# Application des réseaux de neurones à la Veille Panoramique Infrarouge (1)

## PAR F. JACQUET (\*), H. NOËL (\*\*), N. DERYCKE (\*\*\*), J. DESMOUCEAUX (\*\*\*) ET G. BUREL (\*\*\*\*)

(\*) Division des Activités Optroniques de THOMSON-CSF (\*\*) SYSECA de THOMSON-CSF (\*\*\*) Division Systèmes Électroniques de THOMSON-CSF (\*\*\*\*) Laboratoires Électroniques de Rennes de THOMSON-CSF

RÉSUMÉ. - Nous utilisons les réseaux de neurones dans le cadre de la Veille Panoramique Infrarouge pour réaliser la détection et le pistage d'alarmes. La plupart des fonctions sont remplies par des perceptrons multicouches, qui en différents points de la chaîne de traitement, réalisent des tâches de prédiction, discrimination de données, optimisation, compression d'images. Nous utilisons également un algorithme dérivé du recuit simulé pour réaliser la fonction de pistage.

ABSTRACT. - This paper presents an application of neural nets to Infrared Panoramic Surveillance. Infrared images are processed by neural units realizing "blobs" detection and tracking. The paradigm mostly used in the processings is the Multilayer Perceptron with learning by Gradient Back Propagation. At different stages of the treatment, MLP are used for image prediction, pattern and image classification, image compression. We also use a model deriving from simulated annealing to solve the tracking problem.

#### 1. INTRODUCTION

Nous présentons dans cet article une application des techniques neuronales à une chaîne complète de traitement d'images. Le thème applicatif retenu est la Veille Panoramique Infrarouge, qui consiste à détecter et caractériser, pour tous types d'environnement et de conditions météorologiques, les engins survenant dans une zone de l'espace observée par une caméra infrarouge absorbant dans les bandes 3-5 ou 8-12 µm.

(1) Manuscrit reçu le 28 janvier 1991.

## APPLICATION DES RÉSEAUX DE NEURONES

#### F. JACQUET et al.

Après avoir décrit les missions d'un système de veille et le découpage en étages fonctionnels des traitements nécessaires à leur réalisation, nous présentons pour chaque module identifié la solution neuronale retenue : pré- et post-traitements éventuels des données, choix et mise en œuvre de l'algorithme, résultats et comparaison avec d'autres techniques existantes.

## 2. PRÉSENTATION DE LA CHAÎNE DE VEILLE PANORAMIQUE INFRAROUGE

## 2.1. Fonctionnalités recherchées

Le capteur infrarouge observe une zone de l'espace de 360° en azimut sur 5° en site. Par numérisation du signal délivré, on obtient une image digitale du panorama pour chaque tour.

Il s'agit d'extraire de ce flot de données l'information pertinente, c'està-dire :

• la *présence* d'objets susceptibles de représenter une menace (hélicoptères, avions, ...);

• leur évolution d'un tour à l'autre.

La chaîne de traitement de veille a donc une double mission :

• *détecter* les alarmes dans l'image avec une probabilité de détection élevée et une probabilité de fausse alarme aussi faible que possible,

• *pister* ces alarmes, c'est-à-dire relier les alarmes d'une image à celles des tours précédents sur des critères de cohérence en taille, luminance, ..., et pouvoir ainsi évaluer leurs vitesses et attitudes.

Les réseaux de neurones nous ont semblé intéressants pour résoudre ces tâches car ils offrent deux caractéristiques importantes : l'aptitude à travailler sur des données bruitées, floues, et la capacité d'élaborer des modèles par apprentissage, ce qui évite à l'utilisateur de le faire et doit faciliter la mise en œuvre. De plus, en vue d'une réalisation matérielle, l'implémentation parallèle des réseaux de neurones permet d'espérer des temps de traitement faibles, malgré l'importance des flux de données à traiter.

218

## 2.2. Découpage modulaire

Étant donné la complexité des fonctions à réaliser et l'ampleur des débits de données, il n'était pas envisageable de traiter le problème posé par un réseau de neurones unique. Le problème a donc été décomposé en plusieurs modules avec un découpage se calquant sur celui de la veille classique mis en place par ailleurs (<sup>2</sup>). La faisabilité par des techniques classiques et la consistance de chaque fonction étant ainsi acquises, il restait à évaluer la faisabilité des méthodes neuronales ainsi que leurs performances. Nous avons utilisé pour nos simulations des séquences d'images prises par des caméras infrarouges fixes et couvrant un secteur d'une dizaine de degrés en azimut.



Fig. 1. – Schéma fonctionnel de la chaîne de traitement et flux de données.

La figure 1 donne un schéma fonctionnel synoptique de la chaîne de traitement. On y retrouve les deux grandes fonctions présentées plus haut : détection et pistage.

(<sup>2</sup>) Simulateur de veille en cours de réalisation par les Divisions des Activités Optroniques, Systèmes Défense et Contrôle et Systèmes Électroniques. La détection présente différents aspects :

Détection globale : on recherche directement des groupes de pixels (ou « plots ») répondant à certains critères de contraste, de forme, ...

Détection élémentaire : on recherche des pixels susceptibles d'appartenir à une menace. Ils sont transmis à l'étage *Formation de Plots* qui examine s'il convient ou non de les regrouper.

Évaluation des plots : on note les plots obtenus par l'une ou l'autre des voies précédentes et on élimine ceux dont la vraisemblance est trop faible.

Caractérisation des fonds : ce module, pouvant coopérer avec les précédents, permet de connaître la nature du fond sur lequel se trouve l'alarme.

Le pistage fait intervenir deux modules qui interagissent :

*Initialisation de pistes :* pour le premier tour ou chaque fois qu'un plot de l'image courante ne peut être associé à ceux des tours précédents de manière satisfaisante, on crée une nouvelle piste.

Association plots-pistes : on associe les plots de l'image courante à ceux des images précédentes.

La suite de l'article présente les solutions retenues pour résoudre ces différentes tâches.

## 3. ÉTUDE DES FONDS

La détection des menaces, première fonction de la veille, peut être aidée par la connaissance de l'arrière-plan de l'image : la présence d'un fond complexe, peu homogène (fond végétal par exemple) rendra *a priori* la détection d'objets peu contrastés ou de petites tailles plus délicate, ce qui conduira à abaisser des seuils de détection éventuels. A un niveau de traitement plus élevé, la connaissance de la nature sémantique du fond (de type « ciel » ou « végétation ») pourra intervenir dans un processus de décision et permettre l'élimination de configurations impossibles (par exemple la reconnaissance d'un objet « char » alors que le fond est de type « ciel »).

Dans les images infrarouges et pour une application utilisant des réseaux de neurones adaptés au traitement de données bruitées, l'information la plus pertinente pour décrire les fonds est la texture, que l'on peut définir comme l'organisation bidimensionnelle des luminances.

somene des fummances.

