RECONOCIMIENTO DE COMANDOS DE VOZ USANDO LA TRANSFORMADA WAVELET Y MÁQUINAS DE VECTORES DE SOPORTE

RESUMEN
Este artículo muestra el análisis comparativo de un clasificador por red neuronal artificial frente a una máquina de vectores de soporte para una aplicación de reconocimiento de comandos de voz, cuya extracción de características está basada en paquetes wavelet. Para evaluar el desempeño de ambos sistemas, se realizaron pruebas variando el esquema de extracción, diferentes arquitecturas de la red neuronal artificial y diferentes funciones núcleo (kernel) para las máquinas de vectores de soporte. Se encuentra que ambos esquemas de clasificación presentan desempeños similares, con porcentajes de acierto superiores al 96%.

PALABRAS CLAVES: Reconocimiento de Voz, Paquetes Wavelet, Máquinas de Vectores de Soporte, Redes Neuronales Artificiales.

ABSTRACT
In this paper, the comparison between an artificial neuronal network (ANN) and a support vector machine (SVM) for a speech recognition application is presented. Both learning machines were trained to learn speech commands by using a feature extraction system based on wavelet packets. The performance of these classifiers was evaluated under different conditions in feature extraction system, ANN architecture and SVM kernel. A similar recognition rate for both classifiers was found, which is above 96%.

KEYWORDS: *Speech Recognition, Wavelet Packet, Support Vector Machine, Artificial Neuronal Network.*

JORGE I. MARÍN
Lic. Electricidad y Electrónica, Msc.
Profesor Asistente
Universidad del Quindío
jorgemarin@uniquindio.edu.co

PABLO A. MUÑOZ
Ingeniero Electrónico, Estudiante Maestría en Ingeniería Eléctrica – FIE.
Profesor Ocasional
Universidad del Quindío
pabloandresm@yahoo.com

FRANCISCO J. IBARGÜEN
Ingeniero Electricista, Msc.
Profesor Auxiliar
Universidad del Quindío
fjibarg@yahoo.com

Grupos GAMA y GDSPROC
CEIFI, Facultad de Ingeniería
Universidad del Quindío.

1. INTRODUCCIÓN

El reconocimiento de voz ha sido un problema de gran interés para muchas áreas de investigación como el procesamiento de señales y los sistemas inteligentes. Dentro de estos últimos se destaca el área del reconocimiento de patrones. Para poder reconocer un comando de voz es necesario disponer de un sistema que procese la señal de voz y extraiga las características más particulares de ella, con estas características se procede a entrenar una Máquina de Aprendizaje o clasificador para que adquiera la capacidad de identificar un comando de voz independiente de la persona que lo pronuncie. El esquema de extracción de características de más amplia difusión en la literatura es el esquema de cálculo basado en transformada de Fourier de corto tiempo (STFT) y los coeficientes cepstrales de Mel (MFCC) [13]. Con el desarrollo de las técnicas de análisis por medio de la transformada wavelet se propusieron diferentes esquemas para sustituir la STFT por una transformada wavelet. De las diferentes versiones de esta transformada, los sistemas de extracción más exitosos, por desempeño y bajo costo computacional, han sido los basados en la transformada discreta wavelet (DWT) y los paquetes wavelet (PW) [7,8,15,16,17]. Aunque se ha propuesto diversos esquemas basados en paquetes wavelet, la mayoría de éstos coinciden en emplear una estructura del árbol cuya respuesta en frecuencia se aproxima a de un banco de filtros de escalas de Mel [15,16,17]. Estos esquemas, han sido combinados con clasificadores como las redes neuronales artificiales (ANN por sus siglas en inglés de *Artificial Neural Network*) y análisis de discriminantes lineales, obteniéndose en aplicaciones para el reconocimiento de comandos, un porcentaje de acierto del 85% para la base datos NIST [17] y 90% para la base de datos TI64 [16], respectivamente. Para el caso de clasificadores como las máquinas de vectores de soporte (SVM por sus siglas en inglés *Support Vector Machines*) se encuentran pocos reportes que los combinen con esquemas basados en paquetes wavelet [6]. Las SVM han sido utilizadas para el reconocimiento de vocales con un rendimiento del 71.72% [10] y 85.13% [11], ambas usando la base de datos TIMIT; algunos resultados también han sido presentados en el reconocimiento de fonemas con un rendimiento del 77.6% [11] y en el reconocimiento de palabras se han presentado resultados con un rendimiento del 88.4% [12], con la base de datos OGI alphadigit.

