

Utilisation des STANN en audio : illustration en reconnaissance de chiffre

David Mercier, Renaud Séguier

Supélec : Équipe Electronique Traitement du Signal et Neuromimétisme
Avenue de la Boulaie - BP28 35511 Cesson Sévigné, France.
e-mail : David.Mercier@supelec.fr, Renaud.Seguier@supelec.fr

Résumé

Notre équipe travaille depuis une dizaine d'années sur un modèle de réseaux de neurones utilisant non pas des entrées continues mais des entrées impulsionnelles, s'inspirant en cela de la nature des entrées des neurones biologiques. Ce modèle, appelé STANN (Spatio-Temporal Artificial Neural Network), permet ainsi de traiter, comme son nom l'indique, des données spatio-temporelles, c'est à dire des données où l'information spatiale évolue au cours du temps. La mise en pratique de cette famille de réseaux de neurones sur les problèmes de l'écriture manuscrite et de la lecture labiale en a montré le potentiel. D'autre part, nous avons également proposé une méthode générique de génération d'impulsion pour leur utilisation. Nous confrontons ici ces outils à l'audio en prenant comme objectif la reconnaissance de chiffres.

1. Introduction

La reconnaissance audio est un problème depuis longtemps étudié, et bien évidemment, de nombreux modèles de réseaux de neurones ont été testés pour cette opération (parmi beaucoup d'autres outils). Mais jusqu'à présent, aucun des récents modèles de neurones à impulsions, s'inspirant en cela de la nature des entrées des neurones biologiques, n'ont été utilisés. La raison la plus probable est que le signal audio est loin d'être un signal impulsionnel et que la conversion est peu évidente au premier abord.

Dans cet article, nous allons montrer que le prétraitement simple et générique que nous avons proposé dans [13] dans le cadre de la lecture labiale pour générer des impulsions peut parfaitement être étendu pour le signal audio et que son utilisation avec le STAN (l'un de ces récents modèles de neurones à impulsions dont l'efficacité a déjà été montrée sur les problèmes de l'écriture manuscrite [16] et de la lecture labiale [2]) donne des résultats très satisfaisants.

Après un rappel du fonctionnement de notre système de classification et des contraintes sur les entrées qui en découlent, nous présenterons les données disponibles dans la base que nous avons utilisée puis nous ferons un rapide tour des différents traitements du son qui existent. Ceci nous permettra de choisir la nature des entrées de notre système et de donner les premiers résultats de nos simulations.

2. Le système de classification

2.1. Le STAN

Le STAN (Spatio-Temporal Artificial Neuron, neurone artificiel spatio-temporel) est un modèle de neurone créé par Vaucher [18], le codage sous-jacent ayant été intégré dans des architectures neuronales classiques [17]. Son principe est de coder des événements discrets ayant deux degrés de liberté (amplitude et date) sous la forme de nombre complexes ayant aussi deux degrés de liberté (amplitude et phase).

Un neurone STAN est caractérisé par quatre éléments (voir Figure 1). Tout d'abord, comme pour le modèle neuronal classique, un STAN se singularise des autres par son *vecteur poids* (W), sa *fonction potentielle* (V ou D) et sa *fonction de transfert* (ou *fonction d'activation* F). Le dernier paramètre, nommé TW , représente la taille de la fenêtre temporelle à l'intérieur de laquelle on désire identifier des séquences d'impulsions (on peut l'assimiler au retard maximal dans un neurone classique dynamique).

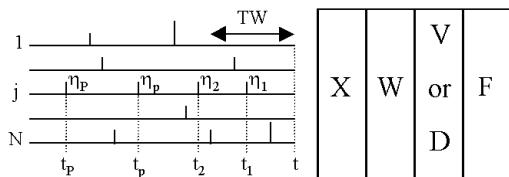


Figure 1: Le STAN (Spatio-Temporal Artificial Neuron)

