TÉCNICAS PARA RECONOCIMIENTO AUTOMÁTICO DE LOCUTORES

MAESTRE R. JOSÉ

SERJE D. CHRISTIAN

UNIVERSIDAD TECNOLÓGICA DE BOLÍVAR

FACULTAD DE INGENIERÍA ELÉCTRICA Y ELECTRÓNICA

CARTAGENA

2007

TÉCNICAS PARA RECONOCIMIENTO AUTOMÁTICO DE LOCUTORES

MAESTRE R. JOSÉ

SERJE D. CHRISTIAN

Monografía presentada para optar al titulo de Ingeniero Electricista y Electrónico

Director

GÓMEZ V. EDUARDO

M.Sc. En Ciencias computacionales

UNIVERSIDAD TECNOLÓGICA DE BOLÍVAR

FACULTAD DE INGENIERÍA ELÉCTRICA Y ELECTRÓNICA

CARTAGENA

2007

Nota de aceptación

_____

_____

_____

_____

_____

_____

_____

Firma del presidente del jurado

_____

Firma del jurado

_____

Firma del jurado

Cartagena de Indias D.T y C 25 de julio de 2007

ÍNDICE DE CONTENIDO

ÍNDICE DE FIGURAS

ÍNDICE DE CUADROS

LISTA DE ABREVIATURAS

AFGS Arquitectura Farcon-Gori-Soda

ANNF *Artificial neural network filtering*

ASR *Automatic Speaker Recognition* – reconocimiento automatico de locutor

CDHMM *Continuos Density Hidden Markov Models* – Modelo oculto de markov de densidad continua.

CSR *Continuous Speech Recognition* - Reconocimiento robusto de la voz

CWR *Connected Word Recognition* - Reconocimiento de palabras conectadas

DFT *Discrete Fourier Transform*- Transformada discreta de Fourier

DHMM *Discrete* **HMM-** Modelos ocultos de Markov Discretos

DT *Decision Trees* – Arboles de decisión

DTW_1 *Dinamic Time Warping* – Alineación Temporal Dinámica

DTW_2 *Discrete transform wavwlet* – Transformada Wavelet Discreta

EM Esperanza Maximización

FAR *False Accentance Rate* **–** Tasa de falsa aceptación

FRR *False Rejeccion Rate* **-** Tasa de falsos rechazos

GMM *Gaussian Mixture Models* – Modelo de mezclas gaussianas

HCM *Hidden Control Model* – Modelo oculto de control

HMM *Hidden Markov Models* – Modelos ocultos de markov

IWR *Isolated Word Recognition* **–** Reconocimiento de palabras aisladas

LPC *Linear Prediction Coefficients* - Coeficientes de predicción lineal

LGRF *Locally Recurrent Globally Feedforward*

LVQ *Learning Vector Quantization*

MARM *Multi-Variate Autoregressive Models*

MFCC *Mel Frequency Cepstral Coefficients* – Coeficientes cepstrales en la escala mel

MFCC+D Delta **MFCC**

ML *Maximum Likelihood* – Máxima verosimilitud

MLP *Multi Layer Perceptrons* – Perceptron multicapa

OPCA *Oriented Principal Component Analisis*

PDM *Prototype Distributing Map*

POF *Probabilistic Optimum Filterig* – Filtrado óptimo probabilístico

POWP Perceptualy Oriented WP – Paquetes de onditas orientados perceptualmente

RBFNN Red Neuronal de Funciones De Base Radial

RIT Red Identificadora de Tipología

RNN *Recurrent Neural Network* **–** Redes neuronales recurrentes

RSR *Robust Speech Recognition* **–** Reconocimiento robusto del habla

RT Red de Tipología

SCHMM *Semi Continuous* HMM – Modelos semicontinuos de markov

SIV *Speaker Verification And Identification* – Identificación y Verificación de locutores

SR *Sparse Representations* – Representaciones ralas

SSR *Spontaneous Speech Recognition* **–** Reconocimiento de habla espontanea

STFT *short time Fourier transform* **–** Transformada de Fourier de corto periodo

SVM *support vector machines* **–** Maquinas de soporte vectorial

TDNN *Time Delay NN* – Redes neuronales con retardos temporales

TW *Trnasform Wavelet* **–** Transformada Wavelet

TWC *Trnasform Wavelet Continuous* **–** Transformada Wavelet Continua

UBM *Universal Background Model* – Modelo Universal

VoiceXML es el formato estándar XML del W3C para especificar diálogos interactivos por voz entre una persona y una máquina. Es el lenguaje más extendido en el desarrollo de portales de voz y comparte plataformas y herramientas de desarrollo del mundo WWW

VQ *Vector Quantization* **–** Cuantificación Vectorial

WP *Wave packects* **–** Paquetes de onditas

W3C *World Wide Web Consortium*

XML Xtensible Markup Language (lenguaje de marcas extensible)

GLOSARIO

AFRICADOS: sonidos que se deben a la concatenación de un sonido oclusivo con otro fricativo sin modificar demasiado el punto en el que se produce la obturación (en la lengua o los labios).

ALÓFONO: (de *alo-* y *–fono*). Cada una de las variantes que se dan en la pronunciación de un mismo fonema, según la posición de este en la palabra o sílaba, según el carácter de los fonemas vecinos, etc

APARATO FONADOR: conjunto de órganos que intervienen en la producción de sonidos. También llamado aparato vocal o articulatorio.

En los humanos, el aparato fonador está formado por la boca, la nariz, la faringe, la laringe, la tráquea, los pulmones y el diafragma. Los órganos que lo integran forman parte a su vez del aparato respiratorio y algunos del aparato digestivo.

CENTROIDE: aquella muestra de voz que representa mejor un conjunto de muestras. En otras palabras es aquella que guarda la menor distancia global con relación a otras muestras de voz

CODEBOOK: en su traducción directa es un libro de códigos, en el contexto de la cuantificación vectorial y otras técnicas es un espacio donde se encuentran las muestras de voz que sirven para la identificación de un locutor

FONEMA: cada una de las unidades fonológicas mínimas que en el sistema de una lengua pueden oponerse a otras en contraste significativo; p. ej., las consonantes iníciales de *pozo* y *gozo, mata* y *bata;* las interiores de *cala* y *cara;* las finales de *par* y *paz.*

FORMANTE: cada uno de los elementos de las palabras que le dan un significado gramatical o léxico. || **2.** *Ling.* Cada uno de los rasgos identificables de un sonido o de un fonema

FRICATIVOS: los sonidos fricativos /s/, /f/, /S/, son semejantes al ruido que producirá una ventisca de aire, y se producen cuando el aire proveniente de los pulmones pasa a través de un estrecho paso causado por los labios, la lengua, etc.

LOCUTOR: traducción más comúnmente utilizada para nombrar a un individuo que es identificado o autenticado por su voz.

NASALES: en estos sonidos la cavidad nasal se convierte en una extensión de la cavidad bucal (oral). Cuando en el momento de producirse una vocal se deja pasar aire por la cavidad nasal, se da lugar a las vocales nasalizadas. Son sonidos generalmente sonoros al haber excitación de las cuerdas vocales. Se produce un cierre total en la parte delantera del tracto vocal, bien sea por efecto de los labios, bien por el efecto de la lengua, que genera sonidos labiales (/m/), alveolares (/n/) o palatales (/G/).

OCLUSIVOS: el sonido empieza al cerrarse completamente el paso del aire en algún punto del tracto vocal. El aire que proviene de los pulmones genera una fuerte presión que finalmente sale en forma impulsiva al ceder la obturación del tracto. Esta obturación puede ser labial (/b/, /p/), alveolar (/d/, /t/) o palatal (/g/, /k/).

SONORO: sonidos que se producen cuando las cuerdas vocales se aproximan y comienzan a vibrar.

SORDO: sonidos que se producen cuando las cuerdas vocales se acercan entre si pero no llegan a vibrar.

VOICEPRINT: huella de voz de un locutor especifico.

RESUMEN

La monografía que se presenta en este documento esta orientada básicamente a la tarea del reconocimiento de un locutor. Para tal propósito, se ha empezado con la definición de esta tarea y de las formas validas que hay para llevarse a cabo. En el capitulo introductorio se establecen las diferencias entre los procesos de verificación e identificación desde la dependencia de texto, independencia de texto y texto solicitado. Para llegar a las técnicas de parametrización antes se citan los niveles de información de acuerdo con la facilidad o complejidad para extraer de estos, características de forma automática. Desde esta instancia se resuelve de manera general un sistema ASR a partir de cada una de sus etapas. En esta sección del documento se presentan los tipos de reconocimiento automático del habla de acuerdo con un conjunto de criterios. Cabe destacar que todas estas formas de reconocimiento se pueden hacer en función del portavoz y en consecuencia, todos los tipos de reconocimiento de habla que se exponen pueden ser validos para el reconocimiento de un locutor.

Los niveles de información que aparecen en el primer capitulo descifran el alcance de las técnicas de parametrización y cual es realmente el estado del arte en cuanto a extracción de características. EL balance que se presenta es bastante completo, se incluyen los métodos tradicionales, pasando por los métodos espectrales y retomando aquellos que aparecen en las últimas publicaciones. De igual modo, en la etapa de clasificación, que no es mas que la técnica o conjunto de técnicas para la clasificación de características. Se ordena casi que de manera cronológica. Claro esta que en la mayoría de ellas, se hace una presentación de los fundamentos y alcances en cuanto al reconocimiento se refiere. En la parte final de esta sección se establece un marco experimental y comparativo entre las técnicas que constituyen el estado del arte. Finalmente, de acuerdo con los procesos de identificación y verificación de la primera parte, se presenta un inventario de las aplicaciones de los sistemas SIV. Aquí se tienen en cuenta las industrias y perfiles que ameritan incluirse en la clasificación de estas aplicaciones. También se explican los procedimientos intermedios involucrados por todo proceso SIV. También se establece un marco categórico para las aplicaciones y se explican detalladamente las aplicaciones comunes que en algunos lugares del mundo se han aplicado o se encuentran vigentes. Esta sección concluye con las tendencias y aplicaciones del futuro que según los expertos dominaran muchos sectores de la economía.

INTRODUCCIÓN

La monografía que constituye este documento se ha enmarcado bajo el objetivo central de Realizar un estudio sobre el reconocimiento de locutores para establecer un marco teórico que permita afrontar con estrategias validas el diseño de arquitecturas integradas y no integradas de reconocimiento. Para tal propósito, se valoraron tres tareas que se reflejan en el desarrollo de este trabajo. Pues de principio se estimo que era conveniente analizar y desarrollar la conceptualización y fundamentación sobre la cual se basa el reconocimiento del hablante, establecer un marco comparativo de experimentaciones y resultados en tecnología de reconocimiento de locutores y finalmente presentar en forma organizada las tendencias, servicios y aplicaciones actuales de esta tecnología.