Se desea entonces comparar el rendimiento, en el reconocimiento de comandos de voz, entre una ANN y una SVM utilizando un esquema de extracción de características basado en paquetes wavelet, y usando una base datos propia en idioma español.

Fecha de Recepción: 31 Enero de 2006
Fecha de Aceptación: 02 Junio de 2006

2. MARCO TEÓRICO

2.1. Máquinas de Vectores de Soporte (SVM)

Las SVM inicialmente fueron desarrolladas por Vapnik y su grupo de colaboradores en los Laboratorios Bell AT&T [5] y presentadas como una novedosa técnica para clasificación. Estos algoritmos de aprendizaje pueden ser considerados como técnicas alternativas para el entrenamiento de clasificadores con Funciones de Base Radial o Funciones Polinomiales, entre otros. Una de las principales ideas detrás de esta técnica es la de separar las clases por medio de una superficie que maximice el margen entre ellas, a diferencia de las técnicas usadas para el entrenamiento de las ANN, las cuales buscan una superficie que separa las clases con el menor número de errores de entrenamiento [5].

Las SVM a diferencia de las ANN fueron desarrolladas de forma inversa, es decir, primero se fundamentaron teóricamente para finalmente llegar a su implementación y aplicación, mientras que las RNA siguieron un camino heurístico, desde las aplicaciones y extensiva experimentación a la teoría [5]. El desarrollo teórico de las SVM se apoya en la teoría del aprendizaje estadístico desarrollada por Vapnik, quien plantea la existencia de dos ramas dentro del análisis de los procesos de aprendizaje, una de ellas es el análisis aplicado, que plantea que para obtener una buena generalización basta con determinar los parámetros libres de la máquina de aprendizaje para los cuales se obtiene el menor error de entrenamiento. Al principio inductivo que fundamenta este análisis se le conoce como principio de Minimización del Riesgo Empírico (ERM: *Empirical Risk Minimization*), las ANN son un ejemplo claro de una máquina de aprendizaje que utiliza este principio. La otra rama dentro del análisis de los procesos de aprendizaje es el denominado análisis teórico. Allí se plantea que es necesario justificar que el mínimo error de entrenamiento logra una buena generalización, por lo tanto debe existir un principio inductivo que permite controlar y mejorar la habilidad de generalización de la máquina de aprendizaje. A este principio se le denominó Minimización del Riesgo Estructural (SRM: *Structural Risk Minimization*). Este principio es el gran aporte de Vapnik y Chervonenekis al análisis de los procesos de aprendizaje [1,2].

Las SVM realizan una transformación del espacio de entrada a un espacio de características de dimensión mayor, posiblemente infinita, donde un problema de separación no lineal se convierte en un problema de separación lineal [1,2] (figura 1), esto se logra realizando una transformación por medio de la aplicación de una función núcleo (kernel) [5]. El principio SRM depende de un parámetro denominado dimensión VC (por sus autores Vapnik-Chervonenekis), que mide la complejidad del espacio de hipótesis (un espacio de Hilbert) donde se encuentra del conjunto de hiperplanos que podrían realizar la separación de las clases con un margen máximo. Este parámetro es difícil de determinar, por lo tanto la implementación del principio SRM se dificulta. Como solución, surge el algoritmo de las SVMque obtiene el valor de la dimensión VC óptimo y logra minimizar el error de entrenamiento al mismo tiempo [5].

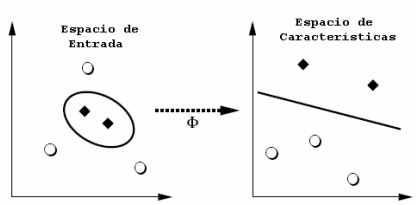

Figura 1. Transformación de un espacio de entrada a un espacio de características. Tomado de [4]

2.1.1 Clasificador No Lineal

Un problema de clasificación no lineal se puede apreciar en la figura 2.

