# SELECCIÓN Y PESADO DE PARÁMETROS ACÚSTICOS MEDIANTE ALGORITMOS GENÉTICOS PARA EL RECONOCIMIENTO DEL LOCUTOR

*Maider Zamalloa[1,2], Germán Bordel[1], Luis Javier Rodriguez[1], Mikel Peñagarikano[1], Juan Pedro Uribe[2]*

(1) GTTS, Departamento de Electricidad y Electrónica, Universidad del País Vasco
(2) Ikerlan – Centro de Investigaciones Tecnológicas
mzamalloa001@ikasle.ehu.es

**RESUMEN**

Los coeficientes cepstrales constituyen la representación acústica más empleada en tareas de reconocimiento del locutor. Generalmente, la dimensión de esta representación se amplía introduciendo características dinámicas. Algunas de estas características pueden depender de otras o ser redundantes. Este artículo estudia la selección del conjunto óptimo de parámetros acústicos para reconocimiento del locutor, mediante una búsqueda heurística basada en algoritmos genéticos. Para mejorar el rendimiento del sistema, tras la selección se ha añadido un pesado. La búsqueda del conjunto óptimo de pesos se ha efectuado también mediante algoritmos genéticos. De este estudio se desprende que el conjunto óptimo de parámetros es el formado por los 12 cepstrales y la energía. Así pues, la información dinámica no parece tan relevante como en reconocimiento del habla. Además, se ha obtenido una reducción relativa del error de reconocimiento del 24.31%. Además de validar la metodología propuesta, estos resultados refuerzan la opción de utilizar representaciones acústicas reducidas para aligerar el coste computacional y aumentar la robustez de los sistemas de reconocimiento.[1]

## 1. INTRODUCCIÓN

El empleo de sistemas biométricos que analizan la voz para verificar o determinar identidades es actualmente la forma más natural y menos intrusiva de solucionar los problemas relacionados con el control de acceso a recursos críticos o privados (edificios, cuentas de Internet, etc.). Las aplicaciones de alta seguridad requieren que la eficacia de estos sistemas sea alta, algo especialmente difícil cuando el sistema interactúa con cientos de usuarios legítimos y miles de potenciales impostores. Sin embargo, es algo que se puede abordar adecuadamente por medio de grandes cantidades de datos de habla y redes de servidores computacionalmente potentes, que permiten la estimación de modelos robustos.

No obstante, también pueden beneficiarse de esta tecnología ciertas aplicaciones cuyo principal requerimiento no es la seguridad, sino la adaptación y personalización automática de los servicios que ofrecen. En este tipo de aplicaciones, la transparencia y la naturalidad son críticas, siendo preciso no interferir con el funcionamiento del servicio prestado. En un escenario cada vez más previsible para este tipo de aplicaciones, el cliente puede haber accedido al servicio por medio de un dispositivo portátil o embebido, con limitaciones de almacenamiento y capacidad computacional, lo que plantea la necesidad de aligerar el coste computacional de los procesos de reconocimiento con el menor impacto posible en su eficiencia.

Una vía para abordar esta reducción de costes computacionales pasa por la elección de parámetros acústicos, que es clave para el reconocimiento automático de locutores. Sería deseable contar únicamente con parámetros que sean relevantes para la tarea de clasificación, ignorando todo dato redundante o que aporte información innecesaria (el vector de parámetros acústicos es portador de información diversa, como el idioma, el locutor, el mensaje emitido, la emoción etc.). Una reducción del conjunto de parámetros permitirá reducir el coste computacional y, en función de las características de los datos a utilizar para la estimación de modelos, incluso aumentar la robustez del sistema. Yendo un poco más allá, aún limitándonos estrictamente al uso de los parámetros significativos, cabría hablar de mejores y peores parámetros en función su aportación relativa a la distinción entre locutores, cuestión que puede tenerse en cuenta para mejorar el rendimiento del sistema dado un determinado conjunto de parámetros.

Un parámetro acústico óptimo debe contar con las siguientes características: (1) alta variación inter-locutor, (2) baja variación intra-locutor, (3) facilidad de medición, (4) robustez frente a ataques por simulación, (5) robustez frente al ruido y (6) independencia frente a otros parámetros. Desafortunadamente, no existe ningún parámetro que cumpla con todos estos requerimientos**.** Los parámetros de alto nivel que caracterizan a un locutor, tales como los patrones de pronunciación, el uso del lenguaje, etc., son robustos frente al ruido pero requieren el empleo de reconocimiento del habla para obtener la secuencia de palabras y una gran cantidad de datos para estimar los modelos acústicos y de lenguaje. Esta complejidad es hoy por hoy inadmisible en cualquier sistema de reconocimiento del locutor. Por lo tanto, son los parámetros acústicos de bajo nivel los más empleados, ya que su extracción es fácil, no requieren el reconocimiento del habla y basta con una relativamente pequeña cantidad de datos para estimar buenos modelos. Su principal inconveniente es que pueden ser fácilmente corrompidos por el ruido ambiental y otras fuentes de distorsión.

