

# Procesamiento de Señales Usando Redes de Neuronas Artificiales: Aplicación a Señales Biológicas

*Francisco Vaz*

Departamento de Electrónica y Telecomunicaciones / INESC  
Universidad de Aveiro 3800 Aveiro, Portugal

*Traducción y Adaptación: Alejandro Pazos y Javier Pereira*

## Resumen

En este capítulo se hace una breve introducción a los conceptos generales de las redes de neuronas artificiales y a sus aplicaciones en la extracción y clasificación de parámetros característicos (*features*) de datos. Será especialmente considerado el caso en el que los datos en análisis sean señales variables en el tiempo, introduciéndose algunas arquitecturas que permiten la clasificación de patrones temporales y un método de descomposición de señal en componentes ortogonales. Serán también expuestos algunos ejemplos de los métodos anteriormente introducidos aplicados al electroencefalograma (EEG) y al reconocimiento de voz.

## 1.- Introducción

Una red de neuronas [1] es un procesador distribuido y masivamente paralelo propenso, por naturaleza, al almacenamiento de conocimiento experimental volviéndolo apto para su utilización. Se parece al cerebro humano en dos aspectos:

- i) El conocimiento es adquirido a través de un proceso de aprendizaje.
- ii) El conocimiento es almacenado en las uniones entre neuronas, conocidas como pesos sinápticos.

El procedimiento que realiza el proceso de aprender se llama “algoritmo de aprendizaje” y está formado por un conjunto de reglas que permiten la modificación de los pesos sinápticos en orden a obtener los objetivos deseados. La modificación de pesos dentro de una topología fija es la forma tradicional usada en las redes de neuronas y es muy parecida a las metodologías usadas en filtros lineales adaptativos. Sin embargo, en una red de neuronas es posible la modificación de su topología, imitando las capacidades de las neuronas biológicas que pueden alterar el número y la topología de las uniones sinápticas.

Las redes de neuronas presentan diversas propiedades que las vuelven muy interesantes en la resolución de un gran número de problemas. Para los fines que nos proponemos analizar, el procesamiento de señales y su utilización en la clasificación de patrones, los aspectos más interesantes a referir son, sin duda, sus propiedades no lineales que abren nuevas perspectivas al procesamiento de señales y las capacidades de adaptación implícita en el proceso de aprendizaje.

Una red de neuronas artificial está constituida por un conjunto de unidades de procesamiento o neuronas densamente interligadas a través de sinapsis. En la figura 1 se representa un modelo de neurona artificial (o unidad de procesamiento) que está constituida por :

- i) Un conjunto de sinapsis (o uniones) de entrada caracterizadas por sus pesos.
- ii) Un sumador que combina las diferentes entradas teniendo en consideración los respectivos pesos sinápticos.
- iii) Una función de activación que fuerza la respuesta de la neurona a excitaciones recibidas. En general, esta función es no lineal y limita la gama de variación de la salida.

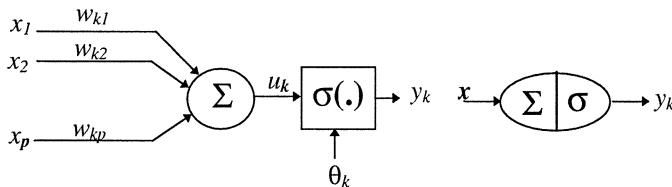


Figura 1.- Modelo no lineal de una neurona.

Analíticamente, una neurona será representada por la siguiente expresión:

$$y_k = \sigma(u_k + \theta_k) = \sigma\left(\sum_{i=1}^p w_{ki}x_k + \theta_k\right) \quad (1)$$

El aprendizaje es el proceso por el cual los parámetros de una red de neuronas son adaptados a través de un proceso de estimulación por el ambiente que envuelve a la red de neuronas. El tipo de aprendizaje está determinado por el modo en que las modificaciones de los parámetros ocurren después de la estimulación.

