

COMPARAISON DES PERFORMANCES DE DEUX METHODES DE CLASSIFICATION ALGORITHME DES CENTRES MOBILES RESEAUX NEURO-MIMETIQUES MULTICOUCHES.

I. COUSTAUX, J.F. TROUILHET, J.P. GUILHOT

Laboratoire d'Acoustique de Métrologie et d'Instrumentation
38, rue des 36 Ponts 31400 TOULOUSE -FRANCE-

RÉSUMÉ

Nous allons présenter la comparaison de deux méthodes de classification utilisées pour la reconnaissance automatique de signature acoustique. Ces deux méthodes sont : l'algorithme des centres mobiles et un réseau neuromimétique multicouche. Nous avons testé leurs performances respectives en terme de probabilités de détection et de fausse alarme sur trois types de signaux différents.

I - INTRODUCTION

Le choix d'une méthode de classification reste une étape très importante face à un problème de reconnaissance de signature acoustique. C'est pourquoi nous nous sommes efforcé de réaliser une comparaison des performances des deux méthodes suivantes :

- algorithme des centres mobiles,
- réseaux neuromimétiques multicouches.

Afin d'évaluer les performances de ces deux méthodes de classification, nous avons procédé en deux étapes. Premièrement, nous avons utilisé deux variétés de signaux simulés puis, dans un deuxième temps, nous avons utilisé un signal biologique particulièrement difficile à reconnaître de manière automatique. En effet, ce signal, obtenu dans un environnement bruyant, est non stationnaire et non reproductible entre deux événements dont la date d'apparition est aléatoire. Bien entendu, ces signaux ont été analysés puis modélisés avant d'être "traités" par les classificateurs.

Nous allons tout d'abord, décrire succinctement les deux méthodes de classification précitées puis, les différents signaux utilisés ainsi que les modélisations appliquées aux signaux réels. Avant de conclure, les performances seront données essentiellement en termes de probabilités conditionnelles de reconnaissances et en taux de réussite.

ABSTRACT

In this article we will show the comparison between two classification methods used in acoustic signature automatic recognition. These two methods are the moving centres agregation algorithm and the feed forward network. We have tested their performances according to their detection and false detection probability on three different signals.

II - PRESENTATION DES METHODES DE CLASSIFICATION

1) Algorithme des centres mobiles

Cette première méthode [1] a pour but de réaliser une partition de l'ensemble étudié. Pour cela, il faut se donner au départ le nombre de classes et une partition initiale. Cette dernière est obtenue soit grâce à une connaissance à priori des classes soit par une répartition au hasard des individus dans les diverses classes.

A ces fins on exécute les opérations suivantes :

- (1) détermination du centre de gravité de chaque classe,
- (2) réaffectation de chaque individu à la classe dont le centre de gravité est le plus proche au sens de la distance Euclidienne,
- (3) on recommence depuis le départ tant que des modifications dans la composition des classes interviennent.

N'ayant pas la certitude d'obtenir la meilleure solution, il est préférable d'exécuter plusieurs fois l'algorithme complet avec des partitions initiales différentes. On retient alors la réalisation qui minimise le moment interclasses.

Afin de diminuer le temps de convergence de cet algorithme, on peut utiliser la variante suivante pour la création de la partition initiale :

- (1) on prend pour la classe n° i un individu au hasard,
- (2) la classe suivante sera obtenue en prenant l'élément le plus éloigné du précédent



(3) on recommence l'étape 2 tant que des classes sont vides,

(4) les individus restant sont alors affectés à la classe la plus proche.

On peut alors commencer l'algorithme des centres mobiles proprement dit. Cette méthode fonctionne bien pour peu que l'on ait légèrement surestimé le nombre de classes.

2) Réseau neuromimétique multicouche

Comme deuxième méthode de classification nous avons choisi d'utiliser un réseau neuromimétique [2] à trois couches aussi appelé "perceptron multicouche non bouclé" (feed-forward networks). Celui-ci comporte donc, outre sa couche d'entrée, deux couches de poids modifiables.