Les calculs se mènent de la façon suivante : une impulsion d'amplitude η_1 émise à l'instant t_1 sur la $j^{\text{ème}}$ composante du vecteur d'entrée X est codée à l'instant courant t par le nombre complexe :

$$x_j(t) = \eta_1 e^{-\mu_s \tau_1} e^{-i \text{atan}(\mu_T \tau_1)}$$

$$i = \sqrt{-1} \quad \tau_1 = t - t_1 \quad \mu_s = \mu_T = \frac{1}{TW}$$

Si une seconde impulsion d'amplitude η_2 est émise à l'instant t_2 sur la même entrée, elle est ajoutée à l'état courant et c'est ce résultat que l'on fait vieillir¹. Ainsi :

$$x_j(t_2) = \eta_1 e^{-\mu_s(t_2 - t_1)} e^{-i \text{atan}(\mu_T(t_2 - t_1))} + \eta_2$$

$$= \rho e^{i\phi}$$

et plus tard :

$$x_j(t) = \rho e^{-\mu_s(t - t_2)} e^{-i \text{atan}(\tan \phi + \mu_T(t - t_2))}$$

Si de nouvelles impulsions arrivent, on recommence la même opération.

Une fois le vecteur d'entrée X calculé, le potentiel est égal soit au produit scalaire her-

mitien :

$$V(X, W) = \sum_{j=1}^N \overline{w_j} \cdot x_j$$

soit à la distance hermitienne :

$$D(X, W) = \sum_{j=1}^N (x_j - w_j) \cdot \overline{(x_j - w_j)}$$

Ces fonctions potentielles dans les complexes sont à associer respectivement au produit scalaire et à la distance euclidienne pour les modèles classiques dans les réels.

La fonction d'activation F appliquée au potentiel détermine alors la sortie y . Lorsque la distance hermitienne est utilisée comme fonction potentielle, le résultat est réel et les fonctions de transfert classiques sont utilisées. Quand c'est le produit scalaire hermitien, le résultat étant en général complexe, des fonctions particulières, respectant certaines propriétés adaptées au corps des complexes [9], doivent être utilisées. La fonction retenue a été proposée dans [12].

$$F(x + iy) = p \cdot x + ip \cdot y$$

$$p = \frac{\tanh(x^2 + y^2)}{\sqrt{x^2 + y^2}}$$

Cette fonction, qui applique une tangente hyperbolique au module sans changer la phase du complexe, permet un bon compromis entre les différents critères nécessaires pour assurer un bon apprentissage une fois ces neurones intégrés dans des architectures neuronales comme le perceptron multicouche (voir ci-dessous).

2.2. Intégration dans des réseaux: les STANN

Disposant toujours d'une algèbre avec la notion de produit scalaire et de distance, il a été possible d'intégrer ce modèle de neurones dans des architectures courantes de réseaux de neurones, en adaptant assez facilement les algorithmes d'apprentissage et d'exploitation à l'algèbre complexe. Ainsi, [17] et [1] présentent des versions spatio-temporelles pour le perceptron multicouche (ST-MLP), pour les réseaux à bases radiales

1. ce qui est différent de "faire vieillir les impulsions indépendamment puis les sommer" à cause de la fonction *arctan*, cf [1]

(ST-RBF), pour les réseaux de Reilly, Cooper et Elbaum (ST-RCE), pour les cartes auto-organisées de Kohonen (ST-Kohonen) et pour leurs versions sans voisinage (ST-Kmeans).

Une procédure générale d'utilisation de ces réseaux utilisant la distance hermitienne pour la classification de signaux spatio-temporels a ensuite été proposée [1].

Pour notre exemple, comme dans [13], nous utiliserons un classifieur ST_RCE. Un RCE est un réseau en deux couches. Sur la première, les poids correspondent à des points de l'espace d'entrées et on calcule les distances de l'entrée présentée à ces points (fonction potentielle D comme dans un RBF). La fonction d'activation est normalement une fonction rectangle qui permet de mettre la sortie à 1 lorsque l'entrée est dans la sphère d'influence du neurone et 0 autrement. Pour le forcer à donner une réponse, nous utilisons plutôt un «tout au vainqueur» (*Winner Takes All*), c'est à dire que le neurone qui a la plus grande activation voit sa sortie à 1 et les autres à 0. La seconde couche possède autant de neurones qu'il y a de classes. Les liens entre la première couche et la seconde sont des liens binaires faits de telle sorte qu'un neurone actif de la première couche active un et un seul neurone de la seconde couche. Durant l'apprentissage, qui se fait en plusieurs passes, le nombre de neurones de la première couche est variable. A l'initialisation, la couche est vide. Pour chaque nouvel exemple, on calcule les distances aux centroïdes de chaque neurone et on le compare à sa sphère d'action. Trois cas se présentent :

- L'exemple n'est dans aucune sphère d'action. On crée alors un nouveau neurone sur la couche ayant pour centroïde l'exemple en question et pour rayon d'action k fois la plus petite distance aux autres neurones (k inférieur mais proche de 1).
- L'exemple est dans une sphère d'action d'un neurone associé à une autre classe : le rayon d'action de ce neurone est diminué à k fois la distance entre le neurone et l'exemple, et un nouveau neurone est

créé comme décrit ci-dessus.