Una de las motivaciones para la realización de este trabajo se puede entender desde la perspectiva de los retos de los sistemas de seguridad en el mundo globalizado. Realmente el ser humano no debería preocuparse por desarrollar estrategias o dispositivos para la seguridad de las sociedades y sus funciones, pero vemos que la realidad es otra y la naturaleza de la misma en cuanto a seguridad se refiere hace que la ciencia se vea involucrada de manera imperativa en el desarrollo y la investigación de este campo. Esta es una de las razones por las que el reconocimiento biométrico constituye en la actualidad uno de las posibilidades tecnológicas más inmediatas para resolver las desventajas de los sistemas clásicos de seguridad que en la mayoría de los casos dependen de elementos complementarios (llaves, tarjetas) o códigos secretos (claves, pin, contraseñas) sobre los que se enmarca el concepto de seguridad. Desde este punto de vista es importante destacar que el complicado tratamiento e intransferencia de los rasgos biométricos quizás ha frenado un poco la evolución de sus aplicaciones tecnológicas. No obstante, los esfuerzos que se hacen en la actualidad para la construcción de sistemas más robustos y confiables generalmente parten de valoraciones comparativas entre los rasgos biométricos proclives de

ser aplicados en sistemas reales. Desde esta premisa, es evidente que las conclusiones de hoy apuntan a los sistemas integrados, pues los diferentes estudios que se han realizado en materia de sistemas de reconocimiento e identificación valoran los grados de libertad que ofrece un rasgo biométrico. Por esta razón se ha establecido de manera diferencial que los rasgos biométricos encontrados en secciones anatómicas como el iris, mapa de la mano, el rosto, huella dactilar y voz ofrecen información suficiente y de relativa confiabilidad para ser abordados dentro de un concepto de reconocimiento y en consecuencia de seguridad.
Si bien, estos son los más considerados de acuerdo con las nuevas tendencias y posibilidades, la voz es el rasgo biométrico con mayor grado de libertad tecnológica y la diferencia radica en que los pineros rasgos biométricos solo dependen de diferencias morfológicas propias de un rasgo físico , mientras que la voz presenta en la extensión de su representación rasgos aprendidos que contribuyen a diferentes niveles de información susceptibles de ser aplicados en el reconocimiento e identificación de un locutor.

La tarea de reconocer un locutor, sea por verificación o identificación, se hacen bajo el contexto de independencia y dependencia de texto y las dos constituyen el estado del arte en reconocimiento de locutor. El reconocimiento de locutor con independencia de texto es una modalidad que se ha resuelto a partir de la representación espectral de las diferencias fisiológicas inherentes a la señal de voz. Este es el caso de menor complejidad en el reconocimiento de locutores, pero a su vez es una de las opciones integrables. Pues se puede decir que este es el primer nivel en la pirámide de complejidad en cuanto al reconocimiento. Este es el punto de partida y que descifra el ordenamiento del inventario de técnicas que en este trabajo se presentan. La pirámide de complejidad se puede ver como el referente de las técnicas de parametrización y su relación con los alcances y limitaciones dentro de cada nivel. De igual manera, la conjugación de los niveles de información con las técnicas de caracterización define y proyecta la explicitud de las técnicas de clasificación.

1 GENERALIDADES

1.1 Convergencias del reconocimiento automático de la voz

La voz es una señal aleatoria que se puede caracterizar a través de diferentes técnicas y herramientas que pueden ser aprovechadas para extraer información de manera automática por otras unidades especializadas y entrenadas para el reconocimiento. Las tendencias más importantes en este contexto son:

- Detección del discurso: ¿Hay alguien ablando? (Detección de actividad)
- Identificación del sexo: ¿Cuál es el género? (hombre o mujer).
- Reconocimiento de idioma: ¿qué idioma se está hablando? (Inglés, español, etc.).
- Reconocimiento de voz: ¿Qué palabras son pronunciadas? (del discurso a la transcripción de texto)
- **Reconocimiento del locutor: ¿Cuál es el nombre del locutor? (Juan, Lisa, etc.)**

A continuación, se presenta un resumen cronológico de los progresos en las tecnologías de reconocimiento de locutores hasta el año 1996. En esta parte se presentan unas de las organizaciones más importantes en el impulso de estas tecnologías incluyendo las formas de parametrización, los métodos de clasificación y los resultados indicando la tarea específica (identificación y/o verificación). A esta altura del documento el lector seguramente no entienda algunos elementos de la siguiente tabla, pero en la medida que siga el documento entenderá por completo este resumen histórico del reconocimiento de locutores. En consecuencia, este apartado debe tomarse como una reseña histórica y no exactamente como un marco comparativo de las técnicas empleadas.

1.2-Progresos del reconocimiento automático de locutores

Cuadro 1. Resumen cronológico de los progresos en reconocimiento de locutores

ORIGEN	ORG.	RASGOS	METODO	ENT.	TEXTO	N	ERROR
Atal 1974	AT&T	Cepstrum	Patter Match	Lab	Dependiente	10	I:2%_0.5s V:2%_1s
Markel y Davis 1979	STI	LP	Long Term Statistic	Lab	Independiente	17	I:2%_39s
Furui 1981	AT&T	Cepstrum normalizado	Patter Match	Teléfono	Dependiente	10	V:0.2%_3s
Schuwartz 1982	BBN	LAR	Nonparametric pdf	Teléfono	Independiente	21	I:2.5%_2s
Li y wrench 1983	ITT	LP, Cepstrum	Patter Match	Lab	Independiente	11	I:21%_3s I:4%_10s
Doddington 1985	TI	Banco de filtros	DTW	Lab	Dependiente	200	V:08%_6s
Soong 1985	AT&T	LP	VQ	Teléfono	10 dígitos aislados	100	I: 5%_1.5s I:1.5%_3s
Haggins y wohlford 1986	ITT	Cepstrun	DTW likelihood Scoring	Lab	Independiente	11	V:10%_2.5s V:4.5%_10s
Attili 1988	RPI	Cepstrun, LP y autocorrelación	Projected Long Term Statistic	Lab	Dependiente	90	V:1%_3s
Haggins 1991	ITT	LAR, LP y cepstrum	DTW likelihood Scoring	Oficina	Dependiente	186	V:1.7%_10s
Tishby 1991	AT&T	LP	HMM (AR mix)	Teléfono	10 dígitos aislados	100	V:2.8%_1.5s V:0.8%_3.5s
Reynolds 1995	MIT-LL	Mel-cepstrun	HMM (GMM)	Oficina	Dependiente	138	I:0.8%_10s V:0.12%_10s
Reynolds y Carlson 1995	MIT-LL	Mel-cepstrun	HMM (GMM)	Oficina	Dependiente	138	I:0.8%_10s V:0.12%_10s
Che y Lin 1995	RUTGER	Cepstrun	HMM	Oficina	Dependiente	138	I:0.56%_2.5s I:0.14%_10s V:0.62%_2.5s
Colombi 1996	AFIT	Cepstrum y coeficientes de energía	HMM monofónico	Oficina	Dependiente	138	I:0.22%_10s V:0.28%_10s
Reynolds 1996	MIT-LL	Mel-cepstrun y delta mel-cepstrum	HMM (GMM)	Teléfono	Independiente	416	V:11%_3s V:7%_10s V:4%_30s

JOSEPH P, Campbell, Speaker recognition: a tutorial. IEEE Vol. 85, No. 9 (septiembre 1997).

1.3-Reconocimiento de locutores

La tarea de reconocer un locutor se puede expresar como: dado un segmento de habla H, y un hipotético locutor L, la tarea de reconocimiento de locutor, se basa en determinar si la locución H fue generada por el locutor L (verificación), o dada una cohorte de locutores (L1, L2...Ln) determinar si la locución fue generada por uno de éstos o no (identificación).
A grandes rasgos, un sistema de ASR actúa como un clasificador de patrones. Cada patrón está formado por un conjunto de características o parámetros, extraídos de una determinada locución y es "enfrentado" o comparado con distintos modelos generados para cada locutor. La salida del clasificador ofrece una verosimilitud o una medida de distancia, entre el patrón de entrada y el modelo; y en última instancia una decisión, basada en un umbral, que clasifica la locución como perteneciente a un determinado locutor.

1.3.1 Identificación automática de locutor.

Desde esta modalidad, no se demanda ninguna identidad del locutor. El sistema (Figura1) debe determinar automáticamente quién está hablando. Ahora, si el locutor pertenece a un sistema predefinido, done los locutores son conocidos, se trata de un sistema cerrado de identificación. Sin embargo, el grupo de locutores conocidos, en los sistemas cerrados de identificación, generalmente es mucho más pequeño que el grupo potencial de usuarios que procuran entrar. En consecuencia, esta ultima situación, se refiere al caso más general donde el sistema tiene que manejar individualidades que quizás no se han modelado y por ende no se encuentran en la base de datos de identificación. La adición de la opción "ninguno-de-los-anteriores" sobre un sistema cerrado de identificación hace que el sistema se convierta en un sistema abierto de identificación. En este caso, el funcionamiento del sistema puede ser evaluado usando una tasa de identificación.

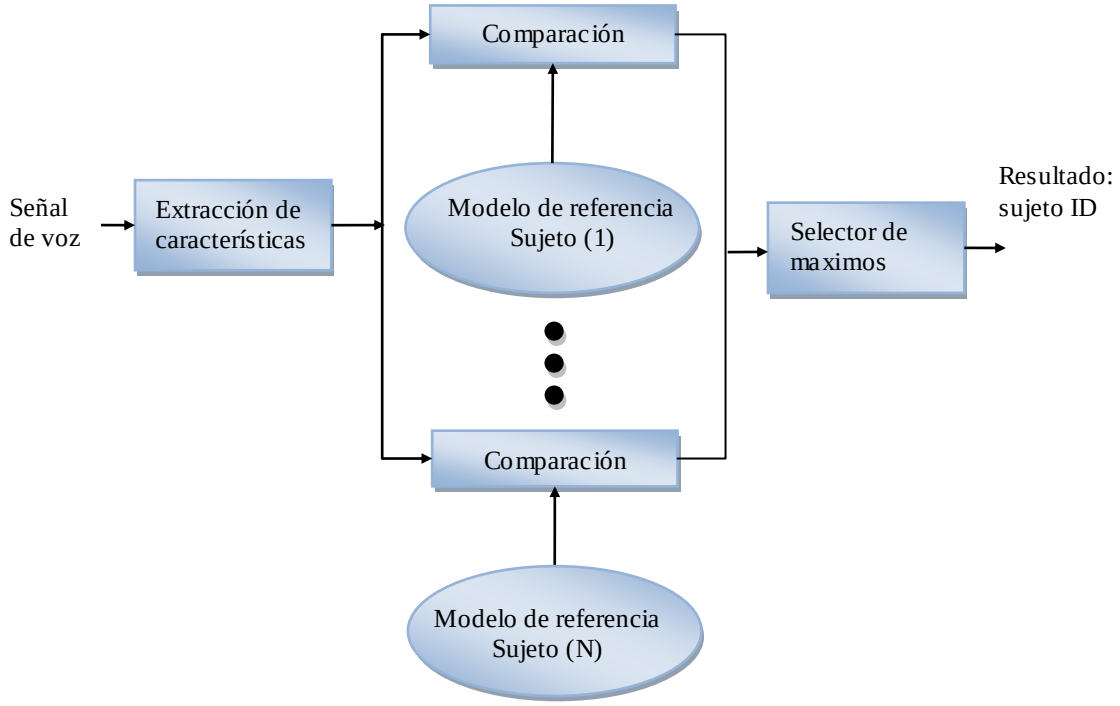
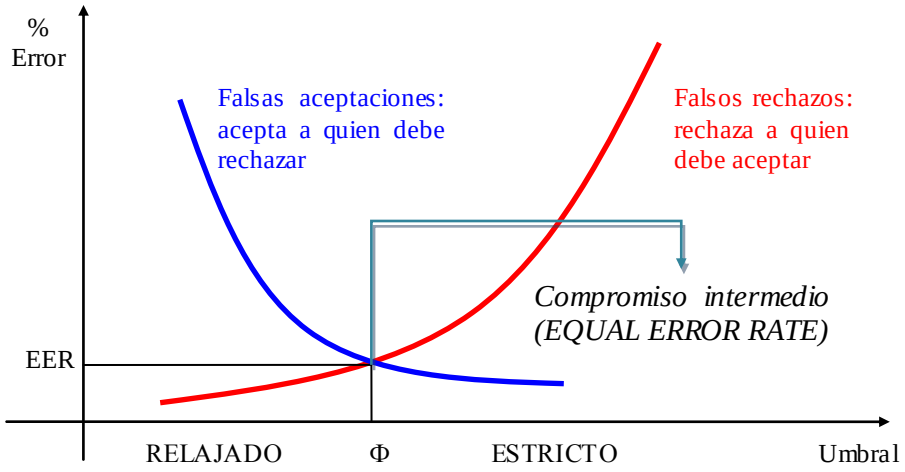