El proceso de aprendizaje puede ser resumido en:

- i) Estimulación por el medio ambiente (presentación de ejemplos en las entradas)
- ii) La red de neuronas se modifica (los pesos son alterados según determinados criterios)
- iii) La red de neuronas pasa a responder de una nueva manera (el mapeado del espacio de las entradas en el espacio de la salida se altera)

Existen diversos medios (reglas) para efectuar el aprendizaje. Aquí, se utilizan reglas basadas en corrección de errores (entre una salida real y una salida deseada) y reglas de aprendizaje Hebbianas que se basan en la maximización de la correlación entre entrada y salida.

## 2.- Procesamiento temporal usando redes de neuronas artificiales

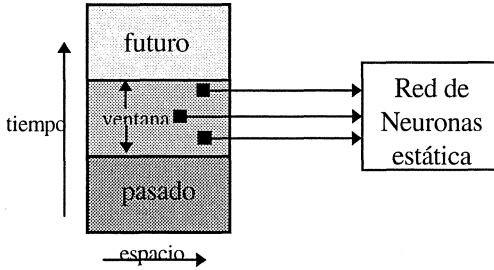


Figura 2.- Utilización de una red de neuronas artificial para procesamiento temporal.

El principal problema en el procesamiento de señales usando las redes de neuronas artificiales reside en la manera de representar la variable tiempo en la estructura de la red. Una forma habitual de tratamiento [1] consiste en hacer una segmentación temporal de la señal (*snapshot representation*) conforme se indica en la figura 2. La señal es segmentada por una ventana temporal y tan pronto como se muestran las entradas dentro de la ventana son representadas a una red estática. La dimensión de la ventana puede incluir a toda la señal con lo que estaríamos en un problema estático, pero también puede suceder que la ventana corresponde a una parte de la señal o incluso que la ventana se reduzca al valor actual de la señal en análisis. La situación intermedia es la más interesante y será la que vamos a explorar. La ventana se desliza sobre la señal; esto es, en cada instante representamos en la red un conjunto de muestras que incluyen a la señal actual y parte de su pasado. La manera tradicional de incorporar el pasado en el modelo de un sistema es incluir memoria en el modelo y, de este modo, pasa a ser capaz de representar los aspectos dinámicos que se verifican en el sistema. Es esta la manera de introducir la variable tiempo en la estructura de las redes de neuronas: la simple introducción de elementos de memoria en las topologías

estáticas. Procediendo de este modo, generalizamos las estructuras estáticas y aprovechamos mucho de lo que fue hecho para este tipo de redes, como por ejemplo los métodos de aprendizaje. De hecho, redes estáticas a las que fueran añadidas líneas de retardo simples pueden ser entrenadas recurriendo a algoritmos que son simples extensiones de los algoritmos usados en el caso estático. Por ejemplo, el algoritmo conocido como *backpropagation through time* es una generalización del algoritmo de retropropagación del error (*backpropagation*).

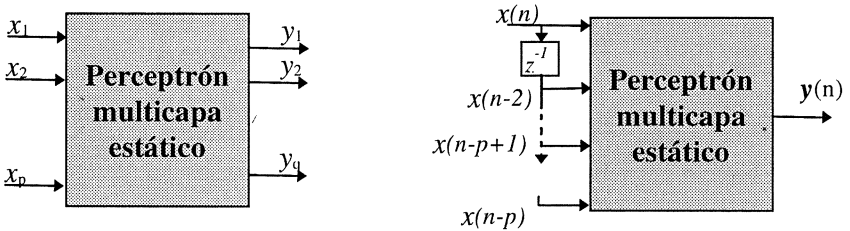


Figura 3.- Perceptrón y TDNN

Una topología muy usada es la red de neuronas con retardos temporales (TDNN-Time Delay Neural Network) que es un perceptrón multinivel en el que fueron incorporadas células de memoria en los diversos niveles. Una TDNN concentrada sólo contiene memoria en el primer nivel (figura 3).

En la figura 4 se indica una generalización posible a esta estructura. De hecho, una TDNN concentrada puede ser interpretada como un modelo MA-moving average-no lineal (de hecho, si no existiera la no linealidad  $\sigma(\cdot)$  la TDNN concentrada se reduciría a un modelo de este tipo). La introducción de una malla de realimentación asociada a una memoria de simples retardos conduce a la segunda estructura indicada en la figura 4, a la que llamaremos, modelo ARMA no lineal.