- L'exemple est dans une sphère d'action d'un neurone associé à la même classe. Aucune modification n'est apportée au réseau.

On réitère l'apprentissage sur la base jusqu'à ce que la présentation de tout les exemples de la base d'apprentissage ne créent aucun nouveau neurone.

2.3. Prétraitement générique

Pour générer simplement des impulsions à partir de signaux multidimensionnels évoluant dans le temps de façon continue, nous avons proposé dans [13] de réaliser une Quantification Vectorielle (QV) statique sur la base des signaux pris à chaque instant. Cette quantification vectorielle permet d'associer à la forme statique (définie à un instant donné par l'ensemble des capteurs) un prototype de forme. La procédure générique définie quatre étapes mais dans le cas particulier de l'application à l'audio, nous l'avons restreinte à trois étapes :

Apprentissage :

- Définition des M prototypes statiques.

Exploitation :

- Identification à chaque instant du prototype P_k qui se rapproche le plus du signal statique $X(t)$ en entrée.
- Emission d'une impulsion. Les M sorties du module de prétraitement sont nulles sauf celle qui correspond au prototype k sur laquelle est générée une impulsion. La valeur de cette impulsion sera ici de 1. Nous n'aurons donc pour l'application du STAN à l'audio, une fois la nature des entrées choisie, qu'à définir un paramètre : le nombre de prototypes à conserver dans la phase de quantification vectorielle.

3. Le problème étudié : reconnaissance audio de chiffres.

Dans [13], afin d'utiliser comme références le protocole et les résultats de [1], nous nous étions intéressés à la lecture labiale dans un cadre monolocuteur avec une base

fabriqué dans notre laboratoire. Pour l'application du STAN à l'audio, nous avons décidé de travailler sur une base déjà définie et connue. Notre choix s'est porté sur une base normalement bimodal, c'est à dire permettant l'exploitation à la fois de la modalité visuelle (les images de la vidéo sont disponibles) et de la modalité audio (le son enregistré) pour faire la reconnaissance automatique de chiffres. De cette base, nous ne nous sommes servis que du son. Il s'agit de la base Tulips1 [15].

Cette base présente douze personnes prononçant deux fois les chiffres de 1 à 4 en anglais. Selon le chiffre prononcé, le locuteur et la séquence, nous disposons de 6 à 16 images pour un chiffre. L'information audio est présente sous deux formes : une forme brute et une forme prétraitée via 26 paramètres son par images : 12 coefficients cepstraux (voir 4.8.), la fonction log-énergie et leurs dérivées.

Maintenant que nous disposons de l'outil et de la base, il nous reste à définir sous quelle forme nous allons prendre le signal son, sachant que la quantification vectorielle doit être robuste sur ce signal (c'est à dire que deux sons quasi identiques doivent être quantifiés identiquement la majorité du temps). C'est le but du paragraphe suivant.

4. Le signal audio et ses traitements : que choisir ?

Comme nous l'avons montré dans [13], la représentation du son doit être robuste pour la quantification vectorielle. Cela exclu d'emblée la forme brute du signal sonore qui est beaucoup trop oscillante et sensible à la phase lorsque l'on cherche à la quantifier. Nous allons donc rappeler l'origine biologique du son et les différents traitements existants puis voir lesquels sont susceptibles de bien fonctionner pour la quantification vectorielle.

4.1. Formation

Le principe de production de la parole est montré Figure 2. Le diaphragme expulse

l'air des poumons, produisant ainsi l'énergie nécessaire à la parole. L'air arrive via la trachée-artère dans le larynx où se trouvent les cordes vocales. Si le son à produire est non-voisé ou chuchoté, les lèvres symétriques constituant les cordes vocales forment une grande ouverture triangulaire nommée glotte et l'air continue son chemin. Si le son est voisé, les cordes vibrent périodiquement (ouverture brusque, fermeture plus progressive) ce qui module le son en impulsions périodiques de pression dont la fréquence est nommée fondamentale ou *pitch* (de 80 à 600 Hz). L'air arrive alors dans le conduit vocal proprement dit, avec d'abord la cavité pharyngienne, puis en parallèle la cavité buccale et la cavité nasale. Cette dernière peut être utilisée ou non grâce au voile du palais qui permet une isolation totale de la cavité nasale.