Figura 1. Esquema de Identificación de locutores

1.3.2 Verificación automática de locutor.

En este acercamiento la meta del sistema (Figura 3) es determinar si la persona es quién (ella /el) pretende ser. Esto implica que el usuario debe proporcionar una identidad y el sistema apenas acepta o rechaza a usuarios según una verificación exitosa o fracasada. A veces este modo de operación es denominado autentificación o detección. El desempeño del sistema se puede evaluar usando una tasa de falsa aceptación (FAR, *false accentance rate*), situaciones donde un impostor es aceptado y la tasa de falso rechazo (FRR, *false rejeccion rate*), situaciones donde un locutor se rechaza incorrectamente, también conocida en teoría de la detección como falsa alarma y misses respectivamente. Este marco nos da la posibilidad de distinguir entre la discriminabilidad del sistema y el sesgo de la decisión. La discriminabilidad es inherente al sistema de clasificación usado y el sesgo de la discriminación se relaciona con las preferencias y necesidades del usuario en relación con la importancia relativa de cada uno de los dos errores posibles (misses vs falsa alarma) que

se pueden cometer en la identificación del locutor. Esta compensación entre ambos errores tiene que ser establecida, generalmente ajustando un umbral de decisión. El desempeño se puede graficar (Figura 2) en un operador receptor característico (ROC) o en un diagrama de compensación de la detección del error (DET).



Estricto: *muy pocas falsas aceptaciones a costa de aumentar el número de falsos* rechazos

Relajado: *permite falsas aceptaciones a costa de muy pocos falsos* rechazos

Figura 2. Criterios de decisión y compensación

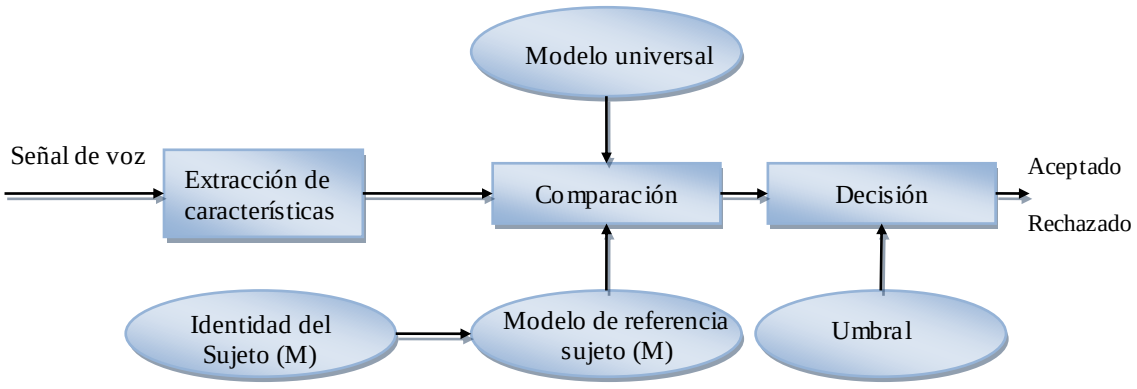


Figura 3. Esquema de Verificación de locutor

La respuesta del sistema será binaria: identidad aceptada o rechazada. Esta decisión binaria puede ser reformulada como un test de hipótesis entre las siguientes hipótesis:

H_0: x pertenece al locutor buscado.

H_1: x no pertenece al locutor buscado

La decisión óptima se obtiene a partir de:

$$T(x) = \frac{f\left(H_0/x\right)}{f\left(H_1/x\right)} \begin{cases} > \Theta \ aceptada \ H_0 \\ < \Theta \ aceptada \ H_1 \end{cases} \qquad [1]$$

Si las funciones de probabilidad f (...) se conocen exactamente y para un umbral Θ dado. T(x) se conoce como test de cociente de verosimilitudes (*likelihood ratio test*). Algunas elecciones comunes para f (...) son *Modelos Ocultos de Markov (HMM)*, *Modelos de Mezclas Gaussianas (GMM)* y *Redes Neuronales Artificiales (ANN)*.

Un sistema de *Verificación* de locutores típico opera de la siguiente manera: el modelo definido por la función f (H1/x) se entrena con las voces de muchos usuarios diferentes y se denota como *UBM (Universal Background Model)* o modelo universal. El modelo del usuario definido por la función f (H_0/x) se entrena solamente con la voz del locutor buscado. Finalmente cuando un locutor proporciona cierta identidad, se toma su señal de voz y se decide.

Según el contenido de la señal de voz empleada las modalidades de reconocimiento se clasifican en *independientes del texto y dependientes del texto*.

1.3.3 Independiente de texto (*Text Independent*).

Éste es el caso general, donde el sistema no conoce el texto hablado por la persona. Este modo de operación es obligatorio en aquellas aplicaciones donde el usuario no sabe que es evaluado por un sistema de reconocimiento, como sucede en algunas aplicaciones forenses,

o al simplificar el uso de un servicio donde la identidad es inferida para establecer un diálogo hombre/maquina, como se hace en ciertos servicios bancarios. Esto permite mayor flexibilidad, pero también aumenta la complejidad del problema. Si es necesario, para el reconocimiento del discurso se puede proporcionar el conocimiento del texto hablado. En este modo uno puede utilizar indirectamente la co-ocurrencia típica de las palabras del locutor, para evaluarlo desde una probabilidad gramatical. Este modelo de co-ocurrencia se conoce como n-gramas, y permiten determinar la probabilidad de que n palabras consecutivas sean pronunciadas por un locutor.

1.3.4 Dependiente de texto (*Text dependen*).

Este modo de operación implica que el sistema conoce el texto hablado por las personas. Puede ser un texto predefinido o inducido. En general, El conocimiento del texto hablado nos permite mejorar el funcionamiento del sistema con respecto a la categoría anterior. Este modo es de gran importancia en las aplicaciones donde se hacen estrictos controles de entrada al usuario de un sistema, o en las aplicaciones donde una unidad de diálogo va orientando al usuario.

1.3.5 Texto solicitado (*text prompted*).

Los sistemas dependientes e independientes de texto, tienen sus ventajas y desventajas, y además cada uno de estos requiere de un tratamiento específico. Así, por ejemplo, el rendimiento de los sistemas que utilizan reconocimiento *dependiente del texto* es mayor, sin embargo son más vulnerables cuando la clave es conocida por personas ajenas a su propietario. El riesgo de la clave es evitado usando independencia del texto, pero el rendimiento de los sistemas que utilizan esta opción es menor, y además, el peligro radica en que cualquier grabación de la voz de una persona puede ser utilizada para un acceso no permitido. Los sistemas *independientes del texto* necesitan también más entrenamiento que

los *dependientes del texto* ya que las características específicas de la frase no están disponibles.

Como alternativa surge una tercera vía intermedia entre las dos anteriores: t*exto solicitado* (*Text Prompted)*. Aquí es el sistema el que escoge el contenido de la muestra de voz. Esta elección puede ser hecha dentro de un conjunto reducido de palabras o frases, o sin ningún tipo de restricción en el texto a pronunciar por la persona. En el reconocimiento dependiente del texto, el sistema conoce a priori el contenido de la señal de voz, por lo tanto los modelos pueden ser mejor aproximados mejorando el reconocimiento. Al mismo tiempo, se evita la violación de la seguridad del sistema por la pérdida del secreto de la clave (ya que esta no es fija), pues es imposible conocer a priori el texto que pedirá el sistema.

1.4 Niveles de información

Para los seres humanos el reconocimiento de individuos es un problema menor, debido al uso de diferentes niveles de información. Después de los rasgos físicos la información que se encuentre por encima de este peldaño es considerada como información de alto nivel. Los niveles de información que están por encima del espectral son un complemento importante para los rasgos físicos ya que estos no son afectados por la variabilidad de canal. Algunos ejemplos de información de alto nivel en señales son: tasa de pausa, tono y patrón de sincronización, uso idiosincrático de palabras o frases, las pronunciaciones idiosincrásicas, etc. Mirando la historia de los primeros sistemas de reconocimiento de locutores se puede establecer por derecho que estos han sido principalmente basados en los rasgos físicos extraídos de las características espectrales de la señal de voz. Hasta ahora, las características derivadas del espectro de la señal de voz son las más eficientes en la implementación de sistemas de reconocimiento automático de locutores, esto se debe a que el espectro refleja la geometría del sistema que genera la señal. Por lo tanto la variabilidad en las dimensiones del tracto vocal se refleja en la variabilidad de espectros entre los locutores. La figura 4 resume diferentes niveles de información apropiados para el reconocimiento de un locutor, la parte superior está relacionada con los rasgos aprendidos y

la parte inferior con los rasgos físicos. Obviamente, no estamos limitados al uso individualizado de estos niveles, también podemos utilizar combinaciones y fusiones de datos para obtener un reconocedor más confiable. Los rasgos aprendidos, tales como semántica, dicción, pronunciación, idiosincrasia, etc. (relacionados con el estado socioeconómico, la educación, el lugar de nacimiento, etc.) son más difíciles de extraer automáticamente. Sin embargo, ofrecen un gran potencial. En la realidad, la aplicabilidad de estos sistemas de alto nivel de reconocimiento está limitada por la gran cantidad de información que se requiere para construir modelos robustos y estables del locutor durante el entrenamiento del sistema. Sin embargo, una herramienta estadística simple, tal como el n-grama, puede capturar fácilmente algunas de estas características de alto nivel. Diversos niveles de información extraída de la señal de voz se pueden utilizar para el reconocimiento del locutor. Los más conocidos son:

Figura 4. Niveles de información

1.4.1 Espectral.

La estructura anatómica del tracto vocal es fácil de extraer de una manera automática. De hecho, locutores diferentes tendrán espectros diferentes. (Localización y magnitud de picos) aun para los sonidos similares. Los algoritmos avanzados de reconocimiento de locutores se basan en los modelos estadísticos de las medidas acústicas, en intervalos cortos de tiempo, proporcionadas por un extractor de características. El modelo más popular es el modelo de mezclas Gaussianas (GMM), y el uso de las máquinas de soporte vectorial. La extracción de la característica es computada generalmente por métodos temporales como (LPC) (*linear predictive coding*) o métodos frecuenciales como (MFCC) (*Mel Frequency Ceptsral Coding*) o ambos métodos como la codificación lineal perceptiva (PLP) (*Perceptual Linear Coding*). Una característica agradable de estos métodos espectrales es que las escalas logarítmicas (amplitud o frecuencia), que imitan las características funcionales del oído humano, mejoran la tasa de reconocimiento.