Los sistemas actuales emplean los mismos parámetros (*Mel-Frequency Cepstral Coefficients*, *MFCC*) tanto para reconocimiento del habla como para reconocimiento del locutor, porque además de recoger la distribución frecuencial para la identificación de sonidos, transmiten información sobre el pulso glotal y la forma y longitud del tracto vocal,

[1] Este trabajo ha sido parcialmente financiado por el Gobierno Vasco, dentro del programa SAIOTEK, a traves del proyecto S-PE04UN18.

que son características específicas de cada hablante. Asimismo, diversos trabajos han demostrado que la información dinámica mejora de forma significativa la eficiencia de los sistemas de reconocimiento del habla, por lo que además de los MFCC y la energía, generalmente se utilizan también sus primeras y segundas derivadas. En función del módulo de procesamiento acústico, el vector de parámetros puede estar compuesto por un número de componentes que típicamente varía entre 20 y 50.

La determinación de un subconjunto de parámetros óptimo partiendo de un vector acústico *D*-dimensional, sin poner límites a toda la combinatoria posible, es computacionalmente prohibitiva incluso para valores moderados de *D,* sobre todo si tenemos en cuenta que hemos de usar la tasa de clasificación como función de evaluación. Una metodología más simple consistiría en evaluar de forma individual los *D* parámetros y seleccionar los *K* más discriminantes, sin tener en cuenta las dependencias entre ellos. De hecho, se han propuesto varias técnicas subóptimas en la literatura que sacrifican la optimización del conjunto seleccionado por la eficiencia computacional [1].

Una estrategia bastante razonable para realizar esta búsqueda sin establecer límites a priori a toda la combinatoria posible es la utilización de *algoritmos genéticos*. Los algoritmos genéticos, introducidos por Holland en 1975 [2], son técnicas de búsqueda heurísticas y aleatorias basadas en las estrategias de la evolución biológica, con tres mecanismos básicos: selección del más fuerte, mezcla y mutación. Las soluciones candidatas se representan por medio de individuos o *cromosomas*. La población inicial de individuos suele generarse de forma aleatoria. Después, el algoritmo genético, de forma iterativa, conduce a la población hacia un punto óptimo, de acuerdo a una métrica compleja (función de evaluación o de adecuación) que mide la eficiencia de los individuos para llevar a cabo una tarea concreta. El algoritmo selecciona los individuos más fuertes (los que reciben mejor evaluación) y a continuación mezcla, muta o mantiene sus representaciones. Una de las principales ventajas de los algoritmos genéticos sobre otras técnicas de búsqueda heurísticas consiste en que no realizan ninguna hipótesis sobre las propiedades de la función de evaluación pudiendo incluso definir y emplear funciones de evaluación multi-objetivo de forma completamente natural [3][4].

Hemos podido localizar varios trabajos que aplican estas técnicas para la extracción y selección de parámetros en tareas de reconocimiento del locutor [5][6][7]. Además de seleccionar los parámetros más adecuados, las mismas técnicas se pueden aplicar al pesado de parámetros, sin más que codificar, en vez de una decisión booleana, el peso que se asigna a cada parámetro. El pesado establece una modulación de los parámetros acústicos, aumentando la influencia de los más relevantes. De esta forma es posible mejorar la eficacia del reconocedor en sistemas de verificación e identificación del locutor [5][8].

En este trabajo se plantea la búsqueda de los parámetros acústicos más relevantes para el reconocimiento del locutor. Se trata de obtener conjuntos reducidos de parámetros que permitan ahorrar esfuerzo computacional con una baja degradación o incluso con una mejora del rendimiento del sistema. Será necesario analizar un gran número de subconjuntos de parámetros de diferentes dimensiones. Se parte de un vector acústico de 38 dimensiones, compuesto por los 12 MFCC, sus primeras y segundas derivadas, la energía y su derivada. Tras una primera fase de selección, en la que se determina el conjunto óptimo de *K* parámetros, se aborda una segunda fase de pesado, en la que se establece el conjunto óptimo de pesos para un conjunto dado de *K* parámetros.

El artículo se estructura como sigue. En el apartado 2 se describe el sistema de reconocimiento del locutor y los métodos de selección y pesado mediante algoritmos genéticos. En el apartado 3 se describen la base de datos de habla utilizada en los experimentos y la configuración del algoritmo genético empleado en las búsquedas. En el apartado 4 se muestran y comentan los resultados obtenidos. Por último, el apartado 5 contiene las conclusiones del trabajo y una breve descripción de las tareas pendientes.