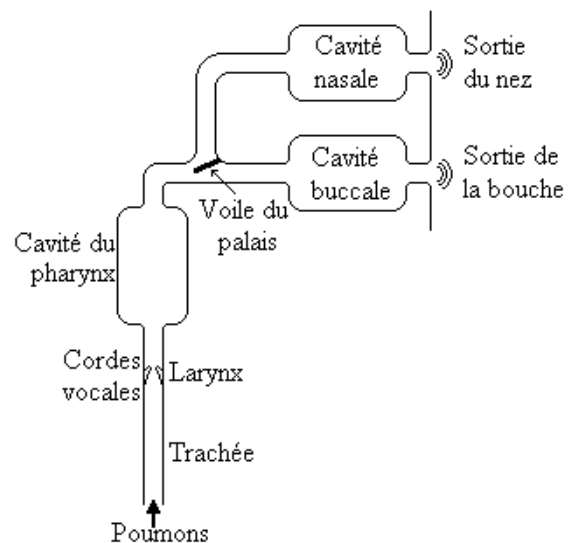


Figure 2: Modèle de production de la parole

4.2. Modèle autorégressif (AR)

Le modèle AR est une modélisation mathématique basée sur la mise en équation simplifiée du modèle physique et aboutissant à une transmittance dite *tous-pôles* du système. Le signal sonore à la sortie du larynx est assimilé soit à un bruit blanc (sons non voisés), soit à la réponse d'un passe-bas d'ordre 2 à un peigne de Dirac. Le conduit vocal lui est modélisé par une succession de tubes acoustiques, c'est à dire une cascade de réso-

nateurs d'ordre 2. Une justification théorique peut être trouvée dans [6].

Au final, le modèle AR consiste à dire que le son X est le résultat du filtrage par un filtre *tous-pôles* H d'une source U qui est soit un bruit blanc centré gaussien, soit un train d'impulsion ayant pour fréquence le *pitch*.

En terme de transmittance, on obtient :

$$X(z) = U(z) \cdot \frac{\sigma}{A(z)}$$

$$H(z) = \frac{\sigma}{A(z)}$$

Si :

$$A(z) = \sum_{i=0}^p a_i \cdot z^{-i} \quad a_0 = 1$$

alors dans le domaine temporel, cela revient à la récurrence suivante :

$$x[k] + \sum_{i=1}^p a_i \cdot x[k-i] = \sigma \cdot u[k]$$

C'est à dire qu'un échantillon est une combinaison linéaire des échantillons précédents et du terme d'excitation. C'est cette récurrence qui définit le modèle autorégressif d'ordre p .

4.3. Analyse par prédiction linéaire

Historiquement, l'analyse par prédiction linéaire est la technique d'analyse de la parole la plus importante. Le principe est d'utiliser la relation de récurrence du modèle AR en considérant que la source U est toujours nulle et que le modèle autorégressif est autoentretenu. Les coefficients a_i sont calculés afin de minimiser l'erreur quadratique moyenne du signal estimé.

4.4. Analyse spectrale

Bien évidemment, tout à fait au début des travaux sur la parole, la transformée de Fourier était très utilisée car constituait le seul outil disponible. Mais cette transformation n'est pas satisfaisante. Si elle permet de connaître les fréquences présentes dans le signal, elle ne permet pas de savoir quand. Or le signal de parole ne pouvant être considéré comme stationnaire que durant 10 à 30 ms,

cette information est vitale pour l'analyse.

La première solution à ce problème a été la Transformée de Fourier à fenêtre glissante. Le concept est de moduler la sinusoïde de la transformée de Fourier en la multipliant par une fonction-fenêtre. En général, cette fonction-fenêtre vaut zéro en dehors d'un intervalle prédéfini (par exemple fenêtre rectangulaire ou fenêtre de Hamming). Toutefois la seule restriction est qu'elle possède une énergie finie donc une gaussienne est parfaitement possible (dans ce dernier cas on parle aussi de transformation de Gabor) ([3] [19]).