1.4.2 Prosódico.

Las características prosódicas están relacionadas con la medida de la tensión, del acento y de la entonación. La manera más fácil de estimarlas es por medio del tono, de la energía, y la duración de la información. La energía y el tono se pueden utilizar de una manera similar a las características de corto periodo presentadas en el nivel anterior con un modelo GMM. Aunque estas características por sí mismas no proporcionan buenos resultados como las características espectrales, una cierta mejora se puede alcanzar combinando ambas clases de características. Obviamente, la fusión de diversos niveles de información puede ser utilizada. Pero en verdad, hay un potencial más amplio usando características a largo periodo. Por ejemplo, seres humanos que intentan imitar la voz de otra persona tratan generalmente de replicar la dinámica de la energía y del tono, en vez de los valores instantáneos. Las características prosódicas se pueden utilizar en dos niveles, en el más bajo, uno puede utilizar los valores directos del tono, energía o duración; en alto nivel, el

sistema pude computar probabilidades de co-ocurrencia de ciertos patrones recurrentes y comprobarlos en la fase de reconocimiento.

1.4.3 Fonético.

Se sabe que los mismos fonemas se pueden pronunciar de diversas maneras sin cambiar la semántica de una locución. Esta variabilidad en la pronunciación de un fonema dado, puede ser utilizada para reconocer las variantes de cada fonema, comparando la frecuencia de la co-ocurrencia de los fonemas de una locución (N-gramas de secuencias del teléfono), con los N-gramas de cada locutor. Con esto se pueden capturar las características dialécticas del locutor, dentro de estas se pueden incluir rasgos geográficos y culturales. Los modelos pueden consistir en N-gramas de secuencias telefónicas. Una desventaja de este método es la necesidad de un sistema automático de reconocimiento de voz capaz de modelar la matriz de confusión (es decir, la probabilidad de que un fonema dado sea confundido con otro). En todo caso, como hay bases de datos dialécticas disponibles para los idiomas principales, el uso de esta clase de información es hoy en día factible.

1.4.4 Idiolectical (*Synthactical*).

El trabajo reciente por G. Doddington "ha encontrado información útil de un locutor usando secuencias de palabras reconocidas"[1]. Estas secuencias se llaman la n-grama, y como se ha explicado anteriormente, consisten en la estadística de la co-ocurrencia de n palabras consecutivas. Estos reflejan la manera de usar la lengua por un locutor dado. La idea es reconocer locutores por el uso de palabras. Es bien sabido que algunas personas usan y abusan de varias palabras. A veces cuando intentamos imitarlos, no necesitamos emular su sonido o entonación. Apenas la repetición de esas palabras "preferidas" es

[1] G. Doddington. Speaker recognition base on idiolectal differences between speakers. Euro speech, Vol. 4 pp. 2521 – 2524, Aaoborg 2001.

bastante. El algoritmo entonces consiste en resolver el entrenamiento de n-gramas de un locutor en función de los datos de prueba. Para el reconocimiento, una clasificación se deriva de ambos n-gramas (usar, por ejemplo, el algoritmo de Viterbi). Este tipo de información es un peldaño más alejado de los sistemas clásicos, porque agregamos un nuevo elemento a los sistemas clásicos de seguridad. Un punto fuerte de este método es que no sólo considera el uso específico del vocabulario del usuario, sino que también tiene en cuenta el contexto y la dependencia temporal entre las palabras, que es más difícil de imitar.

1.4.5 Dialógico.

Cuando tenemos un diálogo con dos o más locutores, quisiéramos segmentar las partes que corresponden a cada locutor. Los patrones conversacionales son útiles para determinar cuándo ha ocurrido un cambio de locutor en la señal de voz (segmentación) y para agrupar segmentos del discurso que corresponden a un mismo locutor (agrupar). La integración de diversos niveles de información, tal como espectral, fonológico, prosódico o sintáctico es difícil debido a la heterogeneidad de las características. Diversas técnicas están disponibles para combinar diferentes tipos de información con notable carga de evidencias (Modelamiento bayesiano). Luego si es posible la integración, se pueden conseguir sistemas más robustos que aquellos que se pueden conseguir con la aplicación de características aisladas.

1.5 Tipos de reconocimiento

Los factores expuestos en el apartado anterior hacen inviable todavía el poder abordar el problema del ASR de forma global. Se hace necesario, por consiguiente, establecer hipótesis simplificadoras que restrinjan el campo de aplicación del sistema. De esta forma se introducen restricciones sobre diferentes aspectos de la señal de voz a reconocer, como

pueden ser el número y tipo de los locutores, tamaño del vocabulario, etc. Atendiendo a las restricciones impuestas, los sistemas de ASR pueden clasificarse según varios criterios.

1.5.1 En función del hablante

Según los grupos de locutores que se utilizan en el entrenamiento y evaluación del sistema, se pueden distinguir tres tipos de reconocedores. En los reconocedores mono-locutor el sistema se entrena y evalúa para un único hablante. En los reconocedores multilocutor el entrenamiento y evaluación se realiza sobre el mismo conjunto de locutores.
Finalmente, en los reconocedores independientes del hablante la evaluación del sistema se efectúa sobre un conjunto de locutores diferente del utilizado durante el entrenamiento.

1.5.2 En función de la manera de hablar y la aplicación

De acuerdo con el tipo de locución requerido para el funcionamiento adecuado del sistema existen los siguientes seis tipos. En los reconocedores de palabras aisladas (IWR, *Isolated Word Recognition*) el reconocimiento se realiza sobre palabras completas emitidas de forma aislada entre sí. En los reconocedores de palabras conectadas (CWR, *Connected Word Recognition*) se utilizan también las palabras como unidades de reconocimiento, pero éstas pueden ser emitidas secuencialmente con pausas entre ellas. Los reconocedores de voz o discurso continuo (CSR, *Continuous Speech Recognition)* realizan la tarea atendiendo a unidades inferiores a la palabra (fonemas, difonos, etc.) y sobre frases completas, sin necesidad de establecer silencios entre las palabras que las constituyen. En los reconocedores de palabras clave (Word *Spotting*) la locución suele ser cuidadosa, aunque no es una condición ya que su finalidad es la detección (o reconocimiento) de las palabras claves contenidas en el vocabulario. Los reconocedores de habla espontánea (SSR, *Spontaneous Speech Recognition*) buscan resolver el caso más complicado, ya que aquí suelen no respetarse algunas reglas del lenguaje, e inclusive es muy probable que en la señal aparezcan eventos acústicos diferentes al habla, como la tos, el estornudo, el hipo, etc.

1.5.3 En función de las distorsiones de la señal

También se pueden tener en cuenta las distorsiones producidas en la señal de voz por distintos factores. De esta forma podemos separar a los reconocedores en dos tipos. Los reconocedores de habla limpia son aquellos entrenados y probados en condiciones de laboratorio mientras que los reconocedores robustos (RSR, *Robust Speech Recognition*) permiten su utilización en ambientes reales, donde varios factores complican la tarea (ruido, reverberación, canal de transmisión, etc.).

1.6 Etapas de un sistema ASR

La estructura general de los sistemas de ASR tiene esencialmente tres módulos o etapas (Figura 5)

a) Procesamiento o análisis del habla (en inglés se conoce como *front-end*): en esta etapa se realiza algún tipo de análisis de la señal de voz en términos de la evolución temporal de parámetros espectrales (previa conversión analógica/digital de la señal). Esto tiene por función hacer más evidentes las características necesarias para la etapa siguiente y a veces también limpiar y reducir la dimensión de los patrones para facilitar su clasificación.

b) Clasificación de unidades fonéticas o modelo acústico: esta etapa clasifica o identifica los segmentos de voz ya procesados con símbolos fonéticos (fonemas, difonos o sílabas). A veces se puede asociar una probabilidad con este símbolo fonético, lo que permite ampliar la información presentada al siguiente módulo.

c) Análisis en función de reglas del lenguaje o modelo del lenguaje: en esta última etapa se pueden aprovechar las reglas utilizadas en la codificación del mensaje contenido en la señal para mejorar el desempeño del sistema y producir una transcripción adecuada.

Aquí se utilizan otras fuentes de conocimiento como la ortográfica, la sintáctica, la prosódica, la semántica o la pragmática.

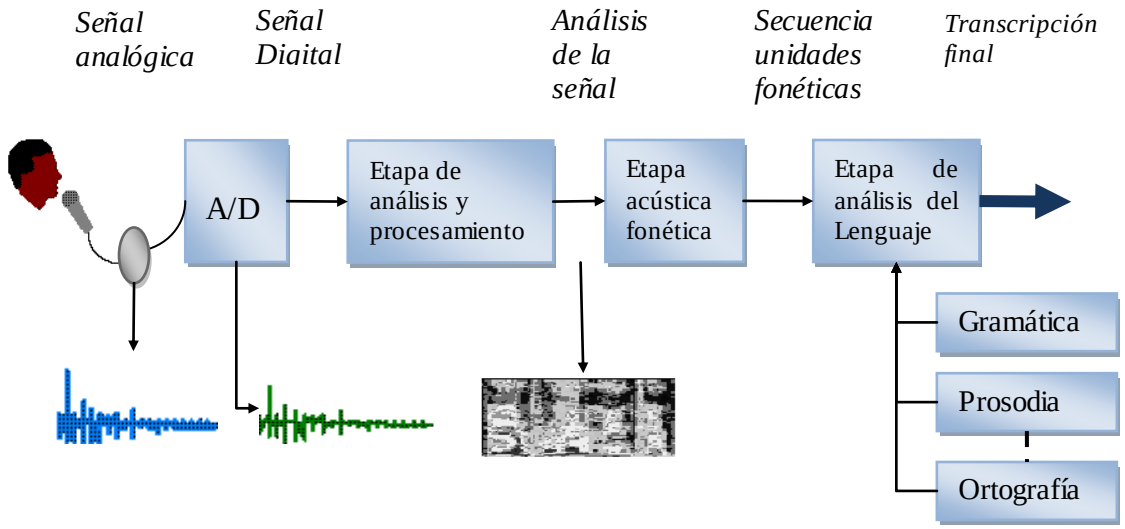


Figura 5. Etapas de un sistema ASR

1.7 Complicaciones propias del ASR

A pesar de la sencillez que parece presentar el problema del habla para los humanos, el estudio de la misma, muestra de forma inmediata, una enorme complejidad. En ella aparecen mezclados varios niveles de descripción, que interactúan entre sí. De esta forma, y según (Rabiner y Juang), "el problema del ASR presenta una naturaleza interdisciplinaria, y para solucionarlo es necesario aplicar técnicas y conocimientos procedentes de las siguientes áreas"[2]: procesamiento de señales, física (acústica), reconocimiento de patrones, teoría de la información y comunicaciones, lingüística, fisiología, informática y psicología. Además de la interdisciplinariedad expuesta, existen algunos aspectos prácticos relacionados con el habla que hacen del ASR una tarea difícil. Estos se pueden agrupar en seis categorías:

