-CLÉS : ACI locale, clustering flou, mouvement des lèvres, reconnaissance du locuteur.

KEYWORDS: Local ACI, fuzzy clustering, lip movement, speaker recognition.

1. Introduction générale

La méthode de l'analyse en composantes indépendantes (ACI) locale a été proposée par Karhunen et *al.* [8] pour la représentation de données ; son principe consiste au partitionnement des données en clusters homogènes, puis à l'association d'un modèle ACI linéaire à chaque cluster. L'utilisation de l'algorithme K-Means a été illustrée dans [8].

En pratique, le clustering flou, où un vecteur peut appartenir à plusieurs clusters avec des degrés différents, s'est montré plus conforme à la réalité que le clustering conventionnel où un vecteur appartient à un seul cluster. Dans ce travail, nous proposons un système de classification en utilisant la méthode ACI locale à base de clustering flou. Le présent travail a deux objectifs, à savoir : la comparaison des performances des combinaisons possibles d'un ensemble de méthodes ACI avec quelques algorithmes du clustering flou et l'amélioration du taux de classification.

Les tests de classification ont été effectués sur une base biométrique bimodale intégrant la voix et mouvement des lèvres. Une phrase audio-visuelle est divisée en séquences de fenêtres temporelles de courte durée ; un vecteur de paramètres est extrait de chaque fenêtre. Les résultats de la classification des vecteurs de paramètres extraits sont comparés avec ceux du classificateur traditionnel SVM (Support Vector Machine).

La section suivante présente une introduction de la méthode ACI locale. Section 3 explique les différents stages du système de la classification proposé. Section 4 est consacrée à la présentation des paramètres biométriques sélectionnés ainsi que le processus de fusion. Les résultats de l'expérimentation sont présentés en section 5. Une dernière section conclura cet article.

2. La méthode ACI locale floue

2.1. ACI linéaire

L'ACI [2][7] est une méthode basée sur les statistiques d'ordre supérieur ; utilisée initialement pour résoudre le problème de séparation aveugle des sources (BSS) [2]. L'objectif de l'ACI est d'exprimer un vecteur de signaux observés sous forme d'un mélange d'un ensemble de composantes indépendantes (CIs) en maximisant leur indépendance statistique. L'ACI standard consiste à un mélange linéaire et instantané :

$$\mathbf{x}(t) = \boldsymbol{A} \times \mathbf{s}(t) \tag{1}$$

avec $\boldsymbol{A}$ une matrice de mélange $N \times M$ de rang complet, $\mathbf{x}(t) = \big(x_1(t), \dots, x_N(t)\big)^T$ et $\mathbf{s}(t) = \big(s_1(t), \dots, s_M(t)\big)^T$ sont respectivement les vecteurs des signaux observés et des CIs à l'instant t. M est généralement supposé égal à N parce que $M = N$ est la valeur

maximale pour que les méthodes ACI du modèle (1) convergent en utilisant seulement l'hypothèse de l'indépendance statistique.

2.2. ACI non-linéaire

Bien que le modèle ACI linéaire ait donné des performances significatives dans de nombreuses applications pratiques, il n'est pas performant pour décrire des données ayant des distributions générales. En effet, le vecteur des variables observées $\mathbf{x}$ souvent dépend non-linéairement des variables sources $\mathbf{c}$, $\mathbf{x} = F(\mathbf{c})$. Dans cette étude, nous ne s'intéressons pas au calcul des sources comme au cas de BSS, mais plutôt nous voulons une représentation adéquate des données observées $\mathbf{x}$ par un mélange non-linéaire

$$\mathbf{x} = G(\mathbf{s}) \tag{2}$$

où $G : R^M \to R^N$ est une fonction de mélange non-linéaire et $\mathbf{s}$ est le vecteur des CIs.