4.5. Formants

On nomme formants les maximums de l'enveloppe de la fonction spectrale à court-terme. Le premier maximum correspondant au *pitch*, seul les deuxième, troisième et quatrième formants sont exploités. Utilisés depuis longtemps pour la reconnaissance de phonèmes [3], ils sont aujourd'hui moins utilisés.

4.6. Banc de filtres

La première application des bancs de filtres a été le spectrogramme qui présente l'énergie à court terme pour différentes bandes de fréquences en fonction du temps (très proche par le concept de la transformée de Fourier à court terme donc). Mais le problème était que la largeur de bande devait être choisie. Une bande étroite permet une bonne résolution harmonique mais génère des imprecisions importantes pour la résolution temporelle. A l'inverse, une bande large permet une bonne résolution temporelle mais une faible résolution harmonique.

Pour compenser ceci et en prenant exemple sur la sensibilité de l'oreille humaine, des bancs de filtres dont les centres suivent une loi non linéaire ont été inventés («non-linear frequency scales»). Le plus connu est probablement le *Mel-Frequency*. Plus la fréquence est élevée, plus la bande passante est large ce qui permet une meilleure résolution temporelle des hautes fréquences.

4.7. Filtrés de Gabor et ondelettes

La transformée de Fourier à fenêtre glissante souffre des mêmes problèmes de résolution que les bancs de filtres dont les fréquences varient linéairement (spectrogramme). Là encore, des méthodes plus récentes comme les filtres de Gabor ou les ondelettes permettent une bonne résolution temporelle à haute fréquence et une bonne résolution harmonique sans grande précision temporelle à basse fréquence.

Pour les filtres de Gabor, une sinusoïde module une fenêtre de forme gaussienne. La formule d'un filtre de Gabor est donc :

$$g(t, \sigma, w_0) = \frac{1}{\sqrt{2\pi\sigma}} \cdot e^{-\frac{t^2}{2 \cdot \sigma^2}} \cdot e^{j \cdot w_0 \cdot t}$$

Pour les ondelettes, les atomes temporels sont plus compliqués (définis comme des filtres passe-bas et des filtres passe-haut complémentaires selon certaines propriétés [4] [19]). Mais à chaque fois, le principe est le même. Depuis la forme de base, appelée fonction mère et notée par exemple $\psi(t)$, la base complète des fonctions $\Psi_{s, \tau}(t)$ (voir [4]) est défini par :

$$\Psi_{s, \tau}(t) = \frac{1}{\sqrt{s}} \cdot \psi\left(\frac{t - \tau}{s}\right)$$

s est le modificateur d'échelle (*scale*) qui dilate ou comprime l'ondelette. τ est le facteur de translation qui sert à déplacer l'ondelette au cours du temps.

4.8. Analyse Cepstrale

Inspiré des traitements homomorphique, l'analyse cepstrale consiste à transformer la convolution temporelle en addition grâce à la fonction logarithme appliquée dans le domaine fréquentiel. En effet, si :

$$x(t) = h(t) \bullet u(t) \Leftrightarrow X(z) = H(z) \cdot U(z)$$

alors

$$\log(X(z)) = \log(H(z)) + \log(U(z))$$

La plupart du temps, l'amplitude est suffisante et la phase n'est pas prise en compte.

Fort heureusement, pour le calcul des coefficients cepstraux, le calcul de deux transformées de Fourier et de la fonction logarithme n'est pas nécessaire. En effet, seuls les premiers coefficients sont intéressants et

ils peuvent aussi être approchés par un banc de filtres non linéaire (MFCC : Mel-Frequency Cepstrum Coefficients) ou calculés depuis les coefficients d'un modèle autorégressif d'ordre p selon la relation de récurrence (voir [3]) :

$$c_k = a_k + \frac{1}{k} \cdot \sum_{i=1}^{k-1} i \cdot c_i \cdot a_{k-i} \quad 1 \leq k \leq p$$

4.9. Objectif : la quantification vectorielle

La quantification vectorielle sur le son à des fins analytiques à déjà été employée.

Dans [7], les auteurs font une quantification vectoriel pour faire de la reconnaissance de mots. Quatre dictionnaires sont créés : un pour le vecteur contenant 12 coefficients ceptraux, un pour le vecteur contenant les dérivées premières des 12 coefficients cepstraux, un pour la dérivée première de la fonction log-énergie et un pour sa dérivée seconde.