## 2. METODOLOGÍA

### 2.1. El sistema de reconocimiento del locutor

#### *2.1.1. Procesamiento acústico*

El habla, adquirida a 16 KHz, se analiza en tramos de 25 ms, a intervalos de 10 ms. A continuación, se aplica una ventana de análisis tipo Hamming y se calcula una FFT de 512 puntos y se calcula la media de las amplitudes de la FFT en 24 filtros triangulares solapados, definidos de acuerdo a la escala Mel de bandas críticas. Por último, se calcula el logaritmo de los coeficientes del banco de filtros y se aplica una transformada discreta de cosenos (DCT), que produce 12 coeficientes cepstrales. Para aumentar la robustez frente a la distorsión del canal se resta la media en el tiempo de los cepstrales (calculada independientemente para cada señal) [9]. También se calculan las primeras y segundas derivadas de los coeficientes cepstrales, la energía (E) y su derivada, conformando un vector acústico de 38 componentes.

#### *2.1.2. Cuantificación vectorial*

Se aplica el algoritmo de cuantificación vectorial LBG [10], que produce como resultado un conjunto de centroides (*codebook*), *C*, que minimiza la distorsión media (distancia euclídea) en la cuantificación de los vectores acústicos del corpus de entrenamiento (que incluye a todos los locutores). A continuación, cada vector acústico de la base de datos se reemplaza por el índice del centroide más cercano. Durante el reconocimiento se aplica el mismo procedimiento. Cada señal de entrada es procesada para obtener una secuencia de vectores acústicos $X=\{X(1),X(2),...,X(T)\}$ y a cada $X(i)$ se le asigna el índice del centroide más cercano del codebook, $Y(i)$, obteniendo una secuencia equivalente de etiquetas acústicas $Y=\{Y(1),Y(2),...,Y(T)\}$.

#### *2.1.3. Modelos de locutor*

La aproximación clásica al reconocimiento de locutor mediante cuantificación vectorial consiste en estimar un codebook específico para cada locutor, y después a cada señal de entrada asignarle aquel locutor cuyo codebook haya producido la mínima distorsión al cuantificar los vectores acústicos. En este trabajo los modelos de locutor no son codebooks, sino distribuciones de etiquetas. El método consiste en calcular un único codebook compartido por todos los locutores, y almacenar como parámetros específicos de cada locutor las frecuencias de las etiquetas. Estos modelos tan sencillos han dado buenos resultados en adaptación al locutor mediante agrupamiento de locutores en tareas de reconocimiento del habla [11].

Supongamos que $U(i)$ es el conjunto de entrenamiento correspondiente al locutor $i$, $c(i)$ el número de etiquetas que

aparecen en *U(i)*, y *c(k,i)* el número de veces que aparece la etiqueta *k* en *U(i)*. La probabilidad de la etiqueta *k* dado el locutor *i*, *P(k|i),* se puede calcular empíricamente de la siguiente manera:

$$P(k \mid i) = \frac{c(k,i)}{c(i)} \tag{1}$$

Finalmente, suponiendo independencia estadística entre etiquetas sucesivas, la probabilidad de la secuencia de etiquetas *Y={Y(1),Y(2),...,Y(T)}*, dado el locutor *i*, puede calcularse como sigue:

$$P(Y \mid i) = \prod_{t=1}^{T} P(Y(t) \mid i) \tag{2}$$

*2.1.4. Reconocimiento del locutor*

Suponiendo que cada señal de entrada ha sido pronunciada por alguno de los *S* locutores conocidos, dada la secuencia de etiquetas *Y={Y(1),Y(2),...,Y(T)}*, el locutor más probable viene dado por la siguiente expresión:

$$\hat{\imath}(Y) = \arg\max_{i=1,\ldots,S} \{P(i \mid Y)\} \tag{3}$$

Aplicando la regla de Bayes, queda:

$$\hat{\imath}(Y) = \arg\max_{i=1,\ldots,S} \left\{ \frac{P(Y \mid i)P(i)}{P(Y)} \right\} = \arg\max_{i=1,\ldots,S} \{P(Y \mid i)P(i)\} \tag{4}$$

ya que la maximización sobre el conjunto de locutores no depende de la secuencia acústica. Por tanto, dado que además suponemos que todos los locutores tienen la misma probabilidad a priori:

$$\hat{\imath}(Y) = \arg\max_{i=1,\ldots,S} \{P(Y \mid i)\}, \tag{5}$$

e introduciendo (2) en (5):

$$\begin{aligned} \hat{\imath}(Y) &= \arg\max_{i=1,\ldots,S} \left\{ \prod_{t=1}^{T} P(Y(t) \mid i) \right\} \\ &= \arg\max_{i=1,\ldots,S} \left\{ \log \prod_{t=1}^{T} P(Y(t) \mid i) \right\} \\ &= \arg\max_{i=1,\ldots,S} \left\{ \sum_{t=1}^{T} \log P(Y(t) \mid i) \right\} \end{aligned} \tag{6}$$

De la expresión (6) deducimos que el coste computacional del reconocimiento es lineal con el número de locutores *(S)* y con la longitud de la frase de entrada *(T)*. En total se requieren *SxT* accesos a memoria, *Sx(T-1)* sumas y *S-1* comparaciones. Por conveniencia, se supone que las probabilidades de las etiquetas se almacenan en forma logarítmica.