C'est donc à partir de la texture que l'on va chercher à reconnaître dans les images les trois classes (<sup>3</sup>) : ciel, végétation et constructions. La classe végétation regroupe les zones de terre, de champs, les arbres, tandis que la classe constructions rassemble toutes les infrastructures humaines, telles que les maisons, les routes, les pylônes électriques,... La classe ciel couvre quant à elle, tous les aspects que peut prendre le ciel sous des conditions météorologiques variées : nuages, pluies, température, alternance jour/nuit.

Les classes de texture choisies ont une valeur sémantique; elles regroupent des natures physiques similaires dont l'apparence en infrarouge peut varier. Le problème se pose donc comme la discrimination de trois classes de texture à partir d'un échantillon.

## 3.1. Comment décrire la texture?

Nous analysons localement la texture par des fenêtres de  $32 \times 32$  pixels, glissant sur l'image complète avec un pas de quelques pixels (<sup>4</sup>).

Il n'est pas envisageable d'utiliser directement l'information des luminances d'une fenêtre, car :

• le réseau devrait avoir une couche d'entrée de 1 024 neurones, ce qui conduirait à des temps de calcul prohibitifs;

• isolé, un pixel de l'échantillon ne porte pas d'information sur la texture; c'est sa relation aux autres pixels, la statistique des luminances sur l'échantillon, qui sont porteuses d'information;

• le réseau devrait apprendre l'invariance en translation et en luminance moyenne, ce qui requerrait un nombre énorme d'exemples d'apprentissage.

On applique donc à l'échantillon un prétraitement pour en extraire un nombre limité d'attributs caractéristiques. Deux approches ont été

<sup>(&</sup>lt;sup>3</sup>) Ces classes sont celles qui apparaissaient dans la base d'images choisie. Cette répartition n'est évidemment pas exhaustive et pourrait être revue pour s'adapter à d'autres sites.

<sup>(&</sup>lt;sup>4</sup>) On prend par exemple un pas de 3 pixels dans les deux directions : l'identification de la nature du fond sera alors valable pour les 9 pixels du centre de la fenêtre. En fait, il faut trouver un compromis entre la précision de la détermination et la rapidité du traitement qui augmente avec le pas.

envisagées :

1. Une approche à 24 attributs :

• 9 attributs résumant l'information de répartition spatiale des pixels, obtenue par une décomposition multirésolution par transformée en onde lette ([7], [8]);

• 15 attributs décrivant la distribution des luminances des pixels : ce sont les fréquences des 15 cases autour de la valeur moyenne de l'histogramme des luminances sous-échantillonné sur 64 niveaux.

Cette approche a été décrite dans [13].

2. Une approche à 210 attributs :

• 40 attributs décrivant la distribution des luminances des pixels : ils sont obtenus par centrage sur la valeur moyenne et sous-échantillonnage avec un pas logarithmique de l'histogramme sur 256 niveaux des luminances;

• 170 attributs décrivant les liens entre pixels de l'échantillon : ce sont les valeurs de la fonction d'auto-covariance calculée avec une extension d'une demi-fenêtre (soit  $\pm 15$  pixels) dans les deux directions et souséchantillonnée d'un facteur linéaire 2 pour les zones éloignées du pixel central de la fenêtre.

## 3.2. Les réseaux de neurones utilisés

Pour les deux jeux d'attributs précédents, le modèle de réseau de neurones retenu est le perceptron multicouche avec apprentissage par rétropropagation du gradient. L'architecture retenue pour le jeu de 24 attributs a quatre couches de dimensions 24-16-16-3 avec connexions totales (soit 723 poids et seuils à calculer); pour le jeu de 210 attributs, on adopte une architecture à trois couches 210-6-3 (soit 1 287 poids et seuils). Dans les deux cas, la classe de l'échantillon est donnée par la position du neurone le plus actif sur la couche de sortie et sa saturation vis-à-vis des autres neurones permet d'élaborer une mesure de confiance de la manière suivante.

Si on note  $S_j$ , avec j=1, 2, 3, les saturations des neurones de la couche de sortie et si  $j_0$  est le neurone le plus saturé, alors :

$$\text{CONFIANCE} = \min_{j \neq j_0} \left( \frac{S_{j_0} - S_j}{2} \right)$$

REVUE TECHNIQUE THOMSON-CSF - VOL. 23 - Nº 1 - MARS 1991

Cette mesure de confiance est utilisée pour des post-traitements visant à atténuer les discontinuités locales des réponses du réseau lors du parcours d'une image entière. Ce post-traitement est de type vote majoritaire dans un voisinage du pixel considéré, avec pondération par la confiance et la distance. En d'autres termes, on donne d'autant plus de poids à un pixel du voisinage qu'il est plus proche du pixel central et que la confiance attribuée à la décision en ce point est plus élevée.

## 3.3. Résultats et commentaires

Avec ces deux approches, nous obtenons les résultats donnés dans le tableau I.

T		-	-	-		
	A.	ъ		E7.	Δ.	
	~ <b>A</b>	n		E	~ A	

	24 attributs	210 attributs
Nombre d'exemples en apprentissage	1 500	960
Performance	95,8 %	97,5 %
Nombre d'exemples en généralisation.	1 500	480
Performance.	88,8 %	86,5 %

Ces résultats s'avèrent meilleurs que les résultats obtenus avec d'autres classifieurs également testés (Plus Proches Voisins, Pseudo-Inverse, loi de Hebb, par ordre décroissant de performance). On constate donc le bon comportement des techniques neuronales sur des attributs différents en nature et en nombre, et une robustesse importante vis-à-vis des attributs choisis (hétérogénéité, redondance). Le choix du pré-traitement s'en trouve simplifié.

Lors du traitement d'images entières, quelques erreurs de discrimination sont apparues aux limites de zones et sur des sites géographiques différents de ceux utilisés pour l'apprentissage. Pour une application réelle, il conviendrait de collecter de nouvelles données *in situ* et de poursuivre l'apprentissage du réseau pour le rendre plus robuste au nouveau site d'implantation du système.

Coste incom de confiance en utilisée nour des positionnesses et attémper y confiance locales dos confiances de la confiance une image et la cost trabaiser est te cost de la confiance position de la cost trabaiser est te cost de la cost position de la cost trabaiser est te cost de la cost de la cost position de la cost trabaiser est te cost de la cost position de la cost d



Fig. 2. – Classification des fonds par l'approche à 210 attributs : à gauche : image infrarouge originale, à droite : zones de fond identifiées (noir=ciel, gris=végétation, blanc=constructions).

## 4. DÉTECTION, FORMATION ET TRI DES PLOTS

## 4.1. Détection globale

La détection globale a pour objectif la localisation dans l'image des groupes de pixels susceptibles de représenter une menace (hélicoptère, avion, ...). L'analyse du contenu de l'image est réalisée par un réseau de neurones auquel on présente en entrée les pixels d'une fenêtre glissant sur l'image avec un pas d'un quart de fenêtre. Pour chaque fenêtre présentée, le réseau élabore une réponse « la fenêtre contient une alarme » ou « n'en contient pas », qui permet d'établir une carte des zones d'alarmes.