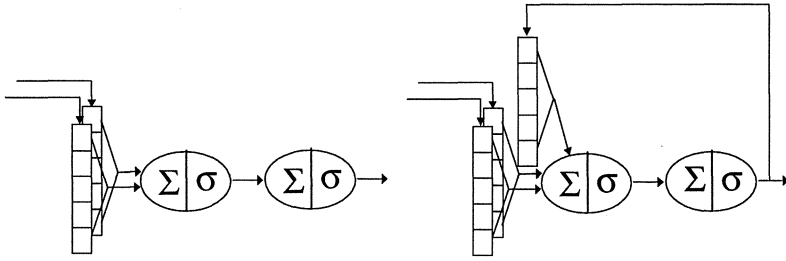


Figura 4.- Modelo MA y ARMA no lineales.

### 3.- La memoria gamma

El término de memoria gamma es usado para indicar la estructura de retardos descrita en la figura 5. En el tiempo esta estructura se describe por la siguiente ecuación:

$$g_0(t) = x(t)$$

$$g_k(t) = (1 - \mu)g_k(t-1) + \mu g_{k-1}(t-1), \quad k = 1, \dots, K \quad (2)$$

La estabilidad de este sistema es trivial, pues, atendiendo a su función de transferencia (3) basta que  $\mu \in [0, 2]$  para que al poner el múltiplo en  $1 - \mu$  esté en el interior del círculo unitario. Nótese que si  $\mu = 1$  la memoria gama se reduce a una simple línea de retardo

$$G(z) = \frac{\mu}{z - (1 - \mu)} \quad (3)$$

Se puede definir una profundidad media de memoria D para una memoria de orden k y probar [2] que está relacionada con el parámetro  $\mu$ :

$$D = \frac{k}{\mu}$$

De esta forma, la profundidad de memoria queda independiente del número de retardos y como  $\mu$  puede ser adaptado a los datos a través de un proceso de aprendizaje [3] pasamos a disponer de una memoria que se adapta a la señal que estamos procesando.

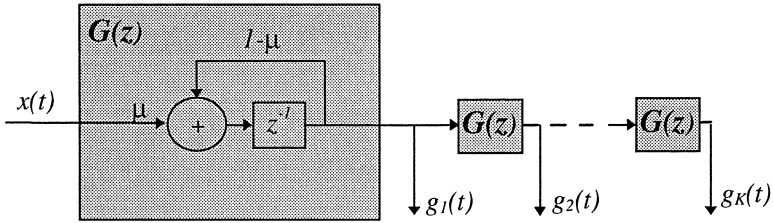


Figura 5.- Estructura de una memoria gamma.

#### 4.- Análisis de componentes principales

Considérese la sinápsis elemental de la figura 6:

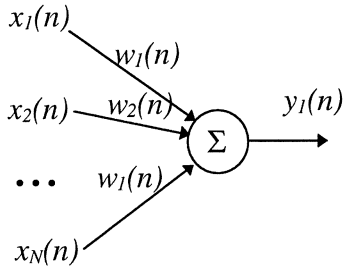


Figura 6.- Sinápsis elemental.

$$y(n) = \sum_{i=1}^N w_i(n)x_i(n) = \mathbf{w}(n)^T \mathbf{x}(n) \quad (4)$$

Se puede aplicar un proceso de aprendizaje no supervisado a esta red de neuronas elemental usando la regla de Hebb para la actualización de los pesos:

$$w_i(n+1) = w_i(n) + \mu y(n)x_i(n) \quad (5)$$

en la que  $\mu$  es la constante de aprendizaje. Esta regla, que procura aumentar la correlación cruzada entre la entrada y la salida, conduce a un

crecimiento no limitado de los pesos. Para solventar este problema es posible usar la siguiente normalización introducida por Oja y que garantiza pesos estables.

$$w_i(n+1) = w_i(n) + \mu y(n)[x_i(n) - y(n)x_i(n)] \quad (6)$$

Si la constante de aprendizaje fuera suficientemente pequeña para que los valores de los pesos se pudieran considerar como un proceso estocástico estacionario, es posible demostrar que:

$$\lim_{n \rightarrow \infty} \mathbf{w}(n) = \mathbf{e}_1 \quad (7)$$

en donde  $\mathbf{e}_1$  es el primer vector propio del vector de entrada  $\mathbf{x} = (x_1(n), \dots, x_N(n))^T$ . En estas condiciones, después de que el proceso de adaptación se haya estabilizado, la salida de esta neurona iguala el valor de la primera componente principal de los datos de entrada.