[2] RABINER L., JUANG B. H. Fundamentals of Speech Recognition. Signal Processing Series. Prentice-Hall, 1993.

a) Continuidad: en el lenguaje natural no existen separadores entre las unidades, ya que no existen silencios, en algunos casos, ni entre las palabras.

b) Dependencia del contexto: cada sonido elemental en los que se puede dividir el habla (fonema) es modificado por el contexto en el que se encuentra. De esta forma, se produce el efecto denominado coarticulación, según el cual los fonemas anterior y posterior a uno dado modifican el aspecto del mismo. Aparecen también efectos de orden superior, dependiendo de la pronunciación de un fonema, de su situación en una palabra o incluso en una frase.

c) Variabilidad: se pueden distinguir dos tipos de variabilidad. La variabilidad intra-hablante está relacionada con las modificaciones introducidas por un mismo hablante sobre diferentes pronunciaciones de los mismos fonemas o palabras. Incluso en idénticas condiciones, cada pronunciación presentará diferencias con las restantes debido a la diferente duración temporal. La variabilidad inter-hablante se debe a aspectos relacionados con el locutor y el entorno, ya que la señal obtenida dependerá de los dispositivos utilizados en su captación, del entorno donde se obtiene y, principalmente, de aspectos anatómicos particulares del aparato fonador de cada hablante.

d) Necesidades de almacenamiento: debido a las causas anteriores, se hace necesario procesar y almacenar grandes cantidades de datos.

e) Estructuración: la misma señal contiene información sobre varios niveles de descripción. De esta forma, una frase puede ser descrita a nivel semántico, sintáctico o fonético. Por otra parte, una señal de voz contiene información sobre el locutor que la emite. Así, es posible distinguir el sexo y la identidad de la persona a partir de la propia señal.

f) Inexistencia de reglas de descripción y redundancia: no existen reglas precisas capaces de describir los diferentes niveles en los que se presenta la información. Es más, cada uno de los niveles citados anteriormente aparece fuertemente relacionado con los demás, dificultando el análisis de la voz. En conclusión, un sistema de ASR tendría que determinar qué información de las citadas es de interés para lograr su objetivo a través de un conjunto de técnicas especializadas

2 TÉCNICAS DE ANÁLISIS

En la práctica, la mayoría de las señales se encuentran en el dominio del tiempo. Esta representación no siempre es la más apropiada cuando se tiene por objetivo su procesamiento para el ASR. En muchos casos, la mayoría de la información distintiva se encuentra oculta en el contenido frecuencial de la señal o en alguna otra forma de representación. La etapa de procesamiento convierte la señal de voz en algún tipo de representación paramétrica (generalmente con una cantidad de información considerablemente menor pero significativa) para su análisis posterior. Se supone que cuanto mejor sea el proceso utilizado para generar los patrones a utilizar, estos son más fáciles de identificar por el clasificador. Probablemente, la representación paramétrica más importante de la voz es la envolvente espectral de corta duración. Por lo tanto, los métodos de análisis espectral son generalmente considerados el núcleo de la etapa de procesamiento de señales en un sistema de ASR. Más recientemente comenzaron a considerarse enfoques basados en otro tipo de representaciones y son a los que se ha dado mayor énfasis en este trabajo. Las técnicas de procesamiento del habla exploradas pueden resumirse en:

2.1 Análisis tradicional

Corresponde esta clasificación con las primeras técnicas que se utilizaron en la caracterización de la voz, estas en su mayoría son herramientas de análisis en el dominio temporal. No obstante de las dificultades y alcances de este dominio para el análisis aun se siguen utilizando.

2.1.1 Análisis por tramos

Sea v(τ) la señal continua de voz para la variable real de tiempo τ. Después de un proceso de muestreo uniforme con período Tv, la señal de voz en la variable natural de tiempo discreto $0 < m \leq N_v$, se representa como $v(mT_v)$ o más simplemente $v(m)$.
Sea la señal $w(m;N_w)$ una ventana de análisis definida para $0 < m \leq N_w$, se dice que esta ventana posee un ancho $T_w=N_wT_v$. De la aplicación de la ventana de análisis temporal se obtienen los tramos de voz:

$$v(t; n) = w(n; N_w)v(tN_d + n); \qquad 0 < n \leq N_w \qquad [2.1]$$

Que representamos en notación vectorial como v_t. Se denomina paso del análisis por tramos al tiempo $T_d = N_dT_v$. Dadas las definiciones anteriores la variable de tiempo por tramos $t \in N$ queda acotada según $0 < t \leq T = (N_v - N_w)/N_d + 1 < \infty$. Si $T(k)$ es un operador para la transformación de dominio, se realiza el proceso de parametrización de la señal de voz según:

$$x(t, k) = T(k)\{v(t, n)\}, \qquad 0 < k \leq N_x \qquad [2.2]$$

Para la que se utilizara la notación vectorial simplificada como $x_t \in \mathbb{X} = \mathbb{R}^{N_x}$. se conoce a $\mathbb{X}$ como el espacio de las características con dimensión N_x. En esta sección se utilizará $0 < k \leq N_x$ como variable independiente discreta en el dominio transformado.

Ventanas de análisis

Las ventanas de análisis más utilizadas se definen para $0 < m \leq N_w$ según:

Ventana rectangular:

$$\omega_R(m; N_\omega) = 1$$

Ventana de Hanning:

$$\omega_h(m; N_\omega) = \frac{1}{2} - \frac{1}{2}\cos(2\pi m/N_\omega)$$

Ventana Hamming:

$$\omega_H(m; N_\omega) = \frac{27}{50} - \frac{23}{50}\cos(2\pi m/N_\omega)$$

Ventana de Barlett:

$$\omega_B(m; N_\omega) = \begin{cases} 2m/N_\omega & si\ 0 < m \leq N_\omega/2 \\ 2 - 2m/N_\omega & si\ N_\omega/2 < m \leq N_\omega \end{cases}$$

Ventana de Blackman:

$$\omega_K(m, N_\omega) = \frac{21}{50} - \frac{1}{2}\cos(2\pi m/N_\omega) + \frac{2}{25}\cos(4\pi m/N_\omega)$$

Estas ventanas pueden ser caracterizadas por el tamaño de los lóbulos de la magnitud de su espectro de frecuencias. La ventana rectangular posee el lóbulo central con menor ancho de banda pero la magnitud de los lóbulos laterales decae muy lentamente. La ventana de Blackman posee la mínima amplitud en sus lóbulos laterales pero su lóbulo principal tiene un ancho de banda tres veces mayor al de la rectangular. "Dado este compromiso entre resolución frecuencial y distorsión armónica en el proceso de ventaneo, para señales de voz suele utilizarse la ventana de Hamming que además, ofrece una posición media en cuanto al costo computacional de su aplicación"[3].

[3] Deller, J. R., Proakis, J. G., y Hansen, J. H. Discrete-Time Processing of Speech Signals. Macmillan Publishing, NewYork.

2.1.2 Coeficientes de energía.

Cuando se confecciona el vector de características para un ASR, es práctica corriente considerar variables que llevan información importante del tramo de voz considerado. Una de estas variables consiste en una medida de la energía que se define simplemente como:

$$\in (t) = log \sum_{k=1}^{Nv} v(t; k)^2 \qquad [2.3]$$

También suele agregarse una estimación de las derivadas temporales de todos los elementos calculados. Para un vector de características x(t; k) dado, se obtienen los coeficientes delta mediante la regresión:

$$\Delta x(t; k) = \frac{\sum_{j=1}^{N_J} j\big(x(t + j; k) - x(t - j; k)\big)}{2\sum_{j=1}^{N_J} j^2} \qquad [2.4]$$

Donde N_J es utilizado para suavizar la estimación a través de los tramos (generalmente $1 \leq N_J \leq 2$). Los coeficientes de aceleración $\Delta x^2(t, k)$ se obtienen por aplicación directa de la ecuación anterior a los $\Delta x(t, k)$.

2.1.3 Número de cruces por cero.

Se denomina “cruce por cero “al hecho de que muestras consecutivas tengan distinto signo algebraico, puesto que en este caso, entre muestra y muestra la señal tendrá que tomar obligatoriamente el valor cero.

La tasa de cruces por cero localizada se define matemáticamente como:

$$Z_S(m) = {}^{1}/_{N} \sum_{n=m-N+1}^{m} \frac{|sgn\{s(n)\} - sgn\{s(n-1)\}|}{2} . w(n-m) \qquad [2.5]$$

Donde la función signo (sgn) toma los valores:

$$sgn\{s(n)\} = \begin{cases} +1 & S(n) \geq 0 \\ -1 & S(n) < 0 \end{cases}$$

La tasa de cruces por cero nos da una idea del carácter sordo/sonoro de una señal (entendiendo que el carácter sordo va ligado a un tramo de alta frecuencia). Para mayor claridad se especifica que la densidad de cruces por cero del silencio por lo general es menor que la de los fonemas sordos y comparable con la de los fonemas sonoros.

2.1.4 Auto-correlación.

La función de auto-correlación localizada mide el parecido de la señal consigo misma en función de una variable de desplazamiento, k. Lo podemos expresar matemáticamente como:

$$R_S(K) = \frac{1}{N} \sum_{n=0}^{N-1-|m|} S_n . S(n + |m|), \quad |m| = 0,1,2,\dots,N-1 \qquad [2.6]$$

Así mismo se puede verificar que:

- La función de auto-correlación localizada es par.
- Tiene un máximo absoluto en *k= 0*, esto es, $R_S(0) \geq R_S(K), \forall\, K$
- $R_S(0)$ es igual a la energía (en señales deterministicas) o la potencia media (en señales periódicas o aleatorias)

Se verifica para valores de desplazamiento iguales al periodo de la señal, la auto-correlación máxima local, por lo que la auto-correlación de señales periódicas será también una señal periódica del mismo periodo

2.1.5 Relación de energía a altas y bajas frecuencias

En los segmentos sonoros la energía se concentra mayoritariamente por debajo de 1KHz mientras que en los sonidos sordos, ésta se concentra por encima de los 2KHz.