Les méthodes ACI usuelles traitent la non-linéarité de deux manières différentes : l'approche basée sur les réseaux de neurones [12][1] et l'approche basée sur la combinaison des modèles ACI linéaires [8]. La deuxième approche procède par la partition des données en clusters. Par la suite, un modèle ACI linéaire est appliqué localement sur chaque cluster. Il est important de rappeler que le clustering est une tâche délicate qui ne converge pas souvent vers la même solution car il dépend de l'initialisation et par conséquent il influence fortement les résultats. En ACI locale, le but du clustering est de produire des clusters ayant une grande vraisemblance d'être approximé par un modèle ACI linéaire. Nous avons opté pour la deuxième approche afin de profiter de sa simplicité d'implémentation et de sa capacité prouvée de généralisation.

L'algorithme de clustering K-Means qui détecte des clusters sphériques n'est pas toujours approprié pour l'extraction des CIs, cependant il est préférable de partitionner les données en un petit nombre de clusters linéaires ayant des grandes vraisemblances d'être issus de mélanges ACI linéaires. Dans le présent travail, nous proposons d'expérimenter deux algorithmes populaires du clustering flou : Gustafson-Kessel (GK) [6] et Fuzzy C-Means (FCM) [3]. Le FCM détecte les clusters sphériques, alors que le GK, qui est une généralisation de FCM, peut détecter les clusters de forme elliptique.

L'algorithme GK est basé sur la minimisation de la fonction objectif suivante :

$$J(X; \boldsymbol{U}; \boldsymbol{V}; \boldsymbol{A}) = \sum_{i=1}^{L} \sum_{j=1}^{K} \left(u_{ij}\right)^m D_{ij\boldsymbol{A}_i}^2 \tag{3}$$

avec L le nombre de clusters, K le nombre de vecteurs de données, m le degré de fuzzification, u_{ij} la probabilité d'appartenance du vecteur j au cluster i et $D_{ijA_i}^2$ la distance entre le vecteur j et le centre du cluster i, basée sur la métrique $\boldsymbol{A}_i$, propre à chaque cluster et se calcule comme suit :

$$D_{ijA}^2 = \left(x_j - v_i\right)^T \boldsymbol{A}_i \left(x_j - v_i\right) \tag{4}$$

où v_i est le centre du cluster i. Les paramètres d'optimisation sont $\boldsymbol{U}, \boldsymbol{V}, \boldsymbol{A}$. Le choix du nombre de clusters est un problème complexe. Dans ce travail, l'estimation de Kest faite par la minimisation de l'indice Xie et Beni (XB) développé spécialement pour la validation du clustering flou [13]. La formulation du FCM est un cas spécial du GK où la métrique $\boldsymbol{A}_i$ dans (3) et (4) est prise égale à la matrice identité **I** pour tous les clusters.

3. Système de classification

Nous utilisons la méthode ACI locale floue pour la construction d'un estimateur de densité par classe. Pour le cas linéaire $X = \boldsymbol{A} \times S$, en utilisant la formule du Jacobien de la transformation des variables, la conjointe P_X peut se simplifier comme suit

$$P_X(X) = \left(\frac{1}{|det\ (Jac(X))|}\right) P_S(\boldsymbol{A}^{-1} \times X) \quad (5)$$

L'avantage d'une telle transformation est de pouvoir factoriser la conjointe P_S car les CIs sont estimées tout en maximisant l'indépendance statistique entre elles, donc

$$P_X(X) = \left(\frac{1}{|det\ \boldsymbol{A}|}\right) \times \prod_{i=1}^{N} P_{S_i}(S_i) \quad (6)$$

L'estimation de P_X est simplifiée davantage et revient alors à l'estimation des densités des variables monodimensionnelles $S_i\ , i = 1, \ldots, N$. Ceci peut être fait par des outils dédiés tels que l'estimateur de Parzen.

Dans le cas non-linaire (cas général), P_X peut être déduite par la marginalisation sur les estimations partielles des clusters comme suit :

$$P_X(X) = \sum_{j=1}^{L} P_X(X, C_j) = \sum_{j=1}^{L} P_C(C_j) \times P_X(X/C_j) \quad (7)$$

$P_C(C_j)$ est la probabilité a priori du cluster C_j, $P_X(X/C_j)$ peut se calculer en utilisant (6) du modèle ACI linéaire local associé au cluster C_j.