El método puede ser generalizado para sinapsis múltiples, en orden a obtener diversas componentes principales.

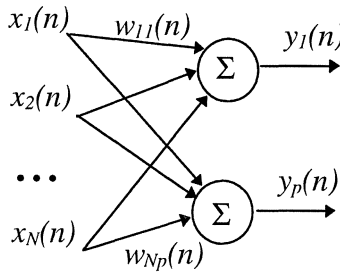


Figura 7.- Sinapsis lineal múltiple.

Definiendo un vector de salidas  $\mathbf{y} = [y_1, y_2, \dots, y_p]^T$  y la matriz de los pesos  $\mathbf{w}(n)$ , continuamos teniendo

$$\mathbf{y}(n) = \mathbf{w}^T(n)\mathbf{x}(n) \quad (8)$$

Usando la siguiente regla de actualización de los pesos, expuesta por Sanger, y si las mismas condiciones se verifican, las componentes de  $\mathbf{y}$



son las  $p$  primeras componentes principales de  $\mathbf{x}$  y las columnas de  $\mathbf{w}$  los respectivos vectores propios.

$$\Delta w_{ij}(n+1) = \mu \left[ y_j(n)x_i(n) - y_j(n) \sum_{k=1}^{j+1} w_{ij}(n+1)y_k(n) \right] \quad (9)$$

Se discutió, hasta ahora, el caso estático. Si se tuviera que trabajar con señales variables en el tiempo, se podría aplicar un procedimiento similar al que anteriormente nos hemos referimos para la introducción de dinámica en las redes: introducimos una línea de retardo en la entrada de la sinapsis, que de esta forma procesará las muestras de la señal contenidas en una determinada ventana temporal. El vector de entrada será ahora:

$$\mathbf{x}(i) = [x(i), x(i-1), \dots, x(i-N+1)]^T$$

y, si la señal es estacionaria, en la salida se obtendrán las componentes principales del pasado reciente de la señal. Atendiendo a una importante prioridad de las componentes principales que deben ser ortogonales, tenemos en la salida una red de descomposición de la señal de entrada en señales ortogonales, cada una de las cuales puede ser vista como una versión filtrada de la señal de entrada obtenida por un filtro lineal adaptativo. Este banco de filtros procura descomponer la señal en señales no correlacionadas, siendo ésta la propiedad más interesante de esta descomposición, que será aprovechada para usar dicha descomposición para la clasificación de las señales.

En las figuras 8 y 9 se representa un ejemplo de aplicación de este método a una señal estacionaria con dos resonancias espectrales en  $f_s/8$  e  $f_s/4$ . Se usaron 10 retardos en la entrada y fueron calculados 6 componentes principales usando  $\mu = 0,01$  (fuera apenas se muestran tres de las componentes). Es posible observar como la red consigue separar los dos ritmos principales presentes en esta señal.

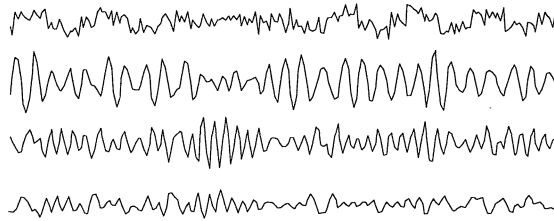


Figura 8.- Señal y las tres primeras componentes principales.

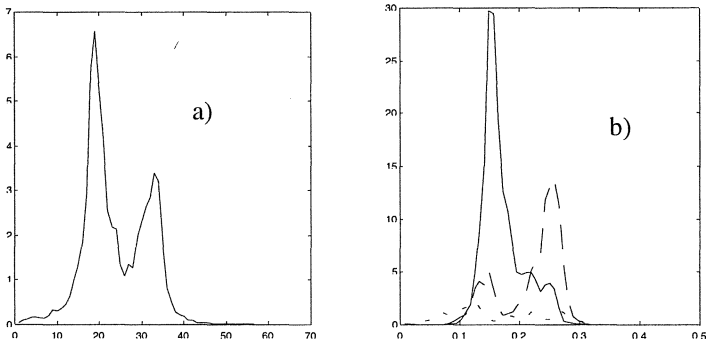


Figura 9.- a) espectro de la señal. b) espectro de las componentes principales.