## 4.1.1. LE RÉSEAU UTILISÉ

Dans un système de veille, on souhaite évidemment détecter les menaces le plus tôt possible : on va donc chercher essentiellement des plots de faibles dimensions (typiquement :  $10 \times 10$  pixels), représentant des engins

REVUE TECHNIQUE THOMSON-CSF - VOL. 23 - N° 1 - MARS 1991

éloignés. Compte tenu de la taille des objets recherchés, nous avons adopté une fenêtre d'analyse de dimensions  $15 \times 15$  pixels, soit 225 neurones en entrée du réseau.

Le réseau est un perceptron multicouche avec apprentissage par rétropropagation du gradient, retenu pour sa simplicité de mise en œuvre. Pour ne pas avoir un trop grand nombre de liaisons décorrélées (ce qui induit un risque de sur-apprentissage car on dispose de peu d'exemples à présenter au réseau), on a utilisé un masque de poids partagés de dimensions  $5 \times 5$ entre la couche d'entrée et la couche cachée, qui se trouve de ce fait dimensionnée à  $11 \times 11 = 121$  neurones. L'utilisation d'un masque correspond à l'approche classique en traitement d'images consistant à traiter localement l'image pour en extraire l'information (comme par exemple les masques de gradient pour déterminer les contours). Le masque est ici calculé par le réseau lors de la phase d'apprentissage, pendant laquelle on lui présente différents modèles d'engin dans des imagettes 15 × 15, d'autres engins simulés par un produit de gaussiennes superposé à un fond 15×15 extrait de l'image (ce qui permet de pallier le manque de modèles dans la base d'images), et autant d'imagettes de fond et de fausses alarmes (points chauds du fond). La couche de sortie, totalement connectée à la couche cachée, comporte deux neurones. La réponse du réseau est donnée par la position du neurone le plus actif.

Dans la phase de détection, pour limiter les fausses alarmes avec l'architecture choisie, on ne prend en compte que les décisions « alarme » pour lesquelles la valeur numérique du neurone le plus actif est supérieure à un certain seuil et on teste également l'écart entre les deux valeurs fournies par le réseau, afin d'éliminer les configurations où la décision alarme/fond n'est pas nette. Cet étage étant suivi d'un étage d'évaluation des plots (qui éliminera les fausses alarmes), on a intérêt à privilégier la détection en prenant des seuils assez bas. La connaissance de la nature du fond peut être utilisée ici pour adapter localement les seuils : plus le fond est texturé, plus la détection est délicate, donc plus les seuils doivent être bas. Par contre, pour les zones faiblement texturées, on peut prendre des seuils élevés pour limiter au maximum les fausses alarmes.

## 4.1.2. Résultats et commentaires

Le réseau est entraîné et testé sur une base de 534 imagettes  $15 \times 15$  isolées hors de l'image, base dont nous avons décrit précédemment les éléments et qui contient autant de vraies que de fausses alarmes. 400

224

226

#### F. JACQUET et al.

patterns sont utilisés pour l'apprentissage : on atteint un taux de reconnaissance correcte de 95,8 %. En généralisation sur le reste des patterns, ce réseau permet d'obtenir 86 % de bonne reconnaissance.

Dans la phase de détection proprement dite, où une fenêtre d'analyse  $15 \times 15$  glisse sur des images entières, la plupart des objets à détecter sont localisés, mais ceux qui sont peu contrastés ou de petite taille (rayon inférieur à 2 pixels) le sont plus difficilement.

## 4.2. Détection élémentaire et formation de plots

Une autre approche consiste à rechercher des pixels susceptibles d'appartenir à une alarme, puis à étudier leur regroupement pour former des plots.

On cherche ici à détecter ces pixels par soustraction à l'image réelle d'une image des fonds. Celle-ci est obtenue par un réseau de neurones qui assure la prédiction de la luminance du fond en chaque point de l'image à partir de la connaissance de son voisinage spatial (en pratique, une couronne centrée sur le point considéré).

L'écart entre la luminance prédite et la luminance réelle fournit une indication sur la probabilité de présence d'une menace en ce point. Un post-traitement consistant à seuiller la valeur absolue de l'écart de luminance permet d'isoler les pixels d'alarme.

Deux approches ont été envisagées pour la prédiction des luminances, que nous allons décrire dans les deux paragraphes qui suivent.

## 4.2.1. UTILISATION DU RÉSEAU EN INTERPOLATEUR

Classiquement, les fonctions d'interpolation permettant de prédire la luminance d'un pixel à partir de celles de ses voisins sont de type bilinéaire ou polynomial. Elles ont en général un effet de lissage, particulièrement sensible sur les contours : un contour est idéalement modélisé par une transition des luminances de type échelon (fig. 3a); mais le profil obtenu par interpolation est en général voisin de celui de la figure 3b.

L'idée sous-jacente de cette étude est d'utiliser les non-linéarités en tangente hyperbolique des neurones pour approximer plus finement le profil d'un contour idéal.

On a utilisé une structure de perceptron à 0 ou une couche cachée. Les neurones d'entrée reçoivent les luminances normalisées d'une couronne de



Fig. 3. – Profils de contour : (*a*) idéal; (*b*) interpolé.

2 pixels d'épaisseur située à une distance (<sup>5</sup>) de 5 pixels du centre soit  $(9+2\times2)^2-9^2=88$  valeurs. Lors de l'apprentissage par rétropropagation de l'erreur, la valeur désirée en sortie est la luminance réelle du pixel central. Le critère de mesure des performances est l'erreur quadratique moyenne, qui doit être aussi faible que possible. La couche cachée a été dimensionnée à 20 neurones.

Pour la prédiction d'images complètes, on obtient une erreur quadratique moyenne dont la racine carrée est d'environ 23 niveaux de gris (sur une dynamique de 256 niveaux). Les alarmes n'apparaissent pas dans l'image et sont donc bien détectées par soustraction. Cependant l'effet de lissage reste fort; les contours sont mal localisés (même si les valeurs de gris de part et d'autre sont justes), ce qui entraîne de fausses détections.

#### 4.2.2. PRÉDICTION PAR CLASSIFICATION

Étant donné la difficulté à obtenir les limites de fond de manière exacte, on va privilégier l'aspect « justesse des contours » en se ramenant à un problème de classification.