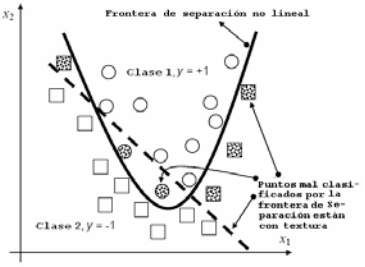

Figura 2. Clasificador No Lineal. Tomado de [4].

Los patrones mostrados en la figura 2 no pueden ser separados linealmente, como lo muestra la línea recta a trazos, pero son separados sin error por una función cuadrática. Este ejemplo es relativamente sencillo, pero si se piensa en problemas de clasificación que presentan características altamente no lineales, como por ejemplo la letra manuscrita, la voz, los rostros, etc; la superficie de separación o el hiperplano de separación se convierte en una curva más compleja de obtener. Para solucionar este problema es necesario determinar un espacio de hipótesis que contenga los hiperplanos que puedan separar las diferentes clases con un margen máximo y un error de entrenamiento mínimo.

El algoritmo para establecer el hiperplano de separación de margen máximo busca determinar los pesos de la máquina de vectores de soporte. Esto se logra resolviendo un problema de programación cuadrática con restricciones lineales de igualdad y desigualdad. Inicialmente se debe minimizar la distancia de los patrones más cercanos al hiperplano de separación, como se muestra en la figura 3.

Se debe minimizar la norma del vector de pesos del hiperplano de separación, esta minimización equivale a maximizar el margen M de separación entre las clases,

margen que es establecido por los vectores de soporte (patrones que están más cerca del hiperplano de separación, figura 3), para lograr esto es necesario:

$$\min \frac{1}{2}\|\mathbf{w}\|^2 + \mathbf{C}$$
$$\textbf{sujeto a } \mathbf{y_i}(\mathbf{w}\cdot\mathbf{x_i}+\mathbf{b}) \geq 1-\boldsymbol{\xi}_\mathbf{i} \quad \mathbf{i}=1,\ldots,\ell \qquad (1)$$
$$\boldsymbol{\xi}_\ell \geq 0 \qquad \mathbf{i}=1,\ldots,\ell$$

donde ℓ es el número de patrones de entrenamiento, C es un parámetro de penalización que realiza un balance (*trade-off*) entre el máximo margen y el número de puntos de datos mal clasificados y la variable ξ es una medida de la distancia de los patrones que están dentro del margen de separación, lo cual se puede observar en la figura 4 [5].

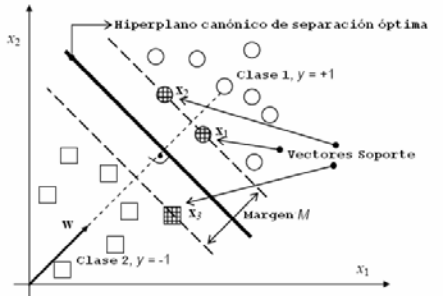

Figura 3. Clasificador de margen máximo en un problema linealmente separable. Tomado de [4].

La ec. (1) es un problema clásico de optimización cuadrático con desigualdad restrictiva, que se resuelve con técnicas de optimización estándar de Programación Cuadrática. La solución a este problema está determinado por el punto de silla del siguiente funcional de Lagrange:

$$\mathbf{L}(\mathbf{w},\mathbf{b},\boldsymbol{\alpha}) = \frac{1}{2}\mathbf{w}^\mathbf{T}\mathbf{w} - \sum_{\mathbf{i}=1}^{\ell}\boldsymbol{\alpha}_\mathbf{i}\left[\mathbf{y}_\mathbf{i}\left[\mathbf{w}^\mathbf{T}\mathbf{x}_\mathbf{i}+\mathbf{b}\right]-1+\boldsymbol{\xi}_\mathbf{i}\right]+\mathbf{C} \quad (2)$$

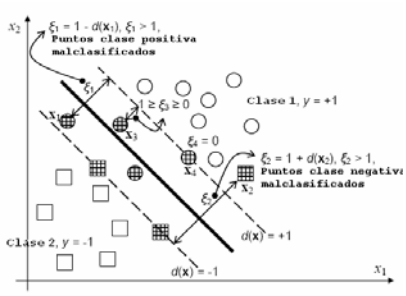

Figura 4. Clasificador de margen máximo en un problema no separable linealmente. Tomado de [4]