2.2 Técnicas de análisis espectral

Se incluyen dentro de esta clasificación aquellas tencnicas que aparecienron conjutamente con la trasformada de furiaer y cuyo análisis se da en el dominio de la frecuancia

2.2.1 Fourier (transformada de Fourier enventanada).

Una posible solución al problema de la localización tiempo-frecuencia proviene de los trabajos de Dennis Gabor[4] que introdujo una función que él denominó "ventana" que permitía una delimitación de la función a estudiar en el tiempo antes de realizar la descomposición frecuencial, según la Ecuación [2.7]. Definiremos la función ventana como g (t). Para poder discriminar, dicha ventana deberá ir "deslizándose" a lo largo del tiempo mediante traslaciones, determinadas por el factor de *traslación* τ. El procedimiento es ampliamente conocido y sencillo. Se toma la señal a analizar, se multiplica con la ventana en la posición adecuada y seguidamente se lleva a cabo la transformación frecuencial:

$$\hat{f}(\omega) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} f(t) e^{-j\omega t} dt \qquad [2.7]$$

[4] D. Gabor. Theory of Communication. Journal IEE, 93: 429-457, 1946.

$$S_\tau(\omega) = \hat{f}_g(\tau, \omega) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} \left(f(t)g(t-\tau)\right) e^{-j\omega t}\, dt \qquad [2.8]$$

La idea básica es conseguir que la función g(t) enfatice la función a estudiar en el instante de tiempo τ eliminando al máximo la aportación de los instantes de tiempo colindantes. Si definimos $f_g(t, \tau)$ como la función resultante tras el enventanado $f_g(t, \tau) = f(t).\, g(t-\tau)$ buscaremos que:

$$f_g(t, \tau) \sim \begin{cases} f(t) \text{ pata instantes de tiempo } t \text{ cercanos a } \tau \\ 0 \text{ para instantes de tiempo } t \text{ lejanos a } \tau \end{cases}$$

Con este objetivo es clara la razón por la que a la función g (t) se la denomina "ventana": Con ella pretendemos percibir tan sólo la influencia de una pequeña porción de la función f (t) alrededor del instante de "observación" t =τ. Estamos, pues, estudiando la distribución de frecuencia alrededor del instante de tiempo de análisis, con lo que estamos alcanzando una cierta localización temporal inexistente en la transformada de Fourier. Esta aproximación es la que nos lleva a la Ecuación [2.8] conocida como Transformada de Fourier localizada en tiempo o enventanada en el tiempo o Transformada Gabor.

2.2.2 Análisis LPC (linear predictive coding).

Durante un tiempo preocupó obtener un modelo matemático del tracto vocal, a partir del cual se estimo que sería fácil aumentar las tasas de reconocimiento de los sistemas en vigor. El modelo mayormente mantenido es el de la concatenación de tubos acústicos que confieren al tracto vocal una función de transferencia del tipo todos polos (aunque pueden incluirse algunos ceros si se desea mejorar la codificación de sonidos nasales). El análisis de predicción lineal (LPC), introducido por primera vez en análisis de voz por Saito e Itakura en 1966, presupone precisamente esta premisa, prácticamente al mismo tiempo que Wiener acunara el término Predicción Lineal. Es una parametrización bastante adecuada para la extracción de los formantes, pues realiza un suavizado del espectro de cada uno de

los segmentos de voz considerados. Uno de los mayores inconvenientes asociados a esta técnica es su poca robustez, por lo que se han derivado soluciones alternativas. El tipo de espectro resultante de este análisis es una suavización del resultante con un análisis por transformada de Fourier enventanada. Se elimina la información asociada a la velocidad de oscilación de los sonidos (pitch) manteniéndose mayoritariamente aquella asociada a la posición (y ancho de banda) de los formantes. Los parámetros del tracto vocal son atacados ya sea por un tren de pulsos periódicos para los sonidos sonoros, o por una fuente de ruido aleatorio para los sonidos sordos

Modelo matemático. Si la señal de voz muestreada se expresa por medio de una combinación lineal entre el valor presente de la señal y los p valores anteriores, como en la ecuación [2.9], donde e[n] es una variable estadística con valor medio igual a cero y varianza σ^2, esta ecuación en diferencias significa que el valor presente de la señal x[n], puede ser predicho linealmente usando los valores previos de la señal.

$$e[n] = x(n) + a_1 x(n-1) + a_2 x(n-2) + \cdots + a_p x(n-p) \qquad [2.9]$$

El valor predicho de la señal en el tiempo **n**, está dado por la expresión:

$$\tilde{x}(n) = -\sum_{k=1}^{p} a_k \, x(n-k) \qquad [2.10]$$

$$e[n] = x(n) - \tilde{x}[n] \qquad [2.11]$$

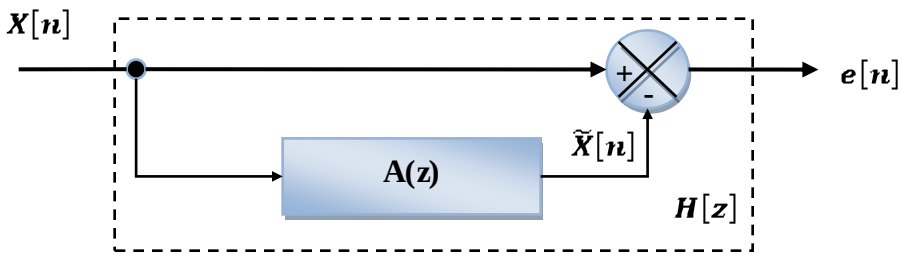


Figura 6. Esquema para obtener la señal predicha y el error de predicción

Los a_k son los coeficientes de predicción lineal y e[n], el error residual de predicción. El método LPC se conoce también como modelado autorregresivo (AR) de la señal y ha sido ampliamente utilizado porque es rápido, simple y permite estimar de forma efectiva los principales parámetros de las señales de voz. Si se considera que un filtro de solo polos con un número suficiente de polos es una buena aproximación del proceso de producción de la voz, este filtro puede modelarse así:

$$H(Z) = \frac{X(Z)}{E(Z)} = \frac{1}{1 - \sum_{k=1}^{p} a_k z^{-k}} = \frac{1}{A(Z)} \qquad [2.12]$$

Mientras el filtro inverso se define como:

$$A(z) = 1 - \sum_{K=1}^{P} a_k z^{-k} \qquad [2.13]$$

Si no se tiene información sobre la distribución de probabilidad de los a_k, un criterio razonable para estimar sus valores, es minimizar el error cuadrático medio. De esta manera, dada una señal $x_m[n]$, definida como un segmento de voz en la vecindad de la muestra m, los coeficientes LPC son los que minimizan el error total de predicción:

$$E_m = \sum_n e_m^2[n] = \sum_n \langle x_m[n] - \tilde{x}_m|n|\rangle^2 = \sum_n \left(x_m[n] - \sum_{j=1}^{p} a_j . x_m|n-j| \right)^2 \qquad [2.14]$$

Tomando la derivada de la expresión anterior con respecto a a_k e igualando a cero, se obtiene:

$$\sum_n e_m(n) . x_n[n-i] = 0, 1 \leq i \leq p \qquad [2.15]$$

Esta condición se conoce como principio de ortogonalidad (el producto punto es cero) y establece que los coeficientes de predicción lineal que minimizan el error de predicción son aquellos para los cuales el error es ortogonal a los vectores de muestras pasados. La ecuación [2.15] puede expresarse como una serie de p ecuaciones lineales:

$$\sum_n x_m[n-i]\,x_m[n] = \sum_{j=1}^{p} a_j \sum_n x_m[n-i]x_m[n-j], i = 1,2,\ldots,p \qquad [2.16]$$

Los coeficientes de correlación pueden definirse así:

$$\phi_m[i,j] \sum_n x_m[n-i]x_m[n-j] \qquad [2.17]$$

Las ecuaciones anteriores pueden combinarse para obtener las ecuaciones Yule-Walker:

$$\sum_{j=1}^{p} a_j \phi_m[i,j] = \phi_m[i,0], i = 1,2,\ldots,p \qquad [2.18]$$

Con la solución de estas ecuaciones se obtienen los coeficientes que minimizan el error de Predicción. Hay varios métodos que se han desarrollado para resolver las ecuaciones Yule-Walker eficientemente, entre estos están el método de la covarianza y el método de la autocorrelación. Si se analiza el comportamiento en el dominio de la frecuencia del filtro IIR de predicción lineal:

$$H\left(e^{jw}\right) = \frac{G}{1 - \sum_{K=1}^{P} a_k e^{-jwk}} \qquad [2.19]$$

Se observa que su respuesta sigue las variaciones lentas del espectro de la señal, en forma de envolvente. A medida que se incrementa p, la envolvente contiene más detalles del espectro y el error de predicción se reduce. Sin embargo, el valor de **p** no debe ser

demasiado alto, porque el filtro empieza a modelar los armónicos, es decir, la fuente de sonido.

2.2.3 Análisis Cepstral.

El análisis homomórfico o cepstral se basa en la suposición de que la voz es el resultado de la convolución de una función de excitación (generada en los pulmones) con la respuesta impulsional del tracto vocal. Se pretende pues deconvolucionar las señales de voz para obtener por un lado la señal de excitación y por el otro la respuesta del tracto vocal. Se emplea una transformación logarítmica para transformar productos (de las respuestas en frecuencia) en sumas. El cepstrum se define como la transformada inversa del logaritmo de la magnitud de la transformada de Fourier de la señal. Los bloques de la figura 8, ilustran este procedimiento:

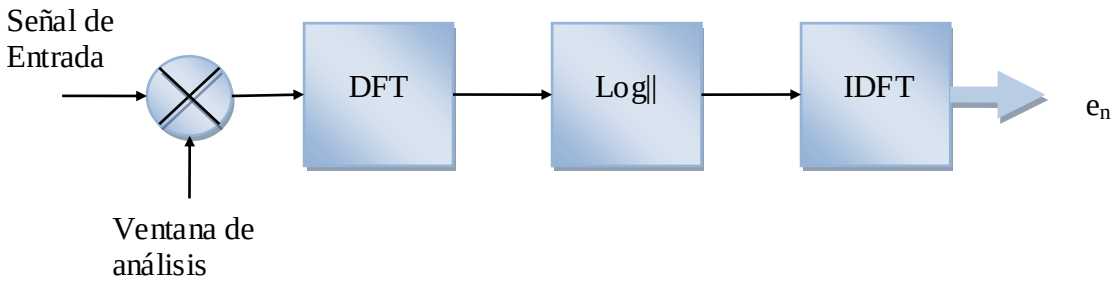


Figura 7. Esquema de obtención de los coeficientes cepstrum

$$e\{x|n|\} = F^{-1}\{log|F\{x|n|\}|\} \qquad [2.20]$$

De hecho las primeras aplicaciones del análisis cepstral fueron en el campo de la determinación del periodo fundamental: Se descompone el espectro como producto de dos componentes, una de variación rápida y otra de variación mucho más lenta (la envolvente espectral). Al tomar el logaritmo, el producto se convierte en suma, y podemos aplicar sin problemas el operador lineal de la transformada de Fourier.

$$s[n] = g[n] * h[n]$$

$$S(e^{jwn}) = G(e^{jwn})H(e^{jwn})$$

$$logS(e^{jwn}) = logG(e^{jwn}) + logH(e^{jwn}) \qquad [2.21]$$

Por tanto, el cepstrum puede expresarse así:

$$e(n) = F^{-1}\{log|X(e^{jw})|\} = F^{-1}\{log|G(e^{jw})|\} + F^{-1}\{log|H(e^{jw})|\} \qquad [2.22]$$

Esta propiedad define el cepstrum como una transformación homomórfica. El primer término en la ecuación [2.22] corresponde a la estructura fina del espectro y se ve reflejado como un pico ubicado en la parte alta de la función cepstrum. El segundo término corresponde a la envolvente y se presenta como una concentración de energía en los primeros milisegundos del cepstrum.

Si el cepstrum se halla con la transformada de Fourier de la señal (magnitud y fase), se denomina cepstrum complejo. El cepstrum puede calcularse empleando la DFT:

$$c(n) = \frac{1}{N}\sum_{K=0}^{N-1} log|X(k)|\, e^{j2\pi kn/N} \qquad [2.23]$$

2.2.4 Mel–cepstrun.

En sus trabajos originales Noll[5] denominó “cuefrencia" a la dimensión “temporal" asociada a los elementos aditivos obtenidos tras el logaritmo. El punto más crítico en todo el desarrollo radica en el operador logaritmo que rompe las linealidades, y que quizás no mantenga altas frecuencias como altas frecuencias y bajas frecuencias como tales (de hecho, este operador no mantiene las “cuefrencias" tan separadas como las “frecuencias").