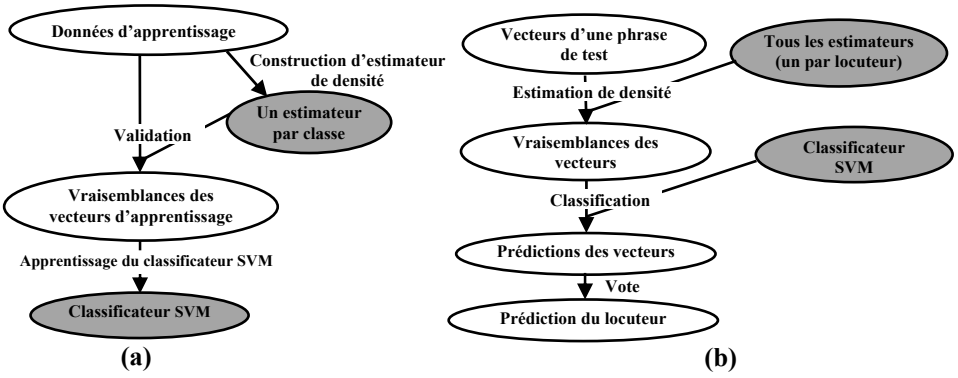

Figure 1. *Etapes de construction du classificateur, (a) apprentissage, (b) prédiction.*

Une classe représente un seul locuteur et pour chaque classe $\omega_i, i = 1, \dots Nc,$ un estimateur de densité π_i est construit en utilisant les vecteurs de données d'apprentissage. Les estimateurs $\pi_i,\ i = 1, \dots, Nc$ sont souvent biaisés différemment parce que chaque π_i est construit séparément en utilisant seulement les données de sa classe i. Par conséquent, le classificateur de maximum de vraisemblance n'est pas approprié. Pour rectifier le biais d'estimation, nous proposons de procéder à une classification supervisée discriminative telle que SVM au niveau de l'espace des vraisemblances des vecteurs. Soit $\pi_i(x)$ la vraisemblance que le vecteur $\mathbf{x}$ soit dans la classe ω_i, les attributs de la classification SVM sont $\pi_1(x)$,…., $\pi_{Nc}(x)$. Les étapes de la classification sont illustrées dans les diagrammes de la figure 1.

4. Extraction des paramètres

La base de données biométrique bimodale BOMIO (audio et vidéo) [9] a été utilisée pour les tests de prédiction. Elle comporte les enregistrements en anglais de 152 locuteurs avec 12 sessions chacun, collectés dans des pays différents. Les séquences audio-visuelles sont enregistrées à l'aide d'un téléphone portable NOKIA N93i.

4.1. Le traking des mouvements des lèvres

Les images sont converties de l'espace de couleurs RGB à YCbCr. L'espace YCbCr subdivise l'espace RGB à la composante de luminance Y, et les composantes de chrominance, Cb et Cr. Seulement Cb et Cr ont été employées car elles ont montré une bonne discrimination entre couleurs de peau et de lèvres. Lors du traking, on est intéressé au calcul de la plus Petite Fenêtre Rectangulaire (PFR) englobant les lèvres. La détection de la PFR est faite automatiquement en exécutant les étapes suivantes :

– La localisation d'une fenêtre d'intérêt plus grande et qui entoure suffisamment la PFR de l'image précédente, la taille de cette fenêtre est choisie de telle sorte qu'elle couvre les éventuels mouvements des lèvres et sans qu'elle atteigne les autres éléments du visage tels que le nez ou l'oreille (voir exemple en Figure 2.a) ;

– La segmentation en utilisant un seuil adaptatif (Figure 2.b) ;

– L'extraction de la plus grande surface blanche connexe et la détection du plus petit rectangle $u \otimes w$ contenant les lèvres (Figure 2.c), PFR nouveau= rectangle $u \otimes w$.

Pour la caractérisation des mouvements des lèvres, nous avons défini des paramètres globaux définissant à la fois la géométrie et la texture des lèvres. Ces paramètres sont respectivement : u la largeur de la bouche , w la hauteur de la bouche , s la surface des lèvres (nombre de pixels blancs sur Figure 2.c), la moyenne et l'écart-type de la PFR de taille $u \times w$. Les cinq paramètres choisis varient d'une personne à une autre et pour la même personne ils varient différemment selon les phonèmes prononcés.