Los $\boldsymbol{\alpha}_\mathbf{i}$ son los multiplicadores no negativos ($\boldsymbol{\alpha}_\mathbf{i}>\mathbf{0}$) de Lagrange y son aquellos parámetros asociados a los llamados vectores de soporte, los cuales corresponden a aquellos patrones que se encuentran sobre o dentro del margen de separación de las clases. Para determinar estos parámetros es necesario encontrar el punto de silla del funcional de Lagrange, por lo tanto se debe maximizar (3) con respecto a $\boldsymbol{\alpha}_\mathbf{i}$:

$$\mathbf{L_d}(\boldsymbol{\alpha}) = \sum_{\mathbf{i}=1}^{\ell}\boldsymbol{\alpha}_\mathbf{i} - \frac{1}{2}\sum_{\mathbf{i},\mathbf{j}=1}^{\ell}\boldsymbol{\alpha}_\mathbf{i}\boldsymbol{\alpha}_\mathbf{j}\mathbf{y}_\mathbf{i}\mathbf{y}_\mathbf{j}\mathbf{K}(\mathbf{x_i},\mathbf{x_j}) \qquad (3)$$

Sometido a las restricciones de la ec. (4), las cuales se obtuvieron después de minimizar (2) con respecto a ***w*** y **b** (es el bias asociado al hiperplano de separación)**:**

$$\sum_{\mathbf{i}=1}^{\ell}\boldsymbol{\alpha}_\mathbf{i}\mathbf{y}_\mathbf{i} = 0 \quad ; \quad 0 \leq \boldsymbol{\alpha}_\mathbf{i} \leq \mathbf{C} \quad \mathbf{i}=1,\ldots,\ell \qquad (4)$$

Los multiplicadores de Lagrange estarán acotados por encima por la constante ***C***, con esto se puede limitar la influencia de los datos de entrenamiento que están en el lado equivocado de la superficie de separación no lineal. Finalmente, el hiperplano que determina la pertenencia de los datos a su clase correspondiente está dada por:

$$\mathbf{f}(\mathbf{x}) = \mathbf{sign}\left(\sum_{\mathbf{i}=1}^{\ell}\mathbf{y}_\mathbf{i}\boldsymbol{\alpha}_\mathbf{i}^*\mathbf{K}(\mathbf{x_i},\mathbf{x_j})+\mathbf{b}^*\right) \qquad (5)$$

con $\mathbf{K}(x_i,x_j)$ es la función núcleo o kernel.

2.2 Transformada Wavelet

El propósito de la Transformada Wavelet (TW) es la descomposición de una señal $x(t)$ en una combinación lineal de versiones dilatadas y desplazadas de la función madre $\Psi(t)$, lo cual se denota a través de [9]:

$$X(\tau,a) = \frac{1}{\sqrt{a}}\int x(t)\Psi^*{}_{\tau,a}(t)dt \qquad (6)$$

con

$$\Psi^*{}_{\tau,a}(t) = \Psi^*\left(\frac{t-\tau}{a}\right) \qquad (7)$$

τ corresponde al desplazamiento de la Wavelet madre y *a* es la respectiva escala. Entre los conjuntos de Wavelets más usados están la Haar, Morlet, Daubechies y Coifman [9]. Sin embargo, para el reconocimiento de voz, se han empleado típicamente la Morlet [8] y la Daubechies [7]. La expresión de la ec. (6), también denominada Transformada Wavelet Continua (CWT), tiene dos inconvenientes: el primero consiste en que el conjunto de wavelets de análisis no necesariamente tienen que cumplir con las condiciones de ortogonalidad y soporte compacto (señal de energía y media cero) y segundo, la complejidad computacional de la ec. (6) es muy alta [9]. Para tal efecto, se propone el uso de una Transformada Wavelet Discreta (DWT) [9] cuyas relaciones en el tiempo (m=0,1,2,...) y escala (n=0,1,2,...) son diádicas:

$$\Psi_{m,n}(k) = 2^{-n/2}\Psi(2^{-n/2}k - mb_0) \qquad (7)$$

En la ec. (7) el factor de escala *a*, de la CWT, se ha sustituido por una potencia de dos. De esta forma, el eje de frecuencias corresponde a una descomposición en

octavas. Además, la ec. (7) permite obtener lo que se denomina un análisis multirresolución y puede ser implementada eficientemente por medio de la Transformada Wavelet Rápida (FWT) [9]. Dicho cálculo puede ser visto como el resultado de la aplicación de un banco de filtro de octavas (Figura 5), donde h_0 corresponde a la respuesta al impulso de un filtro pasabajos y h_1 al de un pasaaltos. Las respuestas al impulso de estos filtros FIR dependen de la wavelet madre que se escoja en particular.