Des couronnes sont découpées comme précédemment dans l'image, puis seuillées de manière adaptative, en prenant un seuil égal à la demi-somme des luminances minimale et maximale sur chaque couronne; les pixels sont ainsi répartis en deux classes : la classe des pixels de valeur supérieure au

<sup>(&</sup>lt;sup>5</sup>) Cette taille de fenêtre est choisie pour que les alarmes jusqu'à 9 pixels de diamètre (que l'on cherche à détecter) disparaissent de l'image des fonds. En effet, comme elles seront entièrement encadrées par la couronne, leurs luminances ne seront pas prises en compte pour calculer le fond.

seuil et celle des pixels de valeur inférieure. Le rôle du réseau est alors de prédire, en fonction de la répartition des classes dans la couronne, à quelle classe de pixels doit être attribué le pixel central. La valeur de gris prise par ce pixel dans l'image prédite est alors le niveau de gris moyen à l'intérieur de la couronne de la classe à laquelle il a été attribué. L'image prédite complète est obtenue en faisant glisser la couronne sur l'image d'entrée. Le principe de cet algorithme est résumé dans la figure 4.



Fig. 4. - Principe de la prédiction par classification.

Le réseau retenu est un réseau à trois couches entièrement connecté. Il comporte 88 neurones en entrée, 20 neurones sur la couche cachée et deux sur la couche de sortie. Le point critique se situant au niveau de la prédiction des limites de zones, ce sont des patterns comportant une frontière qui sont majoritairement représentés dans la base. On a ajouté aux patterns obtenus par seuillage de couronnes réelles (souvent bruités, c'est-à-dire avec des pixels blancs et noirs inversés de part et d'autre de la frontière) des patterns générés artificiellement comportant une frontière du coudée. L'apprentissage se fait en deux temps : on présente d'abord au réseau uniquement ces patterns idéaux jusqu'à convergence (96,6 % de bonne reconnaissance en apprentissage sur 1 460 exemples), puis on ajoute à la base de données 1 460 patterns réels et on poursuit

l'apprentissage : le taux d'apprentissage est alors de 86,7 %. En généralisation, sur une base composée de 1000 patterns réels, le taux de reconnaissance est de 72 %. Le neurone le plus actif en sortie indique la réponse du réseau et les valeurs numériques des sorties permettent d'élaborer une mesure de confiance; lorsque celle-ci est trop faible, on effectue un posttraitement de type vote majoritaire parmi les plus proches voisins. La classe du pixel central étant déterminée, on lui alloue la valeur de gris moyenne de sa classe.

Avec cette méthode, la racine de l'écart quadratique moyen entre les images prédite et originale est plus faible (entre 10 et 15 niveaux de gris) que dans la méthode précédente : la prédiction est donc globalement meilleure. Cette approche permet de plus de prédire et de localiser correctement les contours rectilignes. Par contre, la prédiction des angles se fait mal : le réseau a tendance à arrondir les angles qu'il prédit, si bien que ceux-ci apparaissent dans l'image des détections. Cela s'explique par le fait que le réseau a une connaissance insuffisante de l'orientation des côtés de l'angle, la couronne ayant deux pixels d'épaisseur seulement.

## 4.2.3. FORMATION DE PLOTS

Les pixels détectés doivent être agglomérés pour reconstituer les plots. La difficulté réside dans la recherche d'un critère permettant de ne pas regrouper des détections provenant de sources chaudes (menaces ou non) distinctes mais proches, tout en regroupant les détections sur des engins ayant plusieurs sources de chaleur (par exemple rotor et stator d'un hélicoptère). Un expert a permis de mettre en évidence les points suivants :

• les plots ne peuvent pas être formés au pixel près; la plus petite entité distinguable est un plot formé avec une connexité à 8 voisins (appelé « plot élémentaire »);

• dans les données disponibles, la distance maximale entre deux groupes de pixels appartenant au même vrai plot est de 4 pixels;

• la distance entre deux groupes n'est pas un critère suffisant; il faut y adjoindre des critères de forme (positions relatives des pixels à l'intérieur d'un groupe) et de luminance des pixels (par rapport aux autres pixels du plot ou par rapport à ceux du fond).

Ces critères étant difficiles à quantifier, l'idéal serait de présenter au réseau des imagettes centrées sur les groupes à agglomérer et de le laisser élaborer les règles de décision par apprentissage. Cette approche a dû être écartée à cause de la non-représentativité de la base de données (seulement

REVUE TECHNIQUE THOMSON-CSF - VOL. 23 - N° 1 - MARS 1991

REVUE TECHNIQUE THOMSON-CSF - VOL. 23 - N° 1 - MARS 1991

33 cas de doublets de plots élémentaires voisins à ne pas regrouper, sur les 140 doublets disponibles). Pour réduire la dimensionnalité du problème, on a donc utilisé des attributs calculés sur les plots élémentaires ainsi que des informations caractérisant l'espace qui les sépare. Les informations *a priori* intéressantes ont été indiquées par l'expert. Pour caractériser un plot, on peut utiliser des attributs géométriques (surface, longueur, largeur, élongation, compacité, ...) ou des attributs de luminance (luminance maximale, minimale ou moyenne sur le plot élémentaire, écart maximal sur le plot par rapport au seuil ayant permis la détection). Pour caractériser l'espace entre deux plots élémentaires, on peut utiliser la luminance minimale sur le segment joignant leurs centres, l'orientation de ce segment, la distance entre les plots, ...

Le réseau utilisé est un perceptron à trois couches. Toujours à cause de la taille réduite de la base d'apprentissage (17 cas sur les 33 où il ne faut pas regrouper les détections, auxquels on ajoute 17 cas où il faut regrouper), on se limite en entrée à quatre attributs (un pour chaque plot élémentaire et deux pour l'espace entre eux). La couche cachée comporte trois neurones. En sortie, la solution la plus simple consiste à avoir un seul neurone qui prendra la valeur 1 dans le cas « ne pas regrouper » et la valeur 0 dans le cas où « il faut regrouper », les valeurs intermédiaires permettant de tester la confiance à accorder au résultat. Le plot est ainsi construit itérativement par agglomération de différentes détections, avec l'algorithme suivant :

1. Choisir une détection qui servira de point de départ à la formation du plot.

2. Pour toutes les détections dans un voisinage restreint (par exemple à une distance euclidienne inférieure à 10 pixels) :

(a) présenter en entrée du réseau les attributs du plot actuel, de la détection et de l'espace qui les sépare;

(b) s'il y a décision d'agglomération, ajouter la détection au plot actuel pour en former un nouveau.

Les attributs les plus discriminants se sont avérés être la luminance minimale de chaque plot, la distance et la luminance minimale entre les plots. Dès que l'on augmente le nombre de neurones ou le nombre d'attributs utilisés en entrée du réseau, un phénomène de sur-apprentissage apparaît et les performances en généralisation se dégradent.