## 5.- Ejemplos de Aplicación

### 5.1.- Detección de “spikes” en el EEG

El análisis del electroencefalograma (EEG) continúa siendo el principal medio no invasivo para el estudio de la epilepsia. Los enfermos epilépticos presentan un EEG, entre crisis, con ondas de estructura especial, normalmente caracterizadas por picos (*spikes*) u ondas abruptas (*sharpwaves*) que sobresalen claramente de la actividad de fondo. La definición de *spike* también puede encontrarse en [4] (Un “spike” es una

señal con una duración entre 20-70 msg que destaca claramente del trazado de actividad de fondo).

El EEG es la señal resultante de la actividad bioeléctrica del cerebro recogido sobre la calota craneal. A veces, cuando la cirugía es posible y aconsejable, se implantan electrodos en el cortex cerebral para la recogida directa de la actividad eléctrica. En esta situación, la señal recogida está designada por electrocorticograma (ECoG)

Un examen de EEG (ECoG) consiste en la recogida durante largos períodos de estas señales medidas simultáneamente en varios canales. La introducción de medios automáticos para la detección de *spikes* es, pues, una necesidad resultante de la enorme cantidad de datos generados en un examen y de su definición bastante imprecisa.

Una búsqueda bibliográfica muestra que muchos detectores automáticos de *spikes* han sido propuestos con mayor o menor éxito [5-6]. Las redes de neuronas artificiales también han sido ya usadas para este fin [7-9], pero en todos los casos la metodología usada era el reconocimiento de patrones: la señal era analizada para extraer parámetros característicos que, a continuación, eran presentados a una red de neuronas artificiales entrenada como clasificador.

Las metodologías presentadas anteriormente nos permiten proponer un nuevo método [10-11]. La señal EEG es presentada directamente a una red de neuronas artificiales del tipo MA o ARMA no lineal que tiene una memoria capaz de almacenar un *spike* y que fue entrenada de acuerdo con un proceso que se esquematiza en la figura 10. Su salida, normalmente a cero, pasará a 1 siempre que en la memoria estuviera almacenada una onda con forma de *spike*.

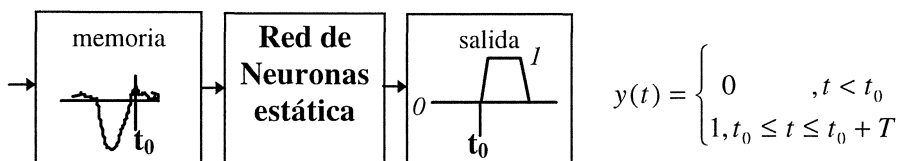


Figura. 10.- Entrenamiento de una red de neuronas artificiales para reconocer patrones temporales.

La señal ECoG utilizada fue mostrada a 250 Hz y digitalizada con un ADC de 12 bits de resolución. A la señal así recogida le era retirada la componente de continua y normalizada a través de una simple transformación lineal que transforma el intervalo  $[m-3v, m+3v]$  en  $[-0.5, 0.5]$ , en donde  $m$  y  $v$  son respectivamente el valor medio y el error cuadrático medio en el intervalo de análisis.

Para la caracterización de la señal fue usado un método de descomposición en componentes principales representando también una caracterización mas tradicional usando la propia señal y su derivada (estimación filtrando la señal por un filtro FIR con ceros en  $\pm 1, \pm j$  e  $\sqrt{2}(-1 \pm j)$ ).