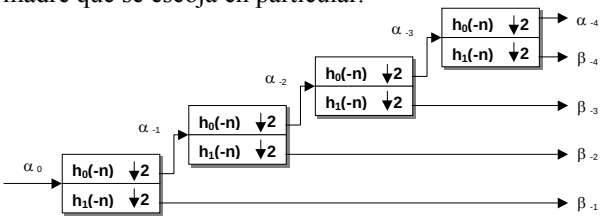

Figura 5. Algoritmo de Mallat

Las señales obtenidas a la salida del conjunto filtro pasabajos y diezmador se conoce con el nombre de aproximaciones (α), y la del conjunto filtro pasaaltos y diezmador, los detalles (β). En la DWT, por cada nivel, solamente la señal a salida de las aproximaciones es la que se somete al particionamiento medio la aplicación del filtrado y diezmado. Cuando se realiza una partición tanto a las aproximaciones como a los detalles, se obtiene lo que se denomina un Paquete Wavelet (PW) [9], que permite obtener una mejor representación de la señal.

3. SISTEMA PROPUESTO

Con el fin de realizar un análisis comparativo del desempeño de un clasificador basado en ANN frente a uno basado en una SVM para una aplicación de reconocimiento de comandos de voz, se exploraron diferentes esquemas de extracción de características basados en el uso de paquetes wavelet. Para validar su desempeño, se construyó una base de datos propia, recolectando para ello 113 muestras de voz de diferentes personas, hombres y mujeres, que pronunciaron los dígitos uno al cinco en idioma español. Estas muestras fueron digitalizadas a 16 bits por muestra y a una frecuencia de muestreo de 8kHz, que corresponde a la calidad telefónica. De las muestras recolectadas, el 90% de ellas fueron empleadas para el entrenamiento de las máquinas de aprendizaje y el 10% restante para la validación.

Previo a la extracción de características se realiza un filtrado de preénfasis [13], que elimina los efectos del tracto vocal, un proceso de segmentación, con el fin de identificar el inicio de la señal y finalmente la normalización de la energía de la señal segmentada. La segmentación se realiza por medio de una técnica que combina la detección del inicio del segmento por medio de un umbral de energía y el número de cruces por cero en un bloque de 5ms. Una vez se detecta el inicio de la señal, ésta se subdivide en bloques de 64ms sin traslape que posteriormente son procesados por un paquete wavelet. Cabe indicar que las condiciones iniciales de los filtros FIR, empleados en el cálculo del paquete wavelet, no son inicializadas al procesar un nuevo bloque, esto con el fin de garantizar una interdependencia entre los bloques que conforman la señal.

Para cada uno de los paquetes wavelet, se empleó la wavelet madre Daubechies con los órdenes 4 y 8, dado que en trabajos previos éstas han demostrado tener un mejor desempeño para el reconocimiento de voz que otras funciones wavelet madre [8,14]. Asimismo, se emplea un esquema de reducción de ruido por medio de la técnica wavelet denoising de umbral suave [9]. El nivel de umbral empleado en el denoising fue estimado con las expresiones dadas en [9]. En la figura 6 se muestra, en forma global, el esquema de extracción de características empleado. Las salidas de los bloques marcados con las etiquetas deltas y cepstro solamente están presentes en algunas versiones del sistema de extracción, como se describe a continuación.