Les attributs cités ont permis d'atteindre un taux de bonne reconnaissance en généralisation de 73,5 %, sur l'ensemble des exemples qui n'apparaissaient pas dans la base d'apprentissage. Les taux de reconnaissance à

4.3.1. Critères de comparaison des performances

Les performances d'un algorithme de discrimination entre deux classes 1 et 2 peuvent être représentées par une courbe « COR » (Caracteristic Operating Rate) qui donne le taux d'éléments de la classe 2 attribués à la

l'apprentissage ne sont guère plus élevés (environ 80 % pour différentes combinaisons d'attributs), ce qui confirme que la formation de plots reste un problème délicat et que toute l'information nécessaire n'est pas contenue dans les attributs utilisés. Les résultats pourraient sans doute être améliorés en travaillant directement sur des imagettes et non sur des attributs qui en sont extraits. Ceci sera possible lorsque nous disposerons d'une base de données plus étendue.

## 4.3. Évaluation des plots

Les plots obtenus par l'une ou l'autre des voies précédentes sont fournis au module d'évaluation qui doit permettre de séparer les objets intéressants (avions, hélicoptères, missiles, ...) des objets parasites (bruit thermique, points chauds du paysage, etc.).

Dans l'optique d'une chaîne de traitement opérationnelle, le débit de plots en sortie du module de détection est évalué à plusieurs milliers par seconde, ce qui implique des risques d'engorgement pour le module de pistage. Il est donc nécessaire de disposer d'une fonction de limitation de charge. L'évaluation doit permettre de fournir un critère au limiteur de charge sous forme d'une vraisemblance de chaque plot.

Le problème de l'évaluation des plots apparaît nettement comme un problème de discrimination. A ce titre, les techniques neuronales vont pouvoir être comparées aux techniques classiques de discrimination (Pseudo-Inverse, Plus Proches Voisins, ...). Deux types de données sont envisagés pour ces algorithmes :

• Tout ou partie des attributs de plots fournis par la fonction de formation des plots : attributs de forme (surface, élongation, compacité, ...), attributs de luminance (luminances moyenne, minimale, ...) et attributs de fond (luminance moyenne du fond, ...). Au total, 13 attributs possibles.

• Une imagette brute 11 × 11 centrée sur le plot considéré.

Dans les deux cas, le classifieur fournit une mesure de la vraisemblance du plot dont la valeur est comprise entre 0 (fausse alarme) et 1 (alarme).

## APPLICATION DES RÉSEAUX DE NEURONES

233

classe 1 (*i. e.* mal classés) en fonction du taux d'éléments de la classe 1 attribués à la classe 1 (*i. e.* bien classés). On en déduit aisément le taux d'éléments de la classe 2 attribués à la classe 2. L'objectif de ce module étant de filtrer les fausses alarmes en conservant la quasi-totalité des vrais plots, on se placera toujours pour comparer les performances, au point de la courbe COR où 95 % des vrais plots sont reconnus vrais.

Outre cette souplesse de réglage des performances, d'autres critères peuvent être pris en compte pour comparer deux algorithmes de discrimination dans la perspective d'une application opérationnelle :

- les taux moyens de reconnaissance;
- la rapidité d'exécution;
- la facilité d'intégration (calculs d'attributs, prétraitements).

#### 4.3.2. CONSTRUCTION DE COMBINAISONS D'ATTRIBUTS DISCRIMINANTES

Les histogrammes de répartition des attributs sur la base de données complète (70 000 plots au total, dont 243 vrais) permettent de déterminer *a priori* les attributs les plus discriminants.

Des combinaisons à 2, 3, 6, 10 et 13 de ces attributs ont été envisagées. On a retenu les combinaisons qui donnaient le meilleur taux de généralisation avec un perceptron multicouche à apprentissage par rétropropagation du gradient. A titre d'exemple, la combinaison luminance minimale sur le plot/écart de luminance par rapport au seuil de détection, s'est avérée la plus discriminante dans le cas de deux attributs.

#### 4.3.3. LES RÉSEAUX MIS EN ŒUVRE POUR LA DISCRIMINATION

Les réseaux utilisés comportent en entrée autant de neurones que d'attributs, soit 2, 3, 6, 10 ou 13. La couche de sortie comporte 1 neurone. La couche intermédiaire comporte entre 3 et 5 neurones, suivant la taille de la couche d'entrée.

Pour la discrimination à partir des imagettes  $11 \times 11$ , on a utilisé des réseaux à deux couches cachées avec connexions locales à poids partagés. Les masques sont de dimensions  $3 \times 3$  ou  $4 \times 4$ . Le meilleur taux de généralisation (faux plots reconnus faux : 55,2 %) a été obtenu avec un masque  $4 \times 4$  entre la couche d'entrée et la première couche cachée et un masque  $3 \times 3$  entre la première et la deuxième couches cachées. Les connexions entre la deuxième couche cachée et le neurone de sortie sont totales. On a donc un réseau d'architecture 121-64-36-1, soit 64 poids et seuils à calculer.

La base de données utilisée comporte 243 vrais plots et autant de faux. L'apprentissage des classifieurs est réalisé sur la moitié des données – imagettes ou attributs – disponibles dans la base, l'autre moitié servant à l'évaluation des performances en généralisation. Les deux classes sont représentées dans les deux bases de manière équiprobable. La répartition base d'apprentissage/base de généralisation a été faite de deux façons différentes :

• soit par tirage aléatoire des exemples;

• soit par tirage aléatoire des séquences, puis tirage aléatoire des exemples dans chaque séquence. Ceci permet d'éviter que des exemples issus d'images d'une même séquence – donc très semblables – ne se trouvent certains dans la base d'apprentissage, d'autres dans la base de généralisation, ce qui a pour effet de biaiser les résultats.

## 4.3.4. Résultats et commentaires

Que ce soit avec les attributs ou les imagettes, c'est presque toujours le perceptron multicouche qui permet d'obtenir les meilleures performances en généralisation. Lorsque ce n'est pas le cas, c'est au prix d'une complexité opératoire beaucoup plus élevée.

On présente dans les tableaux II et III les performances obtenues par le perceptron multicouche, ainsi que celles obtenues par différents algorithmes de discrimination opérant sur les mêmes données. Les valeurs indiquées sont les taux moyens de performance en généralisation, obtenus pour cinq tirages aléatoires des bases d'apprentissage et de généralisation. Le tableau II correspond au tirage aléatoire des exemples; le tableau III au tirage aléatoire des séquences. Il faut noter que les variances des mesures sont nettement plus élevées dans le second cas que dans le premier.

Les performances du perceptron varient donc entre 65 % pour le tirage des séquences et 85 % pour le tirage des exemples, à 95 % de vrais plots reconnus vrais. La chute des performances entre le tirage des exemples et le tirage des séquences s'explique par le fait qu'avec certains tirages, des cas délicats se trouvent dans la base de généralisation, alors que l'apprentissage n'a été réalisé qu'à partir de séquences « simples ». Ces deux valeurs sont les bornes du taux de réjection des faux plots que l'on peut espérer obtenir à terme, en augmentant le nombre et la diversité des cas dans la base d'apprentissage, qui actuellement n'est manifestement pas assez représentative du problème.