Dans [8], un système basé sur des réseaux de neurones pour faire l'estimation objective de la qualité de reconstruction d'un son utilise 14 MFCC. Ayant testé à la fois le MLP et le RBF, les auteurs indiquent une meilleure robustesse pour le RBF, c'est à dire le réseau de neurones utilisant des prototypes.

Dans [10], la quantification vectorielle est testée sur les coefficients d'ondelettes et les LPC (coefficients de la prédiction linéaire). Dans le cas particulier de la reconnaissance de chiffres, la seconde option est meilleure.

Dans [14], un système pour générer 8 paramètres d'animation faciale (définis dans la norme MPEG-4) est présenté. Deux solutions sont envisagées. Dans la première, une quantification vectorielle est faite sur un vecteur contenant les 8 paramètres vidéo et 16 coefficients de prédiction linéaires. La norme cepstrale est utilisée pour la partie son. La seconde solution consiste à utiliser un perceptron multicouche prenant comme entrée 16 coefficients cepstraux calculés depuis le modèle de prédiction linéaire.

Dans [5], 12 MFCC sont quantifiés (via une carte de Kohonen ou un *Neural gas*) pour permettre ensuite l'apprentissage non

supervisé d'un TOM (Temporal Organization Map).

Au final, puisque les coefficients cepstraux, le log-énergie et leurs dérivées sont fournis avec la base Tulips1 et puisque la quantification vectorielle fonctionne entre autres sur ces paramètres, nous avons décidé de les utiliser. Plusieurs possibilités s'offrent toutefois encore à nous. Nous pouvons utiliser les 12 coefficients cepstraux seuls, les 12 coefficients cepstraux et leurs dérivées, les 12 coefficients cepstraux et le log-énergie, ou bien les 26 paramètres au complet.

5. Les tests

Pour avoir une idée des capacités d'apprentissage et de robustesse du STAN en audio, nous avons décidé de tester la reconnaissance de chiffres dans trois types de conditions distinctes : la reconnaissance monolocuteur, la reconnaissance multilocuteur et la reconnaissance sur locuteur inconnu.

5.1. La reconnaissance monolocuteur

Le principe est très simple : on utilise la première séquence pour apprendre et la seconde pour tester. On réitère le principe sur les douze personnes de la base en faisant à chaque fois une quantification vectorielle et un apprentissage du RCE. Les meilleurs résultats ont été obtenus en appliquant la quantification vectorielle sur les 26 paramètres et en cherchant 29 prototypes. Le résultat est de 97,6%, ce qui revient à une seule confusion : pour la personne nommée Isaac, le 4 est reconnu comme un 1. Avec 10 prototypes et l'utilisation de 24 paramètres (12 cepstraux et leurs dérivées), on atteint toutefois déjà 89,6% avec comme matrice de confusion :

		Chiffre reconnu			
		1	2	3	4
Chiffre prononcé	1	12	0	0	0
	2	0	12	0	0
	3	1	1	10	0
	4	3	0	0	9

5.2. La reconnaissance multilocuteur

Le principe ici est d'apprendre dans un même réseau la prononciation de plusieurs personnes. Une séquence pour chaque personne est prise afin de constituer la base d'apprentissage. Les prototypes puis le ST-RCE sont ensuite définis. Les tests se font sur les 12 séquences restantes (une par personne également). Les meilleurs résultats sont obtenus avec 45 prototypes quantifiant les 26 paramètres. Le taux de réussite est de 93,8% avec comme matrice de confusion :

		Chiffre reconnu			
		1	2	3	4
Chiffre prononcé	1	10	0	1	1
	2	0	12	0	0
	3	0	0	12	0
	4	1	0	0	11

Toutefois, avec seulement 25 prototypes, un taux de réussite de 91,7% est déjà obtenu avec la matrice de confusion suivante :

		Chiffre reconnu			
		1	2	3	4
Chiffre prononcé	1	9	1	1	1
	2	0	12	0	0
	3	0	1	11	0
	4	0	0	0	12

5.3. Locuteur inconnu

Le protocole suivi est celui de [11] : 22 séquences sont utilisées pour l'apprentissage (deux séquences par 11 personnes) et les tests sont faits sur les deux séquences de la douzième personne. On réitère douze fois l'opération en changeant la personne inconnue. Les meilleurs résultats ont été obtenus avec 25 prototypes sur les 26 paramètres. Le résultat moyen est de 82,3% avec comme

matrice de confusion :

		Chiffre reconnu			
		1	2	3	4
Chiffre prononcé	1	18	2	1	3
	2	0	22	1	1
	3	2	3	18	1
	4	3	0	0	21

6. Conclusions et perspectives

Dans cet article, nous confrontons l'outil «générateur d'impulsions par quantification vectorielle - STANN» à trois problèmes classiques en traitement de la parole : la reconnaissance de chiffres en conditions monolocuteur, la reconnaissance de chiffres en conditions multilocuteur et enfin la reconnaissance de chiffres par un locuteur inconnu.