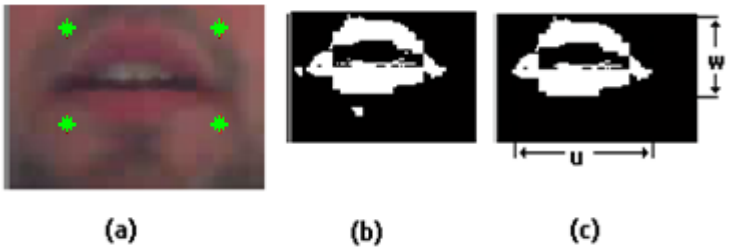

Figure 2. *Extraction des paramètres des lèvres.*

4.2. La voix

Nous optons pour l'application des coefficients MFCC (*Mel-Frequency Cepstral Coefficients*) pour la reconnaissance du locuteur en mode *indépendant du texte*, car ils ont montré une bonne discrimination entre locuteurs dans de nombreux travaux [10]. Le calcul des MFCC est fait en choisissant une taille de fenêtre égale à 25 ms et un pas d'avancement de 6.25 ms. Nous retenons seulement 12 MFCC. L'énergie qui ne représente pas une variable discriminante est écartée. Par conséquent, nous obtenons pour chaque phrase une séquence de vecteurs MFCC comportant chacun 12 MFCC.

4.3. Fusion

La fusion est réalisée par la concaténation des paramètres des lèvres avec les MFCC. La fréquence du signal vocal est de 160 trames/sec alors qu'elle est de 30 images/sec pour la vidéo. De ce fait, les deux signaux ont besoin d'être synchronisés. Pour chaque vecteur labial, le plus proche vecteur acoustique par rapport à l'ordre chronologique lui est ajouté, ainsi on aura 30 vecteurs de fusion par seconde. Les paramètres ne sont pas normalisés avant la fusion car les paramètres des lèvres ont une grande variance et nous voulons des clusters qui dépendent plus des lèvres que des MFCC, dans le but d'avoir un cluster pour chaque configuration de la bouche.

5. Expérimentation

Un échantillon de 10 locuteurs est tiré aléatoirement du corpus BOMIO pour l'expérimentation. 05 phrases d'apprentissage et 05 phrases du test sont prises aléatoirement pour chaque personne. Les vecteurs de données extraits comportent 17 composantes (12 MFCC du signal vocal et 05 paramètres des lèvres). La distribution des nombres de vecteurs d'apprentissage/test est donnée dans le tableau 1.

Classe	1	2	3	4	5	6	7	8	9	10
Apprentissage	3413	1884	1505	1709	1010	1987	1876	1628	1676	1384
Test	1279	1463	1495	1184	944	1338	1200	963	1112	848

Tableau 1. *Nombre de vecteurs d'apprentissage et du test par classe.*

Les résultats obtenus sont comparés à ceux du SVM (appliqué directement sur les vecteurs de données) où les fonctions noyaux, linéaire, polynomial, Gaussien et Laplace ont été expérimentées. On a réalisé les différentes combinaisons des méthodes ACI, FastICA [7], JADE [4], KDICA [5] et FastKernelICA [11], avec les algorithmes du clustering flou GK et FCM. En effet, il parait logique que les performances des méthodes ACI linéaires varient, même légèrement, en fonction de la forme de clusters.

		Voix et lèvres		Uniquement la voix		Uniquement lèvres	
	ACI linéaire	GK	FCM	GK	FCM	GK	FCM
Approche proposée	FastICA	0.7623	0.7648	**0.2431**	0.2391	0.7068	0.7160
	JADE	0.7650	0.7658	0.2407	0.2389	0.7029	0.7118
	FastKernelICA	0.7435	0.7344	0.2195	0.2244	0.7100	0.6988
	KDICA	**0.7726**	0.7710	0.2418	0.2380	0.7146	**0.7170**
SVM		**0.7245** (Pol, deg=3)		**0.2471** (Gaus, σ=0.035)		**0.6783** (Gaus, σ=0.002)	