La señal del EEG no es estacionaria, por lo que la aplicación del método propuesto de descomposición en componentes principales debe ser hecho con cuidado, así como la interpretación de los resultados. Desde que se escoge una constante de aprendizaje conveniente, la descomposición es válida para la actividad de fondo, que se comporta como una señal quasi-estacionaria. Un *spike* es una señal no estacionaria sobre esta actividad de fondo, y la descomposición deja de tener sentido. Sin embargo, en más de una ocasión, si la constante de aprendizaje está bien seleccionada, durante el acontecimiento del *spike* los filtros que sirven para descomponer la señal en componentes ortogonales no se alteran significativamente, y la parte no estacionaria de la señal continúa a ser descompuesta en componentes ortogonales. Un ejemplo de descomposición de ECoG es representada en la figura 11.

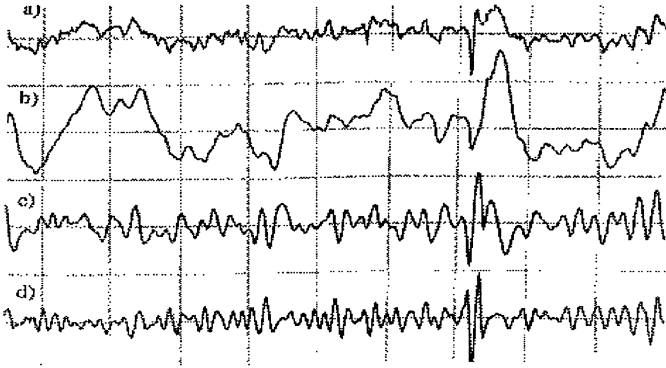


Figura 11.- Descomposición de la señal ECoG en componentes ortogonales.

Cuatro redes diferentes han sido construidas y entrenadas utilizando el simulador NeuroSolutions [12]:

- NN1: red con 20 retardos, una capa oculta, 16 retardos en la malla de realimentación, una capa oculta con 8 unidades y entrenada con una señal y su derivada.
- NN2: red recurrente con 20 retardos, 16 retardos en la malla de realimentación, una capa oculta con 8 unidades y entrenada para una señal y su derivada.
- NN3: red con 20 retardos, una capa oculta con 6 unidades y entrenada con 3 componentes principales.
- NN4: red recurrente con 20 retardos, 16 retardos en la malla de realimentación, una capa oculta con 8 unidades y entrenada con 3 componentes principales.

Los datos disponibles para probar este método están constituidos por un conjunto de entrenamiento con 59 *spikes* y un conjunto de test con 49 *spikes*, ambos construidos por selección visual en la pantalla de un ordenador y concatenados en segmentos.

Se han obtenido los siguientes resultados para la detección de *spikes*

TABLA 1

	Conjunto de entrenamiento			Conjunto de test	
	Detecciones	Negativos falsos	Falsos Positivos	Detecciones	Negativos falsos
NN1	49	0	0	57	2
NN2	49	0	0	58	1
NN3	46	3	0	52	7
NN4	48	1	0	56	3

Se puede verificar que, las redes recurrentes, se comportan mejor, así como las redes que procesan la señal y su derivada. También se puede ver que todas las redes se comportan mejor en el conjunto de test que en el de entrenamiento. Tal hecho se debe a la selección demasiado exigente de ejemplos para el conjunto de entrenamiento: algunos de los “*spikes*” seleccionados eran eventos de muy baja amplitud, lo que dificulta mucho su clasificación.

Los resultados de la caracterización con componentes principales son ampliamente compensados por la facilidad de ser calculados en “tiempo real” por el método representado.

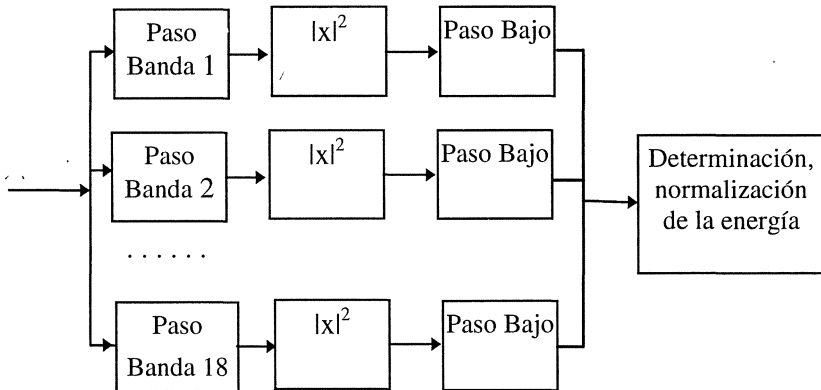
Han sido testeadas las características de generalización sobre los datos de otros pacientes que indicaron, como era de prever, que no es fácil la generalización. Sin embargo fue posible verificar que las redes entrenadas con datos de un paciente eran fácil y rápidamente re-entrenadas con datos de otro.