*2.1.5. Selección y pesado de los parámetros acústicos*

El proceso de selección consiste en establecer cuáles son las *K* componentes del vector acústico más relevantes desde el punto de vista de la clasificación. Entre las númerosas tecnicas de búsqueda y criterios de optimización que se han propuesto a lo largo de los años (vease [12]), por razones que han quedado expuestas anteriormente, se ha optado por una búsqueda heurística basada en algoritmos genéticos.

El proceso comienza normalizando la base de datos para que las *D* componentes del espacio vectorial tengan media 0 y varianza 1. Cada subconjunto de *K* parámetros propuesto como candidato, $\Gamma^{(K)}=\{j_1, j_2,..., j_K\}$ ha de ser evaluado, para finalmente quedarse con aquel subconjunto que reciba una mejor evaluación. Los pasos a seguir para hacerlo son: (1) cada vector de la base de datos es sustituido por el vector reducido, formado por las componentes enumeradas en $\Gamma^{(K)}$; (2) se genera un codebook reducido *C'*, utilizando las componentes $\Gamma^{(K)}$ del codebook completo *C*; (3) se etiquetan los corpus de entrenamiento y validación reducidos mediante *C'*; (4) utilizando las etiquetas del corpus de entrenamiento se estiman los modelos de locutor; (5) se aplican los modelos de locutor para clasificar las señales del corpus de validación; y (6) se asigna al conjunto $\Gamma^{(K)}$ la tasa de reconocimiento obtenida.

Una vez se ha establecido el subconjunto óptimo de *K* componentes, $\hat{\Gamma}^{(K)} = \{\hat{j}_1, \hat{j}_2, ..., \hat{j}_K\}$ , se calcula un nuevo codebook, $C(\hat{\Gamma}^{(K)})$, se etiquetan con él los vectores reducidos de los corpus de entrenamiento y test, y se calculan los modelos de locutor. Por último, el subconjunto óptimo $\hat{\Gamma}^{(K)}$ se evalúa clasificando las señales del corpus de test. Los tres conjuntos de señales utilizados en este trabajo, entrenamiento, validación y test, son independientes, ya que contienen señales distintas del mismo conjunto de locutores.

Eliminar parámetros acústicos poco relevantes reduce, sin duda, los requerimientos computacionales del reconocedor, pero tal como apuntan los resultados obtenidos en un trabajo previo [8], aún es posible mejorar el rendimiento del sistema buscando y aplicando un conjunto adecuado de pesos *W=(w1,w2,...,wK),* que transforma cada vector acústico *X=(x1,...,xK)* en un vector ponderado *X'=(w1x1,w2x2,...,wKxK)*.

Es necesario, por tanto, efectuar un proceso de búsqueda en el espacio *K*-dimensional de los pesos. Cada conjunto de pesos candidato *W* se aplica a los vectores reducidos de la base de datos para obtener los vectores ponderados. Después se calcula un codebook específico *C(W),* utilizando los vectores ponderados del corpus de entrenamiento. Cada vector ponderado es sustituido por el índice del centroide más cercano en *C(W)*. Por último, utilizando las etiquetas del corpus de entrenamiento se estiman los modelos de locutor, y se asigna a *W* la tasa de reconocimiento de locutor obtenida sobre el corpus de validación utilizando dichos modelos. Una vez obtenido el conjunto de pesos óptimo, $\hat{W}$, como medida de evaluación final se calcula la tasa de reconocimiento sobre el corpus de test reducido y ponderado.

**2.2. Búsqueda basada en algoritmos genéticos**

La selección y el pesado de los parámetros acústicos se realizan por medio de un conocido algoritmo genético (*Simple Genetic Algorithm*, *SGA*) [13]. Tanto el codebook de cuantificación vectorial como los modelos de locutor se estiman a partir de los datos del corpus de entrenamiento. Sin embargo, la evaluación que tiene lugar dentro del proceso de búsqueda se efectúa a partir de los datos del conjunto de validación, lo que asegura una mayor robustez e independencia de los resultados. En este trabajo se utiliza como función de evaluación la tasa de reconocimiento de locutor sobre el corpus de validación.

El algoritmo SGA comienza evaluando todos los individuos de una población de partida generada

aleatoriamente. Por individuo entendemos o bien un subconjunto de $K$ componentes en el caso de la selección, o bien un conjunto de $K$ pesos en el caso del pesado. Una vez evaluados todos los candidatos, se seleccionan algunos de ellos, normalmente los más fuertes o adecuados, es decir, aquellos que ocupan los primeros puestos de acuerdo a la función de evaluación, para mezclarlos y mutarlos con objeto de obtener la población para la siguiente generación. La convergencia del algoritmo depende de varios elementos: la forma de representar la información, el tamaño de la población, la forma de obtener la población inicial, el criterio de selección de individuos, el tipo de mezcla y la tasa de mutación. En concreto, hay tres métodos de selección muy comunes: proporcional, por rango y por torneo. Los dos primeros favorecen mucho a los individuos más fuertes, mientras que la selección por torneo da más probabilidad a los individuos menos adaptados. La mezcla siempre se realiza entre dos individuos previamente seleccionados, pero puede realizarse de muchas formas. El método más común consiste en definir $n$ puntos de corte, que determinan $n+1$ segmentos en cada individuo, e intercambiar los segmentos pares para producir los dos individuos hijos. Una mutación consiste en invertir el valor de ciertos bits tomados al azar en la representación de un individuo. Se introduce para evitar en lo posible que la búsqueda quede atrapada en extremos locales. La llamada *tasa de mutación* establece la probabilidad de que tras una mezcla tengan lugar mutaciones.