Ces premiers résultats de reconnaissance audio par des réseaux de neurones à impulsions sont déjà satisfaisants et nous ouvrent deux voies de travail.

D'une part, tester différents traitements du son pour optimiser la robustesse à la quantification vectorielle, surtout dans le cadre d'un locuteur inconnu. Nous avons utilisé ici les coefficients cepstraux et la fonction log-énergie mais quels seraient les résultats avec une quantification faite sur les coefficients de prédiction linéaire (LPC) ou bien après une transformée par ondelettes comme cela a été testé dans d'autres systèmes ?

D'autre part, puisqu'avec un système parfaitement identique nous réussissons à faire la lecture labiale et la reconnaissance de parole, il serait intéressant de voir si les deux informations peuvent être exploitées conjointement dans un seul et même système faisant un système de reconnaissance bimodal très efficace.

Références

[1] A.R. Baig «Une approche méthodologique de l'utilisation des STAN appliquée à la reconnaissance visuelle de la parole» *Université Rennes I, PhD Report*, 2000.

[2] A.R. Baig, R. Séguier et G. Vaucher «A Spatio-temporal Neural Network applied to visual speech recognition», *ICANN*, 1999.

[3] R. Boite, and M. Kunt «Traitement de la parole», *Presses polytechniques romandes, complément au traité d'électricité*, 1987.

[4] I. Daubechies «Ten lectures on wavelets», *SIAM*, 1992.

[5] S. Durand et F. Alexandre, «Learning Speech as Acoustic Sequences with the Unsupervised Model, TOM», *Proceedings 8th International Conference on neural Networks and their Applications*, 1995

[6] J.L. Flanagan «Speech analysis, synthesis and perception». *Springer-Verlag, 2nd edition*, 1972.

[7] V. Fontaine, H. Leich et J. Hennebert «Influence of vector quantization on isolated word recognition», *Proceedings of EUSIPCO*, pp. 115-118, 1994.

[8] Q. Fu, K. Yi et M. Sun «Speech quality objective assessment using neural network», *ICASSP, vol 3, pages 1511-1514*, 2000.

[9] G.M. Georgiou et C. Koutsougeras «Complex domain backpropagation» *IEEE trans. on circuits and systems - II : Analog and digital signal processing*, Mai 1992.

[10] M. Krishnan, C. Neophytou et G. Prescott «Wavelet transform speech recognition using vector quantization, dynamic time wrapping and artificial neural networks», *Preprint*, 1994.

[11] J. Luettin «Visual Speech And Speaker Recognition», *Université de Sheffield, PhD Report*, 1997.

[12] T. Masters «Signal and image processing with neural networks», *John Wiley & Sons, Inc.*, 1994.

[13] D. Mercier et R. Séguier «Un prétraitement générique pour les STANN : Illustration en lecture labiale», *NSI*, 2000.

[14] S. Morishima et H. Harashima «Speech-to-signal media conversion based on VQ and neural network», *ICASSP, pages 2865-2868*, 1991.

[15] J.R. Movellan «Visual Speech Recognition with Stochastic Networks», in G. Tesauro, D. Toruetzky, & T. Leen (eds.)

Advances in Neural Information Processing Systems, Vol 7, MIT Press, Cambridge, 1995.

[16]N. Mozayyani, A.R. Baig et G. Vaucher «A Fully Neural Solution for on-Line Handwritten Character Recognition», *IJCNN*, 1998.

[17]N. Mozayyani «Introduction d'un codage spatio-temporel dans les architectures classiques de réseaux de neurones artificiels : Application à la reconnaissance de caractères manuscrits», *Université Rennes I, PhD Report*, 1998.

[18]G. Vaucher «A la recherche d'une algèbre neuronale spatio-temporelle», *Université Rennes I, PhD Report*, 1996.

[19]M. Vetterli et J. Kovacevic «Wavelets and subband coding», *Prentice Hall PTR*, 1995.