## 5.2.- Reconocimiento de la voz.

Siendo la voz el medio de comunicación por excelencia del ser humano, la introducción de técnicas directamente basadas en la interacción de la voz con los ordenadores, es un asunto de gran importancia en el desarrollo de numerosas aplicaciones. Es conocida la

facilidad que tenemos de reconocer a una persona a través de la voz. A continuación presentamos, resumidamente, el método y algunos resultados conseguidos en el área del reconocimiento de la voz usando redes de neuronas artificiales.

La señal de la voz es descompuesta usando un banco de filtros desde la banda de los 200 Hz hasta los 3.300 Hz, compuesto por 18 filtros paso-banda GammaTone[14] calculados de acuerdo con el trabajo realizado de Moore y Glasberg[14]. Para cada filtro, es estimada la energía de acuerdo con la figura 12 y, a continuación, es hecha una descomposición, después de los filtros paso bajo, para obtener una frecuencia de muestreo de 30 Hz.



*Figura 12.- Procesamiento de la señal de la voz para su caracterización.*

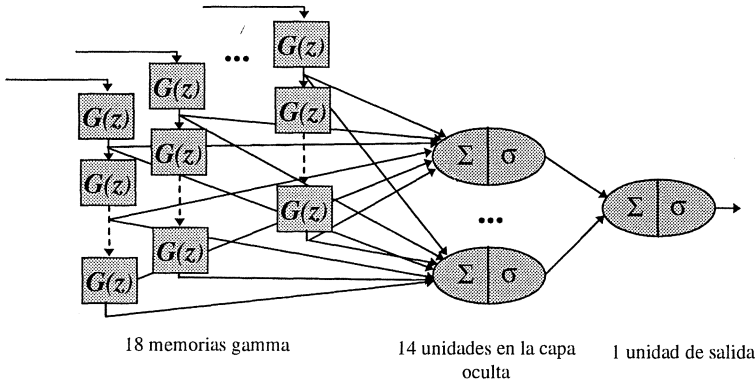
Para una frase dada, la salida del banco de filtros es una matriz en la que una dimensión es el orden de los filtros y la otra el tiempo.

Después de efectuar esta caracterización de la voz, se entrenó una red de neuronas artificiales constituida por una primera capa de unidades dotadas de memoria gama (en número igual al número de filtros), una capa oculta con 14 a 16 unidades, y una capa de salida con un número de elementos dependiente de la aplicación.

Fueron desarrolladas dos aplicaciones: la identificación del orador y la verificación del orador. La identificación del orador consiste en descubrir

dentro de una población de oradores aquellos que tienen un patrón de voz más semejante al orador en evaluación. La verificación del orador es una tarea mas simple, pues sólo se pretende verificar si dado un patrón de habla y la identificación del orador, el patrón es suficientemente semejante al patrón de referencia asociado a la identidad reclamada.

En la tarea de verificación se entrenó una red para cada cliente del sistema. El entrenamiento fue hecho de forma que se active la salida a 1 siempre que en la entrada se presentase un patrón del cliente correcto. El entrenamiento incluyó, además, la representación de patrones de impostores para aumentar la robustez del sistema. En la siguiente figura se esquematiza la red utilizada.



*Figura 13.- Red utilizada para la verificación del orador.*

En las pruebas efectuadas usamos 5 clientes y 10 impostores, cada uno pronunciando algunas decenas de veces la frase “Universidade de Aveiro”, escogida por representar una buena variedad fonética de la lengua portuguesa. Los datos fueron recogidos en un laboratorio con ruido ambiente moderado y la segmentación fue hecha usando un editor de señal.

Los resultados que se obtuvieron fueron de un 77% de aciertos con 4.5% de falsos negativos y 18% de falsos positivos. Considerando los clientes aisladamente, los resultados muestran que el mejor acierto que se consigue es de un 93.5% y el peor de cerca del 74%.