L'utilisation de ce type de réseau de neurones se décompose en deux parties : la phase d'apprentissage et la phase de généralisation. Lors de la première phase, on effectue un apprentissage supervisé de manière à adapter les poids au problème posé. Nous avons utilisé l'algorithme de rétro-propagation du gradient stochastique pour modifier les poids. Les principales équations relatives à cet algorithme sont :

Soit une base d'apprentissage $\{X^{(l)}, Y^{(l)}, 1 \leq l \leq N\}$

- $X^{(l)} = (X_1^{(l)} \dots X_n^{(l)})$ un vecteur d'entrée,

- $Y^{(l)} = (Y_1^{(l)} \dots Y_m^{(l)})$ un vecteur de sortie désirée,

- $S^{(l)} = (S_1^{(l)} \dots S_m^{(l)})$ un vecteur de sortie effectivement obtenu,

- w est la matrice des poids de la couche intermédiaire,

- W est la matrice des poids de la couche de sortie,

- f est la fonction d'activation d'un neurone et f' sa dérivée

On cherche à minimiser l'erreur obtenue suite à un passage complet de la base d'apprentissage : $E = \sum_1 E^{(l)}$

Ce qui revient à minimiser l'erreur globale pour chaque élément de la base d'apprentissage :

$$E^{(l)} = \frac{1}{2} \sum_k (S_k^{(l)} - Y_k^{(l)})^2$$

Pour cela, on va modifier les poids du réseau en utilisant une méthode du gradient de pas μ ce qui donne :

- pour les poids de la couche de sortie :

$$\Delta W_{kj} = \mu (S_k^{(l)} - Y_k^{(l)}) f' \left(\sum_j W_{kj} f \left(\sum_i w_{ji} X_i^{(l)} \right) \right) f \left(\sum_i w_{ji} X_i^{(l)} \right)$$

- pour les poids de la couche intermédiaire :

$$\Delta w_{ji} = \mu \sum_k \left[(S_k^{(l)} - Y_k^{(l)}) W_{kj} f' \left(\sum_j W_{kj} f \left(\sum_i w_{ji} X_i^{(l)} \right) \right) \right] f' \left(\sum_i w_{ji} X_i^{(l)} \right) X_i^{(l)}$$

i est l'indice relatif à la couche d'entrée, j celui relatif à la couche intermédiaire et k celui correspondant à la couche de sortie.

Nous avons utilisé la notion de biais qui n'apparaît pas dans ces équations. Comme les biais ont exactement le même comportement que les poids, on peut utiliser les équations à condition de considérer que l'entrée correspondante vaut toujours un.

Le nombre de neurones par couche sera adapté à chaque problème que nous allons présenter au réseau. Pour la couche d'entrée, il y a autant de neurones que de paramètres retenus pour décrire les individus. Au niveau de la couche cachée, on prendra généralement entre quatre et cinq fois le nombre de neurones de la couche d'entrée. Pour la couche de sortie, on a pris un neurone par classe (1 signifie appartient à la classe, 0 n'appartient pas à la classe) et non un neurone pour deux classes (1 pour une classe et -1 pour l'autre).

Le fait d'utiliser un neurone par classe permet d'introduire la notion de rejet. En effet le réseau peut ainsi estimer qu'un individu appartient à deux classes différentes ou à aucune d'entre elles, ce qui donne une information supplémentaire. Les individus correspondant aux cas précédent sont évidemment considérés comme hors classe.

Lors de la phase d'apprentissage du réseau, tous les individus de la base d'apprentissage sont présentés en entrée du réseau de manière aléatoire. Ceci a pour but de diminuer les risques de convergence vers un minimum local. De plus, pour diminuer le temps que nécessite cette phase, nous avons limité les cas de calculs de la rétro-propagation quant l'erreur commise pour un individu est très faible. Ceci permet généralement de gagner plus d'un facteur trois sur le temps que demanderait normalement cette étape.

III- DESCRIPTION DES SIGNAUX

1) Les signaux simulés

Nous avons utilisé des individus à deux paramètres afin de faciliter la représentation graphique. Les classes des signaux simulés vont alors correspondre à la répartition spatiale de leurs coordonnées.

Premièrement des classes sphériques de répartition Gaussienne ont été considérées. L'ensemble simulé comporte cinq classes réparties "comme sur un dé". Le bord des classes se chevauche. Une représentation est donnée sur la figure 1.

Cette répartition a été simulée car elle représente une configuration de prédilection pour l'algorithme des centres mobiles.

Notre deuxième type de représentation est constitué de deux classes non séparables linéairement "en forme de virgule" qui s'entrecroisent. Une représentation est donnée sur la figure 2.

2) Les signaux réels

Le signal que nous avons à reconnaître de manière automatique est le son produit par l'aloise (Aloisa aloisa

L.) lors de sa ponte à la surface de l'eau [1]. La surveillance du nombre de ponte a pour but d'effectuer le recensement de la population de ces poissons migrateurs de la Garonne.