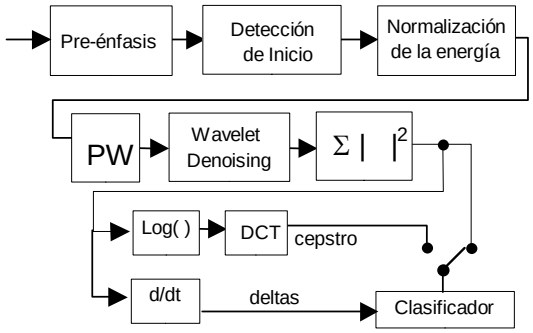

Figura 6 Esquema general de la extracción de características

El primer esquema de extracción consistió en emplear un paquete wavelet cuyo árbol de descomposición se aproxima a los filtros de Mel que se emplean típicamente en los sistemas de reconocimiento basados en transformada de Fourier de corto tiempo (STFT) [15]. En este esquema, por cada uno de los 24 nodos de descomposición se calcula la energía, obteniéndose de esta forma 24 características por cada bloque de 64ms. Se exploraron dos alternativas a este esquema, una de 3 bloques (PW3-db8) y otra de 5 bloques (PW5-db8), para un total de 3x24=72 características para la versión de tres bloques y 5x24=120 características para la versión de cinco bloques. Dado que no existe traslape entre los bloques, este tipo de extracción de características se denomina estático, pues no se le brinda al clasificador información alguna acerca de la dinámica de cambio de las características. Una forma típica en los sistemas basados en STFT para incluir características dinámicas es el empleo de las derivadas temporales de las características [13]. Es así como, otro esquema de extracción evaluado en este trabajo consistió en incluir, además de los parámetros del método PW5-db8, sus derivadas temporales. A este esquema se le denominó PW5-db8+deltas. En este esquema se tiene un total de

120+4x24=216 características, dado que por cada bloque se calculan 24 derivadas, una por cada característica, procedimiento que solamente se puede realizar a partir del segundo bloque.

Otra función wavelet madre que ha mostrado tener buen desempeño en los sistemas de reconocimiento es la Daubechies de orden 4. En este caso, para evaluar su desempeño en el sistema de reconocimiento, se analizaron dos variantes: PW5-db4 y PW5-db4+deltas, esquemas que emplean la misma topología de PW5-db8 y PW5-db8+deltas, respectivamente.

Por otra parte, cuando se emplean paquetes wavelet, es posible calcular el árbol de descomposición que mejor se adapte a la representación de la señal. Esto se hace mediante el empleo del algoritmo de selección de la mejor base [9]. El árbol de descomposición se calculó para una función madre Daubechies 8 y 6 niveles de descomposición. El esquema de extracción que hace uso de este árbol se nombró como PWOpt5.

Finalmente, en [15] proponen incluir el cálculo del cepstro a un sistema de extracción de características basado en paquete wavelet, similar a como se hace para el cálculo de los coeficientes ceptrales de Mel (MFCC). De esta forma, posterior al cálculo de la energía de cada nodo de descomposición, se calcula su logaritmo y finalmente la transformada discreta del coseno (DCT). Esto tiene como fin obtener un conjunto de parámetros altamente no correlacionados [13]. Las versiones de los esquemas de extracción antes descritos que hacen uso del cepstro, se denotan como: PW5 -db8 +cepstro, PW5 -db4 +cepstro, PWOpt5 +cepstro y PW5-db8 +deltas +ceptro.

4. RESULTADOS OBTENIDOS

En la tabla 1 se presentan los porcentajes de acierto para cada uno de los esquemas de extracción de características discutidos en la sección anterior y empleando dos clasificadores, por un lado un perceptrón multicapa, para el cual se indica el número de neuronas de la capa oculta en la tercera columna de la tabla 1. Cada red cuenta con 5 salidas, de tal forma que solamente una salida está activa por cada patrón. Estas redes fueron entrenadas con el algoritmo *backpropagation* con gradiente conjugado escalado. Para la SVM se emplearon dos funciones núcleo (kernel), por un lado kernel de tipo RBF (*Radial Basis Functions*) y por otro lado funciones polinomiales de orden diferente (poly-n), pero con mejores resultados para polinomio de orden 2 y 3 (columna 6 en la tabla 1). Para determinar el punto de silla del funcional de Lagrange de la ec. (2) se utiliza el algoritmo de *sequential minimal optimization* (SMO) [3]. Para ambos clasificadores se especifican en la tabla los porcentajes de acierto empleando el conjunto de validación.