TABLEAU II

Méthode	2 attr. (%)	3 attr. (%)	13 attr. (%)	Imagettes (%)
Rétropropagation du gradient.	82,4	86,1	83,9	58,9
Nuées Dynamiques	81,9	80,7	81,7	72.8
Plus Proches Voisins.	82,5	83,5	76,8	81.8
Loi de Hebb	64,7	64,3	49,9	
Pseudo-Inverse.	64,5	57,6	58,7	37.7
Second ordre	76,3	76,4	82.0	51,1
Second ordre incrémental.	77,0	76,4	81.8	46.5
High Order Polynomial Input	79,2	80,6	81.0	72 7
Discrimination linéaire.	66,6	58,5	66.2	50.0

TABLEAU III

3 attr.

(%)

63.8

30,0

41.9

13 attr.

(%)

64.7

25.8

65.5

Imagettes

(%)

48,9

\_

-

2 attr.

(%)

65.7

33,6

37.2

51,4

Méthode

Rétropropagation du gradient. . . .

Plus Proches Voisins.

Loi de Hebb.

46.1 37.4 \_ Pseudo-Inverse. 49.2 52,4 48,4 37.7 47.5 49.0 59.4 Second ordre incrémental. 47.9 51.8 53.5 31,2 High Order Polynomial Input. .... 51.0 59,9 59.1 46.3 Discrimination linéaire. 47.2 53.8 52.2 35,2 De façon générale, les algorithmes traitant les imagettes sont de

complexité (en taille de réseau, temps de calcul, ...) nettement plus élevée que les algorithmes traitant les attributs, même si on inclut dans ces derniers les pré-traitements de calcul des attributs. Ces attributs sont simples à calculer (au moment de la formation de plots par exemple) et

REVUE TECHNIQUE THOMSON-CSF - VOL. 23 - N° 1 - MARS 1991

peuvent être utilisés pour d'autres modules de la chaîne (pistage). Les réseaux qui les utilisent sont de faibles dimensions, donc facilement intégrables et peu gourmands en temps de calcul (complexité linéaire en fonction du nombre de neurones pour les tailles considérées). Par ailleurs, l'algorithme neuronal envisagé permet de se positionner finement sur la courbe COR pour réaliser la limitation de charge des modules en aval. Il conviendrait de vérifier sur davantage d'exemples que les combinaisons de paramètres retenues restent bien les plus discriminantes. Enfin, il serait certainement possible d'améliorer le taux de réjection des faux plots en optimisant finement l'architecture des réseaux.

## 5. PISTAGE

#### 5.1. Initialisation de piste

Les plots détectés et triés sont transmis à l'étage de pistage, dont le rôle est d'associer, de tour à tour, les plots correspondant à la même alarme, de manière à pouvoir évaluer son mouvement.

Si N plots triés sont transmis à chaque tour, le nombre de pistes possibles après n tours est  $N^n$ . Au tour suivant, il faudra tenter d'associer les N nouveaux plots à ces N<sup>n</sup> pistes, ... ce qui s'avère rapidement inextricable. Il est donc important d'avoir un module d'initialisation des pistes qui ne transmet au module d'association plots/pistes que des séquences a priori cohérentes. En retour, le module d'association lui fournit les cas de plots n'ayant pu être associés à aucune piste existante de manière satisfaisante, afin qu'il examine l'opportunité d'en créer de nouvelles.

L'initialisation des pistes s'effectue classiquement en deux étapes :

1. Recherche des séquences de plots répartis sur plusieurs tours et dont les écarts sont cohérents avec le modèle d'évolution de l'engin recherché.

2. Évaluation de chacune de ces séquences en terme de vraisemblance d'une présence effective de menace. Si la vraisemblance d'une piste dépasse un certain seuil, elle est initialisée et transmise au module d'association plots/pistes.

Le réseau de neurones va traiter l'étape de calcul de vraisemblance des séquences qui lui seront présentées. On a choisi d'évaluer la cohérence de séquences de trois plots à partir d'imagettes 11×11 centrées sur chaque

REVUE TECHNIQUE THOMSON-CSF - VOL. 23 - N° 1 - MARS 1991

plot. Les imagettes permettent de présenter au réseau toute l'information disponible, sans avoir à rechercher les attributs les plus discriminants (<sup>6</sup>).

## 5.1.1. CONSTRUCTION DE SÉQUENCES DE PLOTS POUR LE TEST DU MODULE

Pour configurer le réseau et le tester sur une base de généralisation, il faut disposer de séquences qui traduisent l'évolution d'un plot au cours du temps. Pour chaque alarme présente dans 3 images successives, on construit 4 séquences de 3 imagettes : une séquence comprenant 3 imagettes de la vraie alarme et 3 séquences comportant 1, 2 ou 3 faux plots (dont la position dans la séquence est tirée aléatoirement); ces faux plots sont choisis dans le voisinage de la vraie alarme avec une probabilité gaussienne. Le critère de constitution d'une séquence est donc la proximité des positions des plots dans l'image. On crée ainsi 1320 séquences de trois imagettes, dont 820 serviront pour l'apprentissage et 500 pour le test du module d'initialisation de piste.

## 5.1.2. Compression des données

Chaque piste comprend une succession de 3 plots, chaque plot étant représenté par une imagette de 11 × 11 pixels. Une présentation directe de ces données en entrée du réseau nécessiterait 363 entrées. Compte tenu de la difficulté de la tâche (l'analyse de l'image doit être réalisée entièrement par le réseau, puisque l'information lui est présentée brute) et du nombre d'exemples à traiter, l'apprentissage risquerait d'être très long. C'est pourquoi on réalise une réduction des entrées par compression d'images, ce qui a l'avantage de n'induire qu'une très faible perte d'information.

La compression est réalisée par un perceptron à trois couches (à architecture « en diabolo » 121-11-121), dont les neurones de sortie sont linéaires. Lorsqu'une imagette de plot est présentée en entrée, on souhaite qu'elle soit restituée aussi fidèlement que possible en sortie (minimisation de l'erreur quadratique). Comme l'information doit passer par la couche intermédiaire de 11 neurones, on réalise une compression des données sur 11 valeurs (<sup>7</sup>). L'apprentissage du réseau de compression est réalisé sur une base de 11 648 imagettes de 11 × 11 pixels représentant des plots vrais ou faux. Les valeurs des luminances, initialement entre 0 et 255, sont ramenées dans l'intervalle [-1, 1] afin d'accélérer l'apprentissage (du fait de la symétrisation).

## 5.1.3. Les réseaux utilisés

Nous devons estimer la vraisemblance de chaque piste en fonction de 33 valeurs représentant sous forme comprimée des séquences de trois plots. Le réseau utilisé est un perceptron multicouche avec apprentissage par rétropropagation du gradient. En sortie, comme il s'agit d'un problème à deux classes (vraie piste vs fausse piste), on a choisi une représentation sur deux neurones. Les architectures suivantes ont été testées : 33-2, 33-10-2 et 33-15-10-2. A des fins de comparaison des performances, d'autres classifieurs ont été utilisés sur les mêmes données : Plus Proches Voisins, loi de Hebb, Pseudo-Inverse.