Le recueil de ces signaux est assuré par un microphone disposé sur les berges du fleuve en période de fraie. Les sons provoqués par les poissons sont difficiles à reconnaître car leur date d'apparition est aléatoire, ils sont non stationnaires et ils sont souvent bruités par des bruits de véhicules routiers, d'oiseaux, de grenouilles...

Une représentation temporelle du signal est donnée sur la figure 3.

Suite à l'étude temps fréquence du signal produit par l'alose (figure 4), nous avons déterminé un spectre moyen qui est à la base d'une des techniques de modélisation employée.

Nous avons premièrement extrait l'énergie moyenne dans trois bandes de fréquence, ce qui nous donne des individus définis par trois paramètres. Dans un deuxième temps nous avons effectué le calcul des premiers coefficients du modèle auto régressif (AR) du signal. Ceci donne des individus représentés par huit paramètres.

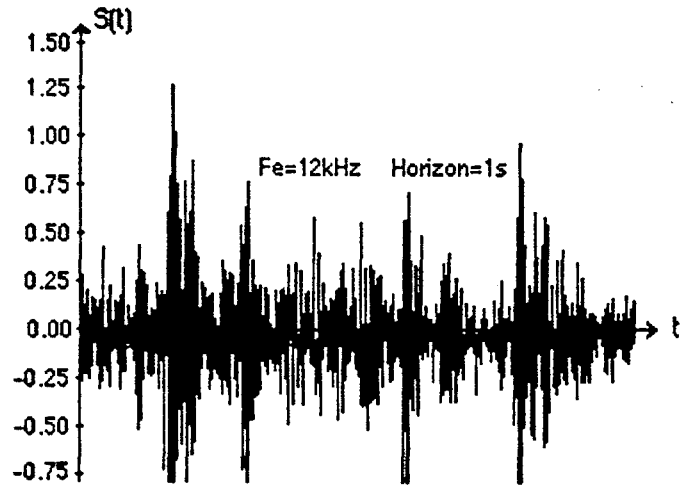


figure 3 : représentation temporelle du signal produit par l'alose

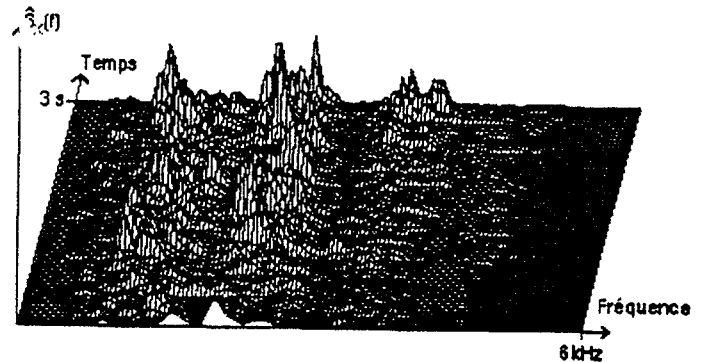


figure 4 : évolution spectrale du signal produit par l'alose

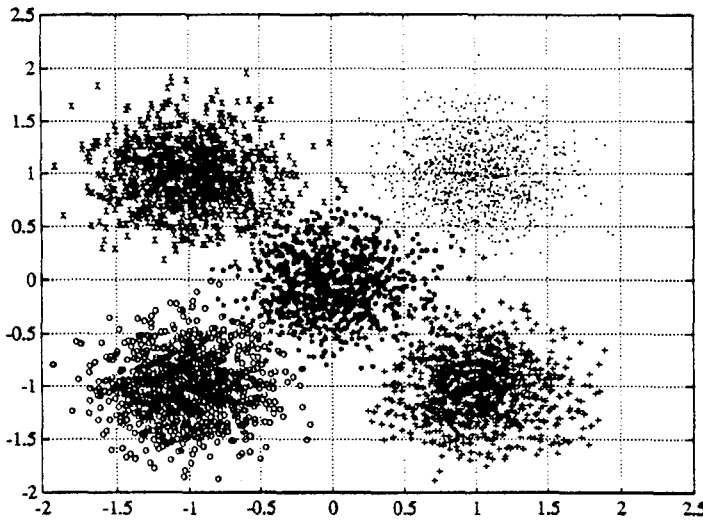


figure 1 : 1^{er} type de signaux simulés

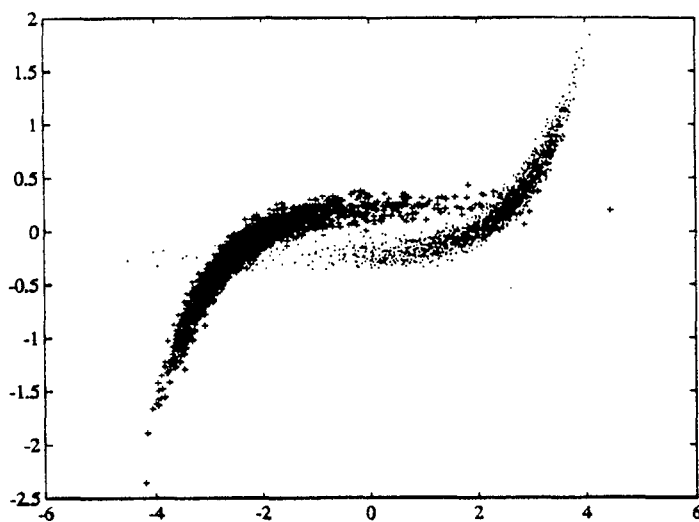


figure 2 : 2^{ème} type signaux simulés

IV - DESCRIPTION DES TESTS ET RESULTATS

Relativement à une classe i , l'univers des événements se divise en deux parties. Celle contenant les individus de la classe (x_i) et celle ne les contenant pas (\bar{x}_i). Par contre, le classificateur va créer deux zones légèrement différentes. L'une où il a déterminé la présence de la classe (dx_i) et l'autre où il ne l'a pas déterminée ($d\bar{x}_i$). Donc si on pose :