Tableau 2. *Taux de classification, méthode proposée est comparée avec SVM.*

La courbe ROC (*Receiver Operating Characteristics*) ou l'indicateur EER (*Equal Rate Error*) sont souvent utilisés pour l'évaluation des performances en biométrie. A raison du nombre relativement petit des locuteurs considérés (10), l'EER de reconnaissance des phrases en appliquant une stratégie de vote est aussi bien inférieur à 1 % pour le SVM standard que pour notre méthode. Donc, nous proposons de mener l'évaluation en termes du taux de classification des vecteurs de paramètres. Le meilleur taux de fusion obtenu par notre méthode est 77.26 % en combinant KDICA et GK, cela est supérieur au meilleur taux SVM 72.45 %, à noter que SVM a été appliquée comme une étape de poste-traitement dans l'approche proposée.

6. Conclusion

Nous avons proposé une méthode de classification en utilisant la méthode ACI locale à base du clustering flou. La classification se déroule en deux étapes : (i) l'estimation de densité des vecteurs de données, et puis (ii) la classification SVM des vraisemblances issues de l'étape précédente. Notre méthode a été appliquée pour la classification des vecteurs d'une base de données biométrique intégrant la voix et les mouvements des lèvres du locuteur. Chaque phrase est divisée en une séquence de fenêtres temporelles de même taille, puis un vecteur est extrait de chaque fenêtre par la

fusion des paramètres de la voix et des lèvres. Les résultats obtenus montrent une amélioration appréciable par rapport à SVM au cas de fusion et au cas de lèvres uniquement, les données dans ces deux cas sont hétérogènes, elles n'ont pas la même échelle. Nous constatons également que GK et FCM ont donné des performances sensiblement similaires. L'avantage de la méthode proposée est qu'elle permet à l'utilisateur de choisir parmi un ensemble riche des méthodes ACI et des algorithmes du clustering. Pour une base de données quelconque, parfois on peut trouver des combinaisons (ACI et Clustering) qui donnent des performances significatives.

7. Bibliographie

[1] L.B. Almeida, « Linear and nonlinear ICA based on mutual information: the MISEP method », *Signal Processing*, vol. 84, no. 2, pp. 231–245, 2004.

[2] S. Amari, A. Cichocki et H. Yang, « A new learning algorithm for blind signal separation », *Advances in Neural Information Processing Systems*, vol. 8, pp. 757–763, 1996.

[3] J.C. Bezdek, « *Pattern recognition with fuzzy objective function algorithms* », Plenum Press, New York, 1981.

[4] J.F. Cardoso et A. Souloumiac, « Blind beamforming for non Gaussian signals », *IEEE Proceedings-F*, vol. 140, no. 6, pp. 362-370, 1993.

[5] A. CHEN, « Fast kernel density independent component analysis », *In ICA '06; Lecture Notes in Computer Science*, Springer, Berlin, vol. 3889, pp. 24-31, 2006.

[6] D.E. Gustafson et W.C. Kessel, « Fuzzy clustering with fuzzy covariance matrix », *Proc. of the IEEE CDC*, San Diego, pp. 761-766, 1979.

[7] A. Hyvrinen, J. Karhunen et E. Oja, « *Independent component analysis* », JOHN WILEY & SONS, INC, 2001.

[8] J. Karhunen, S. Malaroiu et M. Ilmoniemi, « Local independent component analysis using clustering », *Proc. Workshop on ICA and Signal Separation (ICA'99)*, pp. 43-48, 1999.

[9] http://www.mobioproject.org/, dec, 2011.

[10] D.A Reynold, « Speaker identification and verification using Gaussian mixture speaker models », *Speech Comm*, vol. 17, pp. 91-108, 1995.

[11] H. Shen, S. Jegelka et A. Gretton, « Fast Kernel-Based Independent Component Analysis », *IEEE Transactions on Signal Processing*, vol. 57, no. 9, pp. 3498-3511, 2009.

[12] E.F. Simas Filho, J.M. de Seixas et L.P Calôba, « Modified post-nonlinear ICA model for online neural discrimination », *Neurocomputing*, vol. 73, pp. 2820-2828, 2010.

[13] X.L. Xie et G. Beni, « A validity measure for fuzzy clustering », *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 13, no. 8, pp. 841-847, 1991.