Respecto a los resultados con el perceptrón multicapa, como era de esperarse, al aumentar el conjunto de características, tomando una mayor cantidad de bloques y usando la misma función wavelet madre, se logra aumentar el porcentaje de aciertos. Esto se verifica al comparar los desempeños de los perceptrones con la extracción PW3-db8 y PW5-db8. Este comportamiento no se encuentra en el desempeño del clasificador con SVM, dado que para las versiones de los sistemas de extracción que no incluían el cepstro, los porcentajes de acierto son prácticamente independientes del número de características.

Extracción de Características		RNA		MVS		
Método	**# Carac**	**#neuron. ocultas**	**% de Acierto**	**# VS**	**Kernel**	**% de Acierto**
PW3-db8	72	50	82.76	56	Poly-4	82.76
PW5-db8	120	50	82.14	57	Poly-3	82.14
PW3-db8	72	100	82.76	53	RBF	82.76
PW5-db8	120	100	85.71	56	RBF	82.14
PW5-db8 +deltas	216	50	82.14	60	Poly-3	82.14
PW5-db8 +deltas	216	100	82.14	62	RBF	82.14
PW5-db4 +deltas	216	50	78.57	60	Poly-3	82.14
PW5-db4	120	50	82.14	60	Poly-3	82.14
PW5-db4 +deltas	216	100	82.14	58	RBF	82.14
PW5-db4	120	100	82.14	61	RBF	82.14
PWOpt5	75	50	82.14	56	Poly-4	78.57
PWopt5 -db8	75	100	82.14	62	RBF	78.57
PW5-db8 +cepstro	120	50	96.43	64	Poly-3	92.85
PW5-db8 +cepstro	120	100	92.86	77	RBF	96.43
PW5-db4 +cepstro	120	50	96.43	56	Poly-2	85.71
PW5-db4 +cepstro	120	100	96.43	77	RBF	92.85
PWOpt5 +cepstro	75	50	85.71	65	Poly-2	92.85
PWOpt5 +cepstro	75	100	92.86	78	RBF	92.85
PW5-db8 +deltas +ceptro	216	50	92.86	63	Poly-3	89.28
PW5-db8 +deltas +ceptro	216	100	92.86	77	RBF	96.43

Tabla 1. Resultados obtenidos para cada uno de los esquemas de extracción de características y clasificadores.

Por otra parte, el empleo de un árbol óptimo (PWOpt5-db8) arroja aciertos comparables o menores al árbol de descomposición que se asemeja al banco de filtros de escalas de Mel (PW5-db8). Aunque no se incluye en la tabla, el árbol óptimo ofrece el mismo porcentaje de aciertos que el perceptrón entrenado con las características PW5-db8, sólo cuando se usa un número elevado neuronas en la capa oculta. En cuanto a la SVM, este esquema de extracción de características arrojó el peor desempeño. Lo anterior indica que el árbol de descomposición usado en las características PW5-db8 es lo suficientemente general para ser usado en aplicaciones de reconocimiento de voz. Además, este árbol muestra un mejor desempeño con la función wavelet madre Daubechies 8 que con la Daubechies 4, tanto para el perceptrón como para la SVM.

En cuanto a las versiones el sistema de extracción de características que incluyen coeficientes dinámicos (esquemas delta), se encuentra que éstos no permiten mejorar el porcentaje de aciertos, inclusive tienden a empeorarlos. Esto quizás se deba a la forma como se calculan los paquetes wavelet de cada bloque, pues las condiciones iniciales de los filtros FIR no se inicializan por cada bloque, lo que implicaría que las características incluyan información sobre la interdependencia de los bloques. De esta forma, al considerar las derivadas, se estaría incluyendo información redundante que tiende a confundir al perceptrón y a la SVM.

Cabe anotar que para las versiones del sistema de extracción de características que emplean el cepstro, se logran resultados superiores a los sistemas de los cuales se derivan. De esta forma, se lograron conseguir porcentajes de acierto superiores al 90% para ambos clasificadores.

Finalmente, para los mejores esquemas de reconocimiento, tanto en el perceptrón como en la SVM, se encuentra que el único patrón que presenta fallas es el reconocimiento del dígito cuatro, el cual se identifica como el dígito cinco. Esto indica que las características seleccionadas permiten a ambos clasificadores generalizar de la misma forma.