## 5.1.4. Résultats et commentaires

Les meilleurs taux de reconnaissance en généralisation ont été obtenus avec les réseaux à une et deux couches cachées (respectivement 91,4 et 91,2 % de bonne reconnaissance, sur les 500 pistes de la base de généralisation). Ces performances semblent meilleures que celles obtenues avec les Plus Proches Voisins (89 %) et sont nettement supérieures à celles de la Pseudo-Inverse (75 %). L'approche neuronale permet donc, pour une complexité opératoire réduite une fois l'apprentissage réalisé, de calculer une vraisemblance de piste avec une bonne fiabilité, sans avoir à modéliser l'évolution des attributs de la menace.

## 5.2. Association plots/pistes

Le module d'association plots/pistes permet de résoudre le lien entre les observations du tour courant (les plots détectés) et les observations des tours passés, résumées dans l'information de piste. Les pistes générées et validées, dans un premier temps, par le module d'initialisation, sont entretenues par l'association, qui indique les couples (plot, piste) qui ont la vraisemblance la plus élevée. Ces couples sont utilisés pour mettre à jour les informations de piste grâce à la nouvelle mesure.

Dans le cas le plus simple, où une seule observation se trouve dans la fenêtre de validation d'une piste, le problème d'association est résolu de fait. Par contre, pour des pistes rapprochées ou en croisement, avec un nombre important d'alarmes vraies et fausses, des situations de conflit

237

<sup>(&</sup>lt;sup>6</sup>) L'approche utilisant des attributs est en cours d'évaluation à la Division Systèmes Défense et Contrôle; elle présente l'avantage d'une complexité opératoire réduite.

<sup>(&</sup>lt;sup>7</sup>) Il ne s'agit pas là d'une extraction d'attribut, car l'information extraite permet de reconstruire fidèlement l'imagette, ce que ne permettent pas en général les attributs.

apparaissent et l'association n'est plus triviale : on peut avoir plusieurs plots dans la fenêtre de validation d'une piste ou un plot susceptible d'être associé à plusieurs pistes.

Moyennant quelques hypothèses simplificatrices :

- deux plots ne peuvent pas provenir de la même alarme;
- un plot ne peut pas provenir de deux alarmes distinctes;
- une alarme peut ne pas être détectée;
- un plot peut être une fausse alarme,

nous adoptons une modélisation probabiliste du problème, qui conduit au calcul d'une matrice de vraisemblance. Les colonnes de cette matrice correspondent aux pistes (résumées par leur prédiction (<sup>8</sup>) à l'instant courant). Les lignes correspondent aux plots courants, décrits par leur vecteur d'attributs, plus une ligne supplémentaire représentant un plot fictif de non détection de l'alarme. Chaque élément *i*, *j* de la matrice est une mesure de la vraisemblance de l'association d'un plot *i* à une piste *j* : les plots dont le vecteur d'attributs se trouve dans le volume de validation d'une piste ont une vraisemblance d'association calculée en fonction de la probabilité de détection d'une alarme, du taux de fausse alarme et du volume de validation; les autres ont une probabilité nulle (association. Un prétraitement supprime les cas triviaux : ligne entièrement nulle (correspondant à un plot non validé), colonne entièrement nulle à l'exception du premier élément (piste non vue).

## 5.2.1. LE RÉSEAU UTILISÉ

La matrice des vraisemblances élémentaires d'association est présentée au réseau qui va déterminer, par apprentissage sur des hypothèses d'association globale, la solution de vraisemblance maximale.

L'algorithme utilisé [10] est un algorithme d'apprentissage par essaierreur, basé sur des techniques de recuit simulé et de mutation génétique. Chaque piste est représentée par un neurone de sortie; les neurones de sortie sont reliés à l'ensemble des neurones d'entrée représentant les plots validés pour la piste considérée. Les poids des liaisons synaptiques sont initialisés à la valeur de la vraisemblance d'association du plot et de la piste considérés. L'apprentissage se fait selon deux phases réitérées :

1. *Essai* : une hypothèse d'association est tirée au sort dans un ordre aléatoire; chaque neurone choisit un plot parmi ceux qui restent libres, avec une probabilité proportionnelle au poids synaptique. L'état du neurone devient la vraisemblance de cette association. La vraisemblance de l'association globale est le produit des neurones.

2. *Adaptation :* une loi d'adaptation décrit la mise à jour des poids synaptiques : s'il y a amélioration (respectivement diminution) de la vraisemblance globale par rapport à l'essai précédent, on renforce (resp. réduit) les poids synaptiques des neurones de la combinaison.

## 5.2.2. Résultats et commentaires

Les performances ont été évaluées sur des données synthétiques, les données réelles n'offrant pas assez de scénarios complexes.

Au cours de l'itération des phases essai-adaptation, certaines combinaisons sont abandonnées, d'autres émergent rapidement. L'apprentissage donne en lui-même la solution du problème. Le seul paramètre à fixer est un facteur de vitesse intervenant dans la loi de mise à jour des poids : il faut trouver un compromis entre la rapidité de convergence et la qualité du résultat. Le critère d'arrêt peut être soit la durée (on arrête après Nitérations et on retient le meilleur résultat trouvé), soit l'atteinte de la solution optimale (on constate que l'algorithme neuronal converge vers la solution du maximum de vraisemblance, en fonction du nombre d'itérations).

Le principal intérêt de cet algorithme est sa complexité linéaire en fonction du nombre de pistes, contrairement aux algorithmes classiques (type Joint Probabilistic Data Association JPDA, [11] et [12]) qui ont une complexité exponentielle, mais permettent d'atteindre l'optimum du maximum de vraisemblance en une seule passe. D'après les simulations, l'approche neuronale est avantageuse en coût de calcul lorsqu'on a un nombre élevé de pistes.

#### 6. CONCLUSION

Les résultats présentés dans cet article montrent la capacité des réseaux de neurones à traiter les différents modules de la Veille Panoramique

<sup>(&</sup>lt;sup>8</sup>) Chaque prédiction est un vecteur d'attributs extrapolés à partir des observations passées : attributs de position angulaire, attributs géométriques, attributs de luminance.

Infrarouge. La majorité des fonctions sont remplies par les réseaux avec des performances très satisfaisantes et une complexité compétitive vis-àvis d'autres méthodes non neuronales.

Les techniques neuronales s'appliquent à des problèmes très divers (discrimination, optimisation, prédiction, compression), permettent un développement rapide et nécessitent peu de modélisation. Toute la connaissance *a priori* que l'on a du problème est utilisée pour définir des pré- et post-traitements adéquats, les réseaux construisant un modèle par apprentissage.