A veces se permite que ciertos individuos (generalmente los más fuertes) se transmitan sin mezclar ni mutar a la siguiente generación, lo que se conoce como *elitismo*. En este trabajo se ha aplicado el caso más simple de elitismo, que consiste en transmitir únicamente al individuo más fuerte. Esto garantiza que la evaluación del mejor individuo mejore monótonamente en sucesivas generaciones. Si la mejora es inferior a un cierto umbral, o si se ha alcanzado un número máximo de generaciones, el algoritmo finaliza y devuelve como solución el mejor individuo.

*2.2.1. Representación de la información en la selección de parámetros*

Para representar la selección de $K$ componentes se utiliza un vector de $D$ enteros en el rango [0, 255], $R=\{r_1, r_2, ..., r_D\}$. Los K valores más altos indican implícitamente qué componentes se están eligiendo: $\Gamma^{(K)} = \{\{j_1, j_2, ..., j_K\} \mid r_{j1} \geq r_{j2} \geq ... \geq r_{jK} \geq r_j, \forall\, j \notin \{j_1, j_2, ..., j_K\}\}$. Obviamente, un cierto conjunto de componente $\Gamma^{(K)}$ puede ser representado por muchos vectores distintos, o lo que es igual, dado un cierto individuo $R$, pequeños cambios en su especificación no modifican la elección de componentes. Esta redundancia en la representación facilita que la evolución del algoritmo sea suave y converja al subconjunto óptimo $\hat{\Gamma}^{(K)}$ idenpendientemente de la población inicial.

## 3. CONFIGURACIÓN DE LOS EXPERIMENTOS

### 3.1. La base de datos

Para los experimentos se ha utilizado *Albayzin* [14], base de datos fonéticamente equilibrada en castellano. Grabada a 16 kHz en condiciones de laboratorio, *Albayzin* fue diseñada para el entrenamiento de modelos acústicos en el ámbito del reconocimiento del habla. La base de datos está compuesta por 204 locutores, cada uno de ellos con un mínimo de 25 señales. La longitud media de cada señal es de 3.55 segundos.

En los experimentos se utilizan casi todas las señales de la base de datos. De cada locutor se toman 25 señales, 10 de ellas para el corpus de entrenamiento, 7 para el de validación y 8 para el de test. Así pues, el corpus de entrenamiento contiene 2040 señales, el de validación 1428 y el de test 1632.

### 3.2. Configuración del algoritmo genético

El algoritmo genético se ha implementado por medio de ECJ *"A Java-based Evolucionary Computation and Genetic Programming Research System"*, una herramienta para el desarrollo de aplicaciones de computación evolutiva y programación genética, creada y mantenida en la universidad George Mason de Estados Unidos, que se ofrece bajo una licencia especial de código abierto [15]. ECJ tiene una arquitectura muy flexible, permite definir representaciones arbitrarias, genomas de longitud fija y variable, métodos de optimización multiobjetivo y diversos operadores de selección.

Con objeto de ajustar los parámetros que controlan el rendimiento y la convergencia del algoritmo genético, se han llevado a cabo experimentos preliminares. El tamaño de la población es uno de los principales parámetros. Poblaciones demasiados extensas retrasan excesivamente la convergencia del algoritmo, mientras que poblaciones demasiado pequeñas limitan el rendimiento de la búsqueda. Se ha comprobado que el tamaño de la población depende del espacio de búsqueda. En los experimentos de selección todos los individuos contienen 38 genes, uno por cada parámetro del vector acústico. Así pues, se ha llegado a un tamaño de población óptimo para todos ellos. Sin embargo, en los experimentos de pesado el número de genes es proporcional al número de parámetros a pesar. Por tanto, los experimentos de pesado requieren poblaciones cada vez más pequeñas a medida que se reduce el número de parámetros.

Se ha comprobado que 40 generaciones son suficientes para alcanzar la convergencia, por lo que no se ha aplicado ningún otro criterio de parada. Para reducir el tamaño de la representación y, por tanto, el coste computacional, se han utilizado tan sólo 8 bits por gen, es decir, enteros en el rango [0, 255]. A la hora de elegir qué individuos de la actual generación van a cruzarse para producir los individuos de la siguiente, se ha optado por la selección por torneo, con una pre-selección aleatoria de 7 o 2 individuos (de nuevo, este número depende del tamaño de la población), entre los que se escoge el más fuerte. Este es el método recomendado por los desarrolladores de la herramienta, ya que permite mantener un cierto grado de diversidad y, al mismo tiempo, dirigir el algoritmo hacia poblaciones cada vez mejor adaptadas. Se ha aplicado una mezcla con un punto de corte, es decir, cada uno de los dos individuos seleccionados se rompe en dos segmentos que se intercambian. Las tasas de mezcla y mutación se han fijado heurísticamente a aquellos valores que han producido mejores resultados (1.0 y 0.01, respectivamente). Por último, se ha aplicado el caso básico de elitismo, manteniendo el mejor individuo de cada generación.