[5] A. Michael Noll. Cepstrum Pitch Determination. The Journal of the Acoustical Society of America, 41: 293-309, 1967.

Una variante de los coeficientes cepstrum son los Mel-Cepstrum que se fundamentan en filtrar la señal por unos filtros de banda crítica para luego aplicar a sus N (20) diferentes salidas (log energía) x_k una transformada coseno discreta:

$$c_i = \sqrt{\frac{2}{N}} \sum_{J=1}^{N} m_j \cos\left(\frac{i\pi}{N}(J-0.5)\right) \qquad [2.24]$$

De hecho el análisis Mel-cepstrum no es más que un análisis en forma de banco de filtros, con una distribución adecuada de los mismos a lo largo de la frecuencia a considerar. Ver figura 9. En este caso y de acuerdo con la ecuación [2.24], m es cada una de las amplitudes del banco de filtros. Si recordamos el funcionamiento del sistema auditivo, veremos que básicamente implementa un banco de filtros (en número mucho mayor). Cualquiera de los sistemas de análisis de voz, que no presupongan ningún modelo de generación, acabará haciendo tarde o temprano, de manera directa, o más enmascarada, un análisis en forma de banco de filtros, en el cual se buscará conocer la distribución espectral a lo largo de la frecuencia para caracterizar los sonidos.

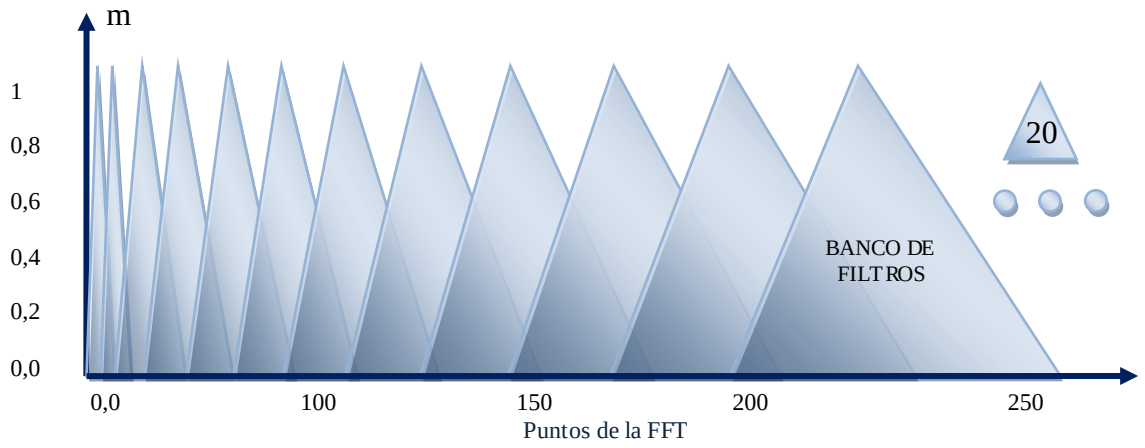


Figura 8. Ejemplo de un banco de filtros triangulares

Si el espectro de la señal de voz $X\left(e^{jw}\right)$ se estima con el análisis LPC, a partir de los coeficientes Cepstrum se pueden obtener los llamados coeficientes LPC-CEPSTRUM mediante la siguiente relación recursiva:

$$c(k) = a_k + \frac{1}{k}\sum_{i=1}^{k-1} ic_k a_{k-1} \qquad [2.25]$$

En la práctica esto se logra a través de filtros perceptuales que se derivan de una técnica denominada PLP. El objetivo que persigue es reproducir el comportamiento del oído. Utilizando unos filtros en frecuencia de una forma lo más similar posible ha los que utiliza el oído.

2.2.5 Delta Mel-cepstral (MFCC+D).

Dado que los MFCC reflejan las propiedades instantáneas de la voz, pues sólo dependen de la trama tratada, es conveniente obtener también las variaciones dinámicas del espectro ya que estas aportan información útil en los sistemas de reconocimiento. Inicialmente se pensó que para este trabajo serían útiles, sin embargo luego se comprobó que no lo son tanto para la detección de voz/ruido. La fórmula utilizada para realizar este cálculo está dada por la ecuación.

$$\Delta MFCC(t) = \frac{\sum_{i=-k}^{k} i.MFCC(t+i)}{\sum_{i=-k}^{k} i^2} \qquad [2.26]$$

Generalmente se utiliza una ventana de cálculo de 5 puntos, por lo que K sería igual a 2, siendo t la trama que se está analizando. Si se realiza este mismo proceso a partir de los parámetros transicionales (Delta) se obtienen los parámetros denominados Delta-Delta o de aceleración.

2.3 Modelos de oído

Surgen de la consideración de un enfoque biológico para la etapa de procesamiento, utilizando modelos de oído. El análisis de ciertos comportamientos del cerebro en función del oído ha permitido encontrar en las onditas la posibilidad de emular tecnológicamente algunas de las ventajas de este tipo de parametrización (biológica)

2.3.1 Análisis mediante onditas.

Las onditas aparecen a principios de los 90 constituyendo una alternativa atractiva por su carácter intrínsecamente transitorio. Sin embargo los intentos por superar definitivamente a las técnicas del estado del arte no han sido absolutos todavía, por lo menos en lo que se refiere a su integración con sistemas basados en HMM. Si bien el número de publicaciones de las onditas en el reconocimiento automático de locutores es escaso no es precisamente por su baja efectividad, aun se siguen explorando alternativas relacionadas para el mejoramiento y la optimización de esta herramienta. Entre 1991 y 1994 aparecieron sólo algunos pocos trabajos, la mayoría de ellos en anales de congresos. En muchos de estos trabajos se reportan algunas mejoras con respecto al procesamiento tradicional, que no suelen ser significativas, y por lo tanto esta técnica no logra desplazar a los enfoques más clásicos (como MFCC). "En Investigaciones anteriores, publicadas por la comunidad de ASR"[6]. Se realizó la comparación entre el análisis basado en onditas y el basado en la transformada de Fourier. A pesar de sus atractivas cualidades para el procesamiento de señales transitorias, la comparación fue desfavorable para las onditas (en realidad se exploró principalmente el caso de la DWT, en contraste con la STFT). Durante esta comparación se advirtió la multiplicidad de facetas que presentaba este tipo de representaciones relacionadas con onditas (elección del tipo de transformación, familia de onditas, parámetros, etc.), y la dificultad de brindar una respuesta definitiva acerca de cuál es la mejor alternativa para esta aplicación. Posteriormente se continuó la exploración de la

[6] RUFINER H. L. "Comparación entre análisis onditas y Fourier aplicados al reconocimiento automático del habla." Master's thesis, Universidad Autónoma Metropolitana, Diciembre 1996.

aplicación de onditas, tratando de superar algunos de los problemas detectados, como el de conseguir una adecuada resolución frecuencial en la zona de las frecuencias bajas. Para ello se recurrió a diferentes aproximaciones derivadas de los WP que mejoraron notablemente el desempeño de esta etapa con respecto a la DWT. Los WP, son considerados como una extensión del clásico análisis multiresolución (MRA). Este esquema MRA, propuesto por Mallat (1999), ha sido ampliamente utilizado en aplicaciones tales como: compresión, eliminación de ruido, caracterización de singularidades y detección de no estacionalidades en el análisis de señales e imágenes.

2.3.2 La transformada wavelet.

Un punto de partida común para todas estas aplicaciones (Fourier, Wavelets, ...) radica en las denominadas descomposiciones atómicas en las que, como en el caso de las series ortogonales de Fourier, se reduce una complicada señal a suma de subunidades (conocidas como átomos).

Como en el caso de Fourier, la forma concreta de estos átomos depende altamente de que es aquello que estamos buscando de la señal que deseamos descomponer. Es decir, no existe ni mucho menos un solo tipo de átomo en el que pueda descomponerse una señal. A principios de los ochenta J. Morlet[7] y sus colaboradores introdujeron en el mundo científico el término "wavelets" como unidades atómicas en las que se podían descomponer señales sin necesidad de utilizar comprometedoras "ventanas de análisis". La idea era descomponer (funciones u operadores) en diferentes componentes frecuenciales, pero de tal manera que cada una de las componentes tuviera una resolución de acuerdo con su escala. No deben confundirse los conceptos de escala y resolución. La noción de escala se relaciona directamente con su interpretación cartográfica. Una versión de una señal cualquiera x (t) aumentada de escala, será una señal similar pero muestreada a una tasa mayor $\left(x(t) \rightarrow \frac{1}{\sqrt{2}} x(t/2)\right)$. De forma similar, disminuir la escala de dicha señal lleva consigo la reducción

[7] J. Morlet, G. Arens, I. Fourgeau, y D. Giard. Wave Propagation and Sampling Theory. Geophysics, 47: 203-236, 1982.

de la velocidad de muestreo, manteniendo una forma de onda similar $\left(x(t) \rightarrow \sqrt{2}x(2t)\right)$. Partiendo de una señal conocida a una escala (de referencia), podremos llegar de muchas maneras a una nueva versión de dicha señal a otra escala (predeterminada), pero eso sí, sin tener siempre la misma señal resultante. Es precisamente la búsqueda de una única señal a la escala destino, lo que nos lleva al concepto de resolución: intuitivamente, esta depende de la cantidad de información presente.

En una señal a más información mayor resolución tendrá esa señal. Es muy importante recordar que si la señal original tiene resolución de referencia 1, nunca podremos aumentar dicha resolución sin añadir más información. En la Figura 11 podemos ver resumidas las operaciones asociadas a cambios de escala y de resolución.

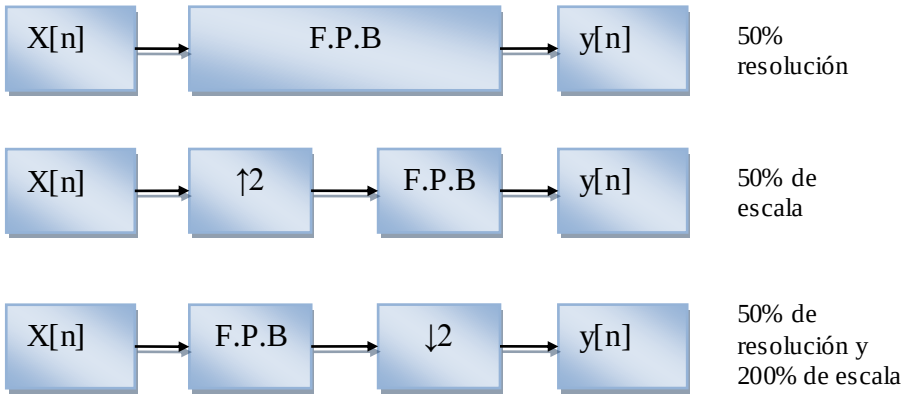


Figura 9 Representación mediante las operaciones de filtrado y diezmado, interpolado de los cambios de escala y de resolución.