Les réseaux de neurones se sont montrés aptes à traiter différents types de données, des images brutes aux attributs de haut niveau, ce qui permet une grande souplesse de mise en œuvre. L'extraction d'attributs, inutile pour certains modules comme la détection globale, s'est parfois avérée indispensable pour réduire la taille du réseau et faciliter l'apprentissage en présentant une information plus pertinente (§ 3).

Au point de vue du choix de l'algorithme, le perceptron multicouche se retrouve à quasiment toutes les étapes du traitement, car il s'est avéré plus performant que les autres techniques. Une éventuelle réalisation matérielle du système global s'en trouvera facilitée. Outre les poids obtenus par apprentissage, le nombre de paramètres à fixer est en général assez faible et l'architecture peut faire l'objet d'optimisations pour gagner quelques points de performance. C'est ce que l'on envisage pour la suite de l'étude, ainsi que la validation des simulations sur des flots de données continus issus de la Veille Panoramique.

## BIBLIOGRAPHIE

- D. E. RUMELHART and J. L. MCCLELLAND, Parallel Distributed Processing. Exploration in the Microstructure of Cognition, vol. 1 : Foundations., Bradford Book, MIT Press, 1986.
- [2] Y. LE CUN, Modèles connexionnistes de l'apprentissage, Thèse Univ. Paris-VI, 1987.
- [3] R. O. DUDA and P. E. HART, Pattern Classification and Scene Analysis, Wiley, New York, 1973.
- [4] F. VALLET, Approche globale du problème de discrimination : Aspects probabilistes, Rapport interne du Laboratoire Central de Recherches de THOMSON-CSF, ARSF-90-11, oct. 1990.
- [5] E. PERNOT, F. VALLET, D. POTIER and J. G. CAILTON, NeuroClass: A Software Environment for Classification of Signals by Neural and Conventional Methods, Laboratoire Central de Recherches de THOMSON-CSF, 1990.

[6] A. GAGALOWICZ, Vers un modèle de textures, Thèse de Doctorat, Univ. Pierre-et-Marie-Curie, Paris-VI, mai 1983.

APPLICATION DES RÉSEAUX DE NEURONES

- [7] S. MALLAT, A Theory for Multiresolution Signal Decomposition: the Wavelet Representation, IEEE PAMI, vol. 2, n° 7, July 1989.
- [8] J. C. FEAUVEAU, Analyse multirésolution par ondelette non orthogonale et bancs de fîltres numériques, Thèse de Doctorat, Univ. Paris Sud, janv. 1990.
- [9] R. A. HUTCHINSON, Development of an MLP feature location technique using preprocessed images, INNC Proceedings, vol. 1, p. 67-70, Paris, juillet 1990.
- [10] K. CHEN, A Simple Learning Algorithm for the Travelling Salesman Problem, 1989.
- [11] K. C. CHANG, C. Y. CHONG and Y. BAR-SHALOM, Joint Probabilistic Data Association in Distributed Sensor Networks, IEEE Trans. Autom. Control, vol. AC-31, p. 889-897, Oct. 1986.
- [12] T. E. FORTMANN, Y. BAR-SHALOM and M. SCHEFFE, Sonar Tracking of Multiple Targets Using Joint Probabilistic Data Association, IEEE J. Oceanic Engineering, OE-8, vol. 5, . 173-184, July 1983.
- [13] J. DESMOUCEAUX et N. DERYCKE, Discrimination de textures infrarouges par réseaux de neurones, Rev. Tech. THOMSON-CSF, vol. 22, n° 4, p. 637-648, déc. 1990.



Florence JACQUET École Supérieure d'Électricité, 1989. Ingénieur à la Division des Activités Optroniques de THOMSON-CSF, Rue Guynemer, BP n° 55, 78283 Guyancourt Cedex. Tél. : 30.96.71.94.



Hervé NOEL École Centrale de Paris, 1989. Ingénieur à SYSECA de THOMSON-CSF, 315, bureaux de la Colline, 92213 Saint-Cloud Cedex. Tél. : 49.11.70.00.



Nicolas DERYCKE École Centrale de Paris, 1988. Ingénieur au Département d'Études Techniques à la Direction Technique de la Division Systèmes Électroniques de THOMSON-CSF, 9, rue des Mathurins, BP n° 150, 92223 Bagneux Cedex. Tél. : 40-84-40-00.

## Jean DESMOUCEAUX

École Nationale Supérieure des Télécommunications, 1987. Ingénieur au Département d'Études Techniques à la Direction Technique de la Division Systèmes Électroniques de THOMSON-CSF, 9, rue des Mathurins, BP n° 150, 92223 Bagneux Cedex. Tél. : 40-84-40-40



Gilles BUREL DEA d'Électronique (Univ. Brest), 1986; École Supérieure d'Électricité, 1988. Ingénieur au laboratoire Intelligence Artificielle en Imagerie des Laboratoires Électroniques de Rennes de THOMSON-SCF, Avenue de Belle-Fontaine, 35510 Cesson-Sévigné. Tél. : 99.25.42.88.

# **Optimisation par réseaux de neurones : application au traitement d'antennes (**<sup>1</sup>**)**

## PAR P. MARTIN ET B. LOBERT

Division Radiocommunications, Guerre Électronique et Sécurité de THOMSON-CSF

RÉSUMÉ. – La méthode du Maximum de Vraisemblance est la méthode optimale d'estimation de la direction d'arrivée de sources bande étroite par un réseau de capteurs. Pourtant, du fait de sa complexité, elle reste peu appliquée. D'autre part, compte tenu des progrès effectués dans la technologie VLSI, l'intérêt dans l'emploi des réseaux de neurones s'est récemment développé. Notamment, Hopfield a montré que les réseaux entièrement interconnectés peuvent résoudre des problèmes d'optimisation complexes, en identifiant la fonction à optimiser avec l'énergie de Lyapunov du réseau. Le parallélisme obtenu peut être intéressant pour une application temps-réel.

Cet article a un double objectif :

- effectuer une synthèse des méthodes neuronales d'optimisation;
- appliquer un réseau de neurones adaptatif à la méthode du Maximum de Vraisemblance.

ABSTRACT. – The Maximum Likelihood Method is an optimal method for the estimation of the directions-of-arrival of narrow-band sources by a passive array of sensors. However, it hasn't been applied, because of its high computational load. With the current advances in VLSI technology, the interest in the use of neural networks has recently grown. Especially, Hopfield has shown that totally interconnected neural networks can resolve complex optimization problems, by identifying the function to minimize to the Lyapunov energy of the network. The obtained massive parallelism can be interesting for a real-time implementation.

The main goals of this paper are:

- to try to resume the neural methods of optimization;
- to show an adaptive neural method for Maximum Likelihood bearing estimation.

(1) Manuscrit reçu le 13 août 1990.

Revue technique thomson-CSF - vol. 23 - n° 1 - mars 1991 0012-9593/1991/243/\$5.00/ $\odot$  Gauthier-Villars