- $Nbr_dx_i - x_i$ = le nombre d'individus affectés à la classe i sachant qu'ils appartiennent à cette classe,
- $Nbr_d\bar{x}_i - x_i$ = le nombre d'individus non affectés à la classe i sachant qu'ils appartiennent à cette classe,
- Nbr_Total = le nombre total d'individus classés.

On obtient :

$$- P(dx_i \cap x) = \frac{Nbr_dx_i - x_i}{Nbr_Total}$$

$$- P(d\bar{x}_i \cap x_i) = \frac{Nbr_d\bar{x}_i - x_i}{Nbr_Total}$$

$$- P(x_i) = \frac{Nbr_dx_i - x_i + Nbr_d\bar{x}_i - \bar{x}_i}{Nbr_Total}$$

$$- P(\bar{x}_i) = \frac{Nbr_dx_i - \bar{x}_i + Nbr_d\bar{x}_i - \bar{x}_i}{Nbr_Total}$$



Bien que :

$$\text{taux de réussite : } T_R = P(dx_i \cap x_i) + P(d\bar{x}_i \cap \bar{x}_i)$$

$$\text{taux d'échec : } T_E = P(dx_i \cap \bar{x}_i) + P(d\bar{x}_i \cap x_i)$$

ces résultats peuvent masquer ce qui nous intéresse lorsque les probabilités $p(x_i)$ et $p(\bar{x}_i)$ sont très différentes. Il vaut mieux considérer les probabilités conditionnelles :

$$P(dx_i / x_i) = \frac{P(dx_i \cap x_i)}{p(x_i)} \quad \text{probabilité de détection,}$$

$$P(dx_i / \bar{x}_i) = \frac{P(dx_i \cap \bar{x}_i)}{p(\bar{x}_i)} \quad \text{probabilité de fausse alarme,}$$

$$P(d\bar{x}_i / x_i) = \frac{P(d\bar{x}_i \cap x_i)}{p(x_i)} \quad \text{probabilité de non détection,}$$

$$P(d\bar{x}_i / \bar{x}_i) = \frac{P(d\bar{x}_i \cap \bar{x}_i)}{p(\bar{x}_i)}$$

En fait deux de ces probabilités suffisent à résumer les performances du classificateur car on a :

$$P(dx_i / x_i) + P(d\bar{x}_i / x_i) = 1 \quad \text{et} \quad P(dx_i / \bar{x}_i) + P(d\bar{x}_i / \bar{x}_i) = 1$$

Pour chaque type de signaux on a utilisé deux bases différentes. Une pour la phase d'apprentissage et une pour la phase de généralisation.