## 4. RESULTADOS DE LOS EXPERIMENTOS

### 4.1. Selección de parámetros

Con objeto de identificar el subconjunto óptimo de parámetros tanto en cuanto a coste computacional como en cuanto a eficacia del sistema de reconocimiento, se ha lanzado

una amplia batería de experimentos con subconjuntos de diferentes dimensiones. Puesto que el coste computacional es lineal con el número de parámetros, los experimentos tratan de encontrar el mejor compromiso posible entre la tasa de reconocimiento y el número de parámetros. Se han llevado a cabo experimentos con *K = 13, 12, 10, 8, 6, 4* y *2*. En la Tabla 1 se muestran los conjuntos óptimos, así como las tasas de error correspondientes, obtenidas sobre el corpus de test.

**Tabla 1.** S*ubconjuntos óptimos de parámetros obtenidos tras la selección mediante algoritmos genéticos, y tasas de error en experimentos de reconocimiento sobre el corpus de test de Albayzin, para varios valores de K. El identificador cXX hace referencia al cepstral XX, y E a la energía.*

| *K* | *%Error* | *Parámetros Seleccionados* |
|---|---|---|
| 13 | 5.45 | c01, c02, c03, c04, c05, c06, c07, c08,c09, c10, c11, c12, E |
| 12 | **5.39** | c01, c02, c03, c04, c06, c07, c08, c09, c10, c11, c12, E |
| 10 | 6.25 | c01, c02, c04, c05, c06, c08, c09, c10, c11, c12 |
| 8 | 9.13 | c01, c02, c04, c07, c08, c09, c10, c12 |
| 6 | 16.23 | c01, c02, c03, c06, c08, c12 |
| 4 | 30.94 | c01, c02, c03, c10 |
| 2 | 73.16 | c01, c04 |

En todos los experimentos de selección desde *K=12* hasta *K=2,* la búsqueda, que se efectúa sobre un espacio de 38 dimensiones, ha dado como resultado un subconjunto óptimo formado por algunos cepstrales y la energía. El hecho de que no haya resultado seleccionada ninguna de las características dinámicas parece indicar que éstas no tienen la misma relevancia en reconocimiento del locutor que en reconocimiento del habla. El experimento de selección con *K=13* se ha llevado a cabo precisamente para verificar esta intuición. Atendiendo a las tasas de error, se concluye que el mejor subconjunto es el compuesto por 12 parámetros (11 cepstrales más la energía). No obstante, los subconjuntos de 10 y 13 parámetros ofrecen un rendimiento muy similar.

Puesto que la energía no aparece en los subconjuntos óptimos de tamaño inferior a 12, se ha considerado oportuno evaluar el rendimiento potencial del conjunto formado por los 12 cepstrales. La tasa de error obtenida sobre el conjunto de test utilizando 12 cepstrales es del *5.94%*, sólo ligeramente peor que la de los subconjuntos óptimos de tamaño 12 y 13 obtenidos mediante algoritmos genéticos.

### 4.2. Pesado de parámetros

En un trabajo previo se presentaron resultados interesantes de pesado con el conjunto completo de 38 componentes [8]. Sin embargo, el objetivo principal de este trabajo es identificar subconjuntos de parámetros que permitan reducir los costes computacionales de los sistemas de reconocimiento de locutor. Pero reducir el número de componentes puede repercutir negativamente en la tasa de reconocimiento, tal como se observa en la Tabla 1 al ir reduciendo el número de componentes de 10 a 2. El pesado se introduce, precisamente, con objeto de mejorar el rendimiento del sistema cuando se utilizan conjuntos reducidos de parámetros acústicos.

Así pues, se ha realizado una búsqueda mediante algoritmos genéticos de los pesos óptimos para los vectores reducidos obtenidos en la fase de selección, para *K=13, 10, 8, 6, 4* y *2*. Además, como referencia, se han obtenido también los pesos óptimos para el vector original de 38 componentes y para el vector formado por los 12 cepstrales. En la Tabla 2 se muestra el error de reconocimiento obtenido sobre el corpus de test utilizando parámetros sin pesar y parámetros pesados con los pesos óptimos.