4. CONCLUSIONES

Es posible emplear un esquema de extracción de características basado en paquetes wavelet con un clasificador por SVM para el reconocimiento de comandos de voz.

Para esta aplicación se encuentra que las SVM presentan una generalización mayor que las ANN para ciertos esquemas de extracción de características y funciones núcleo. Esto se verifica al analizar los resulados de los esquemas de extracción que hacen uso de un árbol wavelet similar al banco de filtros de escalas de Mel, función wavelet Daubechies 8, cálculo de los coeficientes cepstrales (esquemas PW5-db8 +cepstro y PW5-db8 +deltas +cepstro), y emplean funciones de base radial como "kernel". Cuando se emplean funciones polinomiales, con los esquemas de extracción de características antes indicados, la ANN tiene un mejor o igual desempeño. Este comportamiento también se encontró para otros tipos de extracción de características, entre las que se resaltan aquellas que hacen uso de únicamente la energía de los coeficientes wavelet (por ejemplo en PW5-db8, PW5-db4 y PWOpt-db8).

Es posible que con un sistema de extracción diferente, que brinde información global sobre la señal, se logre aumentar el rendimiento de la SVM, ya que su capacidad de inferir sobre los datos no depende de características particulares.

5. BIBLIOGRAFÍA

[1] VAPNIK, V. Statistical Learning Theory. J.Wiley & Sons, Inc., New York, NY, 1998.

[2] VAPNIK, V. The Nature of Statistical Learning Theory, Segunda Edición, 313 páaginas, Springer Verlag Inc, New York, NY, 2001.

[3] CHANG, Ch. And Lin, CH. LIBSVM: A Library for Support Vector Machines. January 3, 2.006.

[4] KECKMAN, V. *Support Vector Machines Basics.* School of Engineering Report 616. The University of Auckland, April, 2.004.

[5] OSUNA, E. Freud, R. and Girosi, F.. *Support Vector Machines: Training and Applications.* Artificial Intelligence Laboratory and Center of Biological and Computational Learning. MIT. A.I. memo No.1602, C.B.C.L paper 144. March, 1.997.

[6] SHAO, Y. and CHANG, C.H. Wavelet Transform to Hybrid Support Vector Machine and Hidden Markov Model for Speech Recognition. ISCAS 2005. Paginas 3833 – 3836, 2005.

[7] TAN, B. FU, M. DERMODY , P. The Use Of Wavelet Transforms In Phoneme Recognition. Proc. ICSLP, 1996.

[8] LONG, C. DATTA, S. Wavelet Based Feature Extraction for Phoneme Recognition. Proc. ICSLP, páginas 264-267, 1996.

[9] RAO, R. BOPARDIKAR, A. Wavelet Transforms. Addison-Wesley, 1998.

[10] CHIN K. *Support Vector Machines Applied to Speech Pattern Classification.* Degree of Master of Philosophy. Cambridge University Engineering Deparment. Cambridge, 1.999.

[11] CLARKSON, P. MORENO, P. *On the Use of Support Vector Machines for Phonetic Classification.* Cambridge Research Laboratory. Cambridge, 1.999.

[12] GANAPATHIRAJU, A. HAMAKER, J. PICOME, J. *Hibrid SVM/HMM Architectures for Speech Recognition.* Institute for signal and Information Processing. Department for Electrical and Computer Engineering Mississippi State University, 2.000.

[13] BECCETTI, C. PRINA, L. Speech Recognition. Wiley, 1999.

[14] MARÍN, J. Sistema de reconocimiento de fonemas en tiempo real usando transformada wavelet y DSP. Revista de Investigaciones Universidad del Quindío, **12**, páginas 42-46, 2003.

[15] FAROOQ, O. DATTA, S. Mel Filter-Like Admissible Wavelet Packet Structure for Speech Recognition. IEEE Signal Processing Letters, **8**, páginas 196-198, 2001.

[16] JIAN, H., JOO, M. GAO, Y. Feature extraction using wavelet packets strategy. IEEE Proc. Of 42nd Conf. On Decision and Control, páginas 4517-4520, 2003.

[17] KARAM, J. PHILLIPS, W. New wavelet packet model for automatic speech recognition system. IEEE Proc. Can. Conf. on Elect. and Comp. Eng.. Páginas 511-514, 2001