Pour les premiers signaux simulés on obtient :

alg. des centres mobiles			réseau de neurones		
T_R	$P(dx_i / x_i)$	$P(d\bar{x}_i / \bar{x}_i)$	T_R	$P(dx_i / x_i)$	$P(d\bar{x}_i / \bar{x}_i)$
99,8	99,5	0,1	99,8	99,6	0,1
99,7	99,4	0,1	99,7	99,3	0,2
99,6	98,8	0,2	99,6	99,4	0,3
99,7	99,4	0,2	99,6	99,5	0,3
98,8	96,7	0,7	98,8	95,6	0,3

Pour les deuxièmes signaux simulés on obtient :

alg. des centres mobiles			réseau		
T_R	$P(dx_i / x_i)$	$P(d\bar{x}_i / \bar{x}_i)$	T_R	$P(dx_i / x_i)$	$P(d\bar{x}_i / \bar{x}_i)$
95,2	95,2	4,7	99,3	99,0	0,5
95,2	95,3	4,8	99,3	99,4	1,0

Pour les signaux de l'alose nous avons utilisé deux fichiers différents, un contenant les signaux de l'alose bien distincts des autres bruits (fichiers en 1) et l'autre contenant des signaux d'alose qui sont très bruités (fichiers en 2). Nous ferons également la différence entre la modélisation en énergie (fichiers en EN) et la modélisation AR (fichiers en AR).

Les fichiers en 12 sont les bases d'apprentissage qui contiennent des signaux d'alose bruités et non bruités ainsi que des bruits.

Pour les fichiers en 1 : $P(\text{Alose})=50\%$ $P(\text{Bruit})=10\%$,
pour les fichiers en 2 : $P(\text{Alose})=12\%$ $P(\text{Bruit})=65\%$.

	algorithme des centres mobiles					
	EN12	E1	E2	AR12	AR1	AR2
T_R (Bruit)	78,7	81,3	91,7	74,8	74,0	89,5
T_R (Alose)	78,7	81,3	91,7	74,8	74,0	89,5
$P(dx_i / x_i)$	92,3	96,1	98,3	93,0	87,0	97,5
	65,0	66,9	50,0	56,7	61,2	41,8
$P(dx_i / \bar{x}_i)$	35,0	33,1	50,0	43,3	38,8	58,2
	7,7	3,9	1,7	7,0	13,0	2,5

	réseau de neurones					
	EN12	E1	E2	AR12	AR1	AR2
T_R (Bruit)	86,5	86,3	86,4	84,8	82,1	81,6
T_R (Alose)	86,0	86,3	86,3	84,7	82,0	81,8
$P(dx_i / x_i)$	90,0	89,9	86,0	89,5	88,6	80,7
	82,0	82,7	87,2	79,7	75,4	87,3
$P(dx_i / \bar{x}_i)$	17,0	17,3	11,0	20,0	24,3	12,6
	10,0	9,9	13,9	10,4	11,3	19,1

V - CONCLUSIONS

Les résultats obtenus pour le classement des signaux répartis comme un cinq sont équivalents . En effet les classes sont circulaires ce qui permet de bien les séparer à l'aide du calcul de la distance Euclidienne. Par contre, pour les signaux répartis en virgules, l'algorithme des centres mobiles commet systématiquement une erreur car les classes ne sont plus séparables simplement.

Sur les signaux réels, les résultats obtenus démontrent une supériorité incontestable du réseau de neurones quand on utilise directement les données brutes. Par contre, si on effectue une étape de prétraitement des paramètres données en [3] avant d'utiliser l'algorithme des centres mobiles, on obtient :

$$T_R(\text{Alose})=90\% \quad P(dx_i / x_i)=79\% \quad P(d\bar{x}_i / \bar{x}_i)=8\%$$

Les résultats deviennent supérieurs à ceux réalisés par le réseau. La solution utilisée fait intervenir le temps ce qui défavorise le réseau de neurones.

L'étape de prétraitement des paramètres est de loin la plus fastidieuse dans un problème de classification car elle impose un test systématique de toutes les solutions envisagées.

En conséquence, le temps d'apprentissage plus important que nécessite le réseau est largement compensé par la suppression de l'étape de prétraitement des paramètres. C'est pourquoi, nous avons adopté la solution du réseau de neurones pour résoudre les problèmes de classification au laboratoire.

[1] J.F. TROUILHET
Etude et réalisation d'un dispositif de reconnaissance automatique de signature acoustique
Thèse de doctorat de l'INP Toulouse Juin 1992

[2] J.M. NICOLAS, A. LEMER & D. LEGITIMUS
Identification automatique de bruits impulsifs en acoustique sous-marine par réseaux multi-couches.
Proc. Neuro Nimes 1989, p.269-278.

[3] J.F. TROUILHET, I. COUSTAUX, J.P. GUILHOT
Reconnaissance des formes dans le plan temps fréquence par modélisation paramétrique.
GRETSI Sept. 93