**Tabla 2.** *Resultados de reconocimiento de locutor sobre el corpus de test de Albayzin utilizando parámetros sin pesar y parámetros pesados con los pesos óptimos obtenidos mediante algoritmos genéticos, para el conjunto original de 38 componentes, el conjunto de los 12 cepstrales y los subconjuntos óptimos obtenidos en la fase de selección.*

| | | *% ERROR* | |
|---|---|---|---|
| *K* | | *Sin pesos* | *Pesos óptimos* |
| 38 (vector completo) | | 19.42 | 5.51 |
| 12 cepstrales | | 5.94 | 4.53 |
| *Subconjuntos óptimos obtenidos mediante algoritmos genéticos* | 13 | 5.45 | **4.17** |
| | 10 | 6.25 | 5.15 |
| | 8 | 9.13 | 7.23 |
| | 6 | 16.23 | 14.83 |
| | 4 | 30.94 | 28.86 |
| | 2 | 73.16 | 74.08 |

Los parámetros pesados proporcionan casi en todos los casos tasas de error más pequeñas. La única excepción se produce para *K=2*, debido a la no convergencia del algoritmo genético. El uso de los 38 parámetros originales pesados conlleva una reducción relativa del error del 71.62%. Por otra parte, la búsqueda de los pesos óptimos para *K=38* presenta una convergencia extraordinariamente rápida. Esto es debido a la normalización de la base de datos, que unifica las varianzas de todos los parámetros. En particular, hace que las derivadas adquieran más relevancia en el proceso de clasificación e inserten ruido. Por tanto, resulta muy beneficioso asignarles un peso pequeño, es decir, devolverles la pequeña varianza que tenían. Este hecho se ve reflejado en los pesos óptimos obtenidos para los 38 parámetros originales, entre los que claramente prevalecen los de los cepstrales y la energía (véase la Tabla 3). Hay que tener en cuenta que los pesos determinan la relevancia de cada parámetro en el proceso de reconocimiento.

**Tabla 3.** *Pesos óptimos obtenidos para el conjunto original de 38 parámetros acústicos. El identificador cXX se refiere al cepstral XX, dXX y ddXX a su primera y segunda derivada, E a la energía y dE a su derivada.*

| Parámetro | Peso | Parámetro | Peso | Parámetro | Peso | Parámetro | Peso |
|---|---|---|---|---|---|---|---|
| c01 | 225 | d01 | 58 | dd01 | 14 | E | 220 |
| c02 | 152 | d02 | 93 | dd02 | 61 | dE | 25 |
| c03 | 228 | d03 | 62 | dd03 | 147 | | |
| c04 | 223 | d04 | 118 | dd04 | 161 | | |
| c05 | 238 | d05 | 33 | dd05 | 104 | | |
| c06 | 218 | d06 | 11 | dd06 | 114 | | |
| c07 | 189 | d07 | 65 | dd07 | 61 | | |
| c08 | 209 | d08 | 39 | dd08 | 14 | | |
| c09 | 207 | d09 | 45 | dd09 | 43 | | |
| c10 | 204 | d10 | 36 | dd10 | 60 | | |
| c11 | 184 | d11 | 124 | dd11 | 112 | | |
| c12 | 237 | d12 | 73 | dd12 | 84 | | |

Atendiendo a los resultados mostrados en la Tabla 2, se puede concluir que el subconjunto óptimo de parámetros acústicos pesados es el seleccionado para *K=13*, compuesto por los 12 cepstrales y la energía. Este conjunto proporciona un error del 4.17%, 1.34 puntos inferior al obtenido con los 38 parámetros originales, lo que supone una reducción relativa del error del 24.31%. Los pesos óptimos para el conjunto formado por los 12 cepstrales y la energía se muestran en la Tabla 4. Finalmente, entre los conjuntos de parámetros pesados, sólo tres superan al formado por los 38 parámetros originales: los subconjuntos seleccionados para *K=10* y *K=13*, y el conjunto formado por los 12 cepstrales.

**Tabla 4.** *Pesos óptimos obtenidos para el conjunto de los 12 cepstrales y la energía.*

| c01 | c02 | c03 | c04 | c05 | c06 | c07 | c08 | c09 | c10 | c11 | c12 | E |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 168 | 174 | 136 | 197 | 189 | 149 | 148 | 149 | 160 | 137 | 166 | 138 | 240 |

## 5. CONCLUSIONES

Los coeficientes cepstrales y sus derivadas son los parámetros más empleados en los sistemas actuales de reconocimiento de locutor. El mismo conjunto de parámetros se emplea también en otras tareas que requieren el procesamiento de señales de voz, como el reconocimiento del habla o la identificación de la lengua. Sin embargo, algunos de estos parámetros pueden ser redundantes, dependientes entre sí o incluso perjudiciales para la clasificación. Desde ese presupuesto, este trabajo trata de determinar el subconjunto óptimo de parámetros acústicos para el reconocimiento de locutor, tanto en cuanto al rendimiento del sistema como en cuanto al coste computacional, que en este caso es lineal con el número de parámetros.

El estudio se basa en la selección y posterior pesado de diferentes conjuntos de parámetros, mediante técnicas heurísticas de búsqueda, en concreto, mediante algoritmos genéticos. El algoritmo de selección trata de identificar los parámetros más relevantes, mientras que el pesado escala los parámetros seleccionados y mejora la robustez del sistema de reconocimiento.