También se realizaron pruebas de identificación del orador. Para esta prueba la red anterior fue modificada incluyendo 5 unidades en la salida (una para cada cliente) y se siguió un procedimiento de entrenamiento semejante. Las pruebas efectuadas llevaron al resultado global de 72% de acierto en la identificación. Individualmente, el acierto varió del 40% al 90% y los impostores del conjunto de entrenamiento fueron rechazados un 82% (resultado que descendió a un 52% para impostores fuera del conjunto de entrenamiento).

Los resultados en la verificación pueden ser considerados buenos atendiendo a las limitaciones en que discurrió el trabajo: existencia de ruido, segmentación poco precisa y base de datos muy pequeña. En la identificación, los resultados son claramente peores, debidos tal vez a la pequeña cantidad de datos, sobre todo de datos relativos a impostores.

Sin duda, son las propiedades de la memoria gamma las que permitirán sobrepasar los dos primeros problemas. La adaptación a la profundidad más conveniente permite una razonable imprecisión en la segmentación y en su respuesta paso bajo aumentando su robustez con la presencia del ruido.

Los resultados detallados pueden ser encontrados en [15]

## Referencias

- [1] Haykin, S. :*Neural Networks: A Comprehensive Foundation*. MacMillan College Publ. Comp. Inc., New York, 1994
- [2] J.C. Principe, B. de Vries: The gamma filter- A new class of adaptive iir filters with restricted feedback *IEEE Transactions on Signal Processing*, 1993.
- [3] T. Oliveira e Silva: Generalized feed-forward filters: Some theoretical results. Proceedings of ICASSP, p.109-112, 1993.
- [4] Ktonas, P.Y.: Automated Spike and Sharp Wave Detection. in *Methods of Analysis of Brain Electrical and Magnetic Signals*. Elsevier Science Publishers, Amesterdam, 1987.
- [5] Smith, J.: Automatic Analysis and Detection of EEG *Spikes*. *IEEE Transactions on Biomedical Engineering*, Vol. BME-21:1-7, Jan, 1974[1]

- [6] Lopes da Silva, F., Van Hulten, K., Lommen., J:G., Storm Van Leeuwen, Van veelen, C.W., Vliegenthart. W.: Automatic Stection and Localization of Epileptic Focci. *Electoencephalography and Clinical Neurophysiology*, 1977, Vol.43:1-13.
- [7] Eberhart, R.C., Dobbins, R.W., Webber, W.R.S.: EEG Analysis using CaseNet. Proceedings of IEEE-EMBS 11th Annual International Conference, Seattle, WA. 1989, 2046-2047
- [8] Espinosa, E.I., Quiza, T.J., Ayhlon, M.A.: Detection and Classification of Neuronal *Spikes* usig DSP Chip and a Neural Network. Proceedings of IEEE-EMBS 13h Annual International Conference, Orland, FL., 1991, 1448-1449.
- [9] Jando G., Siegel, R.M., Horvath, Z., Buzsaki, G.: Pattern Recognition of the EEG by artificial neural networks. *Electoencephalography and Clinical Neurophysiology*, 1993, vol.86:100-109.
- [10] Vaz, F., Príncipe, J.C., Zahalka, A.: Automatic EEG Spike Detection Using Neural Networks..... Santiago
- [11] Vaz, F., Príncipe, J.C.: Neural Networks for EEG Signal Decomposition and Classification... Canadá
- [12] Neuro Solutions. Neuro Dimension Inc. Gainesville, FL, USA, 1994
- [13] John Holdsmith, Ian Nimmo Smith, Roy Patterson, Peter Rice: *Implementing a GammaTone Filter Bank, Annex C of the SVOS Final Report (Part A: the Auditory Filter Bank)*, Fev. 1988
- [14] B.C.J. Moore, B.R. Glasberg: *Cochlear Modeling*, J. Acoust. Soc. Am., 74, p750-753, 1983.
- [15] A.J.S. Teixeira: *Reconhecimento do Orador com Redes Neurais*. Dissertação para obtenção do grau de Mestre. Aveiro, Setembro, 1993