Los resultados obtenidos en los experimentos realizados sobre la base de datos de habla leída *Albayzin*, con 204 locutores, determinan que el subconjunto óptimo pesado para el reconocimiento del locutor es el compuesto por los 12 cepstrales y la energía. Cabe destacar que este subconjunto, que excluye la información dinámica, es más eficaz que el conjunto pesado compuesto por los 38 parámetros originales, con una reducción relativa del error del 24.31%. Este resultado pone en evidencia la importancia de eliminar la información no relevante del vector acústico, puesto que al excluir los parámetros dinámicos no sólo disminuye el coste computacional sino que aumenta el rendimiento del sistema de reconocimiento. En definitiva, se puede concluir que los parámetros dinámicos no aportan información relevante para la clasificación de locutores, sino que más bien entorpecen dicha tarea.

Puesto que el pesado no es más que una transformación lineal con matriz diagonal, una vez comprobado el efecto positivo que tiene esta medida, se plantea como trabajo futuro la búsqueda de matrices de transformación que optimicen el rendimiento del sistema de reconocimiento. En un sentido más práctico, actualmente se está implementando una Aplicación Orientada a Servicio de reconocimiento de locutor para dispositivos con recursos computacionales limitados (teléfonos móviles y PDAs).

## 6. BIBLIOGRAFÍA

[1] A.K. Jain, R.P.W. Duin, J. Mao. *"Statistical Pattern Recognition: A Review"*, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 22, No. 1, pp. 4—37, Enero 2000.

[2] J.H. Holland. *"Adaptation in natural and artificial systems"*. University of Michigan Press, 1975 (reeditado en 1992 por MIT Press, Cambridge, MA).

[3] J. Yang, V. Honavar. *"Feature subset selection using a genetic algorithm"*, IEEE Intelligent Systems, Vol. 13, No. 2, pp. 44—49, Marzo 1998.

[4] L.S. Oliveira, R. Sabourin, F. Bortolozzi, C.Y. Suen. *"A Methodology for Feature Selection Using Multiobjective Genetic Algorithms for Handwritten Digit String Recognition"*, International Journal of Pattern Recognition and Artificial Intelligence, Vol. 17, No. 6, pp. 903—929, 2003.

[5] D. Charlet, D. Jouvet. *"Optimizing feature set for speaker verification"*, Pattern Recognition Letters, Vol. 18, No. 9, pp. 873—879, Septiembre 1997.

[6] M. Demirekler, A. Haydar. *"Feature Selection Using a Genetics-Based Algorithm and its Application to Speaker Identification"*, Proceedings of the IEEE ICASSP'99, pp. 329—332, Phoenix, Arizona, 1999.

[7] C. Charbuillet, B. Gas, M. Chetouani, J.L. Zarader. *"Filter Bank Design for Speaker Diarization Based on Genetic Algorithms"* Proceedings of the IEEE ICASSP'06, Toulouse, Francia, Mayo 2006.

[8] M. Zamalloa, G. Bordel, L.J. Rodriguez, M. Peñagarikano, J.P. Uribe. *"Using Genetic Algorithms to Weight Acoustic Features for Speaker Recognition"*, Proceedings of the ISCLP´06, Pittsburgh, USA, September 2006.

[9] A.E. Rosenberg, C.H. Lee, F.K. Soong. *"Cepstral Channel Normalization Techniques for HMM-Based Speaker Verification"*, Proceedings of the ICSLP'94, pp. 1835—1838, Yokohama, Japón, 1994.

[10] Y. Linde, A. Buzo, R.M. Gray. *"An Algorithm for Vector Quantizer Design"*, IEEE Transactions on Communications, Vol. 28, No. 1, pp. 84—95, Enero 1980.

[11] L.J. Rodríguez, M.I. Torres. *"A Speaker Clustering Algorithm for Fast Speaker Adaptation in Continuous Speech Recognition"*, in P. Sojka, I. Kopecek and K. Pala Eds., Proceedings of the 7th International Conference on Text, Speech and Dialogue (Brno, República Checa, Septiembre 2004), pp. 433—440, LNCS/LNAI 3206, Springer-Verlag, 2004.

[12] M. Zamalloa, G. Bordel, L.J. Rodriguez, M. Peñagarikano. *"Features Selection Based on Genetic Algorithms for Speaker Recognition"*, IEEE Speaker Odyssey: The Speaker and Language Recognition Workshop, Puerto Rico, Junio 2006.

[13] D.E. Goldberg. *"Genetic Algorithms in Search, Optimization and Machine Learning"*, Addison-Wesley, 1989.

[14] F. Casacuberta, R. García, J. Llisterri, C. Nadeu, J.M. Pardo, A. Rubio. *"Development of Spanish Corpora for Speech Research (Albayzín)"*, in G. Castagneri Ed., Proceedings of the Workshop on International Cooperation and Standardization of Speech Databases and Speech I/O Assessment Methods, pp. 26-28, , Italia, Septiembre 1991.

[15] ECJ 15, http://cs.gmu.edu/~eclab/projects/ecj/.