Sustituyendo la señal a cada una de las escalas de trabajo por la mejor aproximación (siguiendo la Ecuación [2.27]) a dicha escala (una sola función). Moviéndonos a lo largo de las escalas iremos llevando a cabo un proceso de refinamiento mediante acercamientos a los puntos de interés (transiciones y discontinuidades en la señal) con el aumento de la escala de trabajo: esta es la idea básica del análisis multirresolución.

$$C_n = \int_a^b f(t)\overline{f_n(t)}dt \qquad [2.27]$$

Donde $\overline{f(t)}$ es el complejo conjugado de una función $f(t)$ y $f_n(t)$ es una familia de funciones ortonormales

El punto de inicio del análisis wavelet está en una función ψ (t) de variable real t que se conoce como función wavelet madre y que debe oscilar en el tiempo y estar bien localizada en el dominio temporal: Al oscilar la asemejaremos con una onda (en ingles *wave*) pero al estar acotada en tiempo quedará reducida a pequeña onda (en ingles *wavelet*). El concepto de localización temporal lo expresaremos en la forma habitual de rápido decaimiento hacia cero cuando la variable independiente t tiende al infinito. La idea de oscilación de la función se traduce en:

$$\int_{-\infty}^{\infty} \psi(t)\,dt = 0 \qquad [2.28]$$

$$\int_{-\infty}^{\infty} t^{m-1}\psi(t)dt = 0 \qquad [2.29]$$

Siendo (m-1) el valor del orden del momento de la función ψ (t).

A partir de la función madre, se generaran el resto de funciones de la familia mediante cambios de escala y traslaciones $\{\psi_{a,b}(t)\ ,\ a > 0\ ,b \in R\}$. La función madre, tradicionalmente, se ajusta a escala unidad. El parámetro de escala "a" queda asociado a un estiramiento o estrechamiento de la función madre. Recordemos rápidamente que dada una función localizada en tiempo s (t) su versión escalada s_a (t) definida como:

$$s_a(t) = \frac{1}{\sqrt{a}} s\left(\frac{t}{a}\right), \qquad a \in R, \qquad a > 1 \qquad [2.30]$$

Mantiene la misma forma que s (t) pero sobre un soporte más grande. Si el parámetro de escala se hace menor que 1 pero manteniéndolo siempre positivo (para evitar una inversión de la función) se obtiene una compresión del soporte de la función.

El parámetro "b", es el parámetro de traslación. A partir de la función madre ψ (t) generaremos las funciones wavelets $\psi_{a,b}$ (t) mediante las operaciones conjuntas de cambio de escala y traslación:

$$\psi_{a,b}(t) = \frac{1}{\sqrt{|a|}}\psi\left(\frac{t-b}{a}\right) \qquad [2.31]$$

Definiremos así, la Transformada Wavelet Continua de f (t) como:

$$W_f(a,b) = \int_{-\infty}^{+\infty} f(t)\,\bar{\psi}_{a,b}(t)\,dt \qquad [2.32]$$

$$\langle W_f(a,b) = \frac{1}{\sqrt{|a|}}\int_{-\infty}^{+\infty} f(t)\,\bar{\psi}\left(\frac{t-a}{b}\right)dt = \langle f(t), \psi_{a,b}(t)\rangle \qquad [2.33]$$

Donde hacemos uso del producto escalar:

$$\langle f(t), g(t)\rangle = \int_{-\infty}^{\infty} f(t)\,\overline{g(t)}\,dt \qquad [2.34]$$

La definición es coherente con lo expuesto sobre descomposiciones ortonormales anteriormente. De hecho, en sus trabajos originales en el tema Arens Grossmann y Jean Morlet[8] demostraron que si la función madre era real, entonces la familia de funciones definidas por su traslación y escalado era una base completa del espacio, y por lo tanto, se podía representar perfectamente cualquier función (cualquier señal de energía finita)

[8] J. Morlet, G. Arens, I. Fourgeau, y D. Giard. Wave Propagation and Sampling Theory. Geophysics, 47: 203-236, 1982.

mediante una combinación lineal de las funciones wavelets $\psi_{a,b}(t)$, calculando los coeficientes de tal descomposición en la forma habitual de producto escalar (Ecuación [2.27]).

La diferencia inicial con la técnica de Fourier es evidente y los primeros comentarios nunca fueron demasiado positivos. Afortunadamente, el ahínco pudo más que las criticas sin fundamento y hoy por hoy podemos hablar sin reparos de una solución de resultados equiparables a los de la teoría clásica de Fourier. De hecho, y ello ya es bastante paradójico, el trabajo de Dennis Gabor bien puede interpretarse como una descomposición wavelet. Su idea era tomar tonos puros que no tuvieran una duración infinita, sino que estuvieran acotados en el tiempo de la misma manera que sucede en el momento de interpretar cualquier partitura.

¿**Cual es pues el gran dilema con el que debe enfrentarse un análisis wavelet**?

Así como en aquellas transformadas donde "todo" radicaba en escoger la ventana (tamaño y forma), aquí los grados de libertad son igualmente ilimitados:

¿Cuál de las posibles familias de funciones $\psi_{a,b}(t)$ escogeremos?

El segundo problema radica en trabajar con una versión continua de la transformada o con una versión discreta de la misma. Expondremos a continuación las diferentes posibilidades para poder entender mejor la respuesta al segundo problema y pasar así a la siguiente sección donde se exponen otros tipos de consideraciones para un análisis wavelet.

La transformada wavelet-continua. La Transformada Wavelet Continua (TWC) viene definida por la Ecuación [2.32], donde tanto el parámetro de escala "a" como el parámetro de traslación "b" van variando de manera continua por todo el eje real (exceptuando el caso a = 0 por razones obvias relacionadas con el escalado). Podremos reconstruir la función original a partir de su TWC utilizando la formula de reconstrucción:

$$f(t) = \frac{1}{C_\psi} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \langle f(\tau), \psi_{a,b}(\tau) \rangle \psi_{a,b}(t) \frac{dadb}{a^2} \qquad [2.35]$$

Donde la constante C_ψ (denominada condición de admisibilidad) depende solo de la función wavelet madre ψ (t), de acuerdo con:

$$C_\psi = 2\pi \int_{-\infty}^{\infty} |\psi(\xi)|^2 \, |\xi|^{-1} d\xi < \infty \qquad [2.36]$$

Es importante en este punto la relación con la propiedad inicial de media nula que postulamos para la función wavelet madre (Ecuación [2.28]): Solo podremos tener valores finitos para esta constante si ψ (w) se anula a frecuencia w= 0. Si tomamos nuestra función madre ψ (t) tal que su transformada de Fourier esté localizada alrededor de la frecuencia w= $w_0 > 0$, y con ancho de banda Δw. Entonces, al dilatar la función $\psi\left(\frac{t}{a}\right)$ su espectro estará ahora centrado alrededor de la frecuencia $w=w_0/a$, con un ancho de banda modificado por el mismo factor Δw/a. Lo que se consigue es que el ancho de banda relativo del espectro (definido como la relación entre el ancho de banda y la frecuencia central) se mantenga siempre constante para cualquiera de las escalas a las que se quiera trabajar. Por el efecto de la dilatación, si nuestra función Wavelet madre tenía un ancho en tiempo definido por Δt, entonces, la nueva versión dilatada se habrá modificado hasta tener un ancho aΔt, pero la relación importante, es que para cualquier escala el producto entre el ancho de banda y el ancho en tiempo se mantendrá siempre constante a ΔtΔw: En todo momento, la resolución que se gana en uno de los dominios se pierde en el otro. Supongamos ahora que la transformada de la función Wavelet madre está centrada en el origen, entonces sus versiones dilatadas tan sólo modificarán su ancho de banda, pero continuaran centradas en el origen. La condición de admisibilidad (Ecuación [2.36]) asegura que la función wavelet madre no tenga contenido a frecuencia nula (o que éste resulte despreciable) (Ecuación [2.28]) y con ello que las versiones dilatadas resultantes de la función madre estén todas centradas a frecuencias diferentes.

La Ecuación [2.35] puede verse como en el caso de Fourier desde dos puntos de vista antagónicos:

- Como una forma de reconstrucción de una función f (t) conocida la transformada wavelet continua W_f (a, b) (Ecuación [2.32]). Corresponderá con la formula de inversión de la transformada (imprescindible en cualquier sistema de síntesis por análisis).

- como una expresión de descomposición atómica de la función f (t) como superposición de funciones de la familia $\psi_{a,b}$ (t) donde los coeficientes de la descomposición corresponden con la expresión de la transformada Wavelet Continua.

Como en el caso de las expresiones enventanadas de Fourier, la transformada f (t) → W_f (a; b) nos representará con mucha redundancia una función de una variable en un espacio bidimensional. Evidentemente, lo que resta por hacer es aprovechar tal redundancia para la extracción, precisa, de la información requerida de la señal.

Propiedades de la transformada Wavelet-continua. Veamos a continuación las principales propiedades de la Transformada Wavelet Continua (Algunas de las cuales están muy relacionadas con las de la Transformada Continua de Fourier).

Linealidad. Evidente partiendo de la linealidad del producto escalar.

Conservación de la energía. Como sucede con la formula de Parseval, la energía de una Función puede medirse en cualquiera de los dos lados de la aplicación lineal:

$$\int_{-\infty}^{\infty} |f(t)|^2 dt = \frac{1}{C_\psi} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} |W_f(a,b)|^2 \frac{dadb}{a^2} \qquad [2.37]$$

Traslación en el tiempo. Supuesta la función f (t) con transformada W_f (a, b), entonces, si trasladamos en el tiempo la función original f_1 (t) = f (t-b_0) obtendremos W_{f1} (a; b) = W_f ((a, b)-b_0). Por lo tanto, como sucede con la transformada de Fourier, ambas transformaciones lineales son invariantes a traslaciones en el tiempo de las funciones.

Escalado. Si cambiamos la escala de una función $f_1(f) = \left(\frac{1}{\sqrt{a0}}\right) f\left(\frac{t}{a0}\right)$ entonces su transformada se escala también $W_{f1}(a, b) = W_f\left(\frac{a}{a0}, \frac{b}{a0}\right)$ Si suponemos que $a_0 > 1$ entonces un átomo del espacio tiempo-frecuencia original de tamaño a1* b1 se transformará en un nuevo átomo de dimensiones reducidas $\left(\frac{a1}{a0} x \frac{b1}{a0}\right)$ Si debemos mantener la norma de la transformación resulta que se ha producido un aumento de la energía contenida en los nuevos átomos con el cambio de escala (en forma cuadrática), de allí el término $(da.db)/a^2$ de la fórmula de reconstrucción. Al producirse un cambio de escala, el átomo cambia tanto de tamaño como de situación, al producirse un cambio en los índices "a" y "b". Un escalado en la transformada Wavelet resulta equivalente a una modulación en la transformada de Fourier. Al hacer un barrido por todas las escalas, estaremos en cierta forma llevando a cabo un barrido (en el sentido contrario) en el dominio de la frecuencia.

Localización. En contraste con la STFT tenemos una localización variable a lo largo del plano tiempo-frecuencia. Concretamente, para altas frecuencias (escalas pequeñas) tenemos una muy buena localización temporal y para bajas frecuencias (escalas grandes) tenemos una muy buena localización frecuencial. Ello queda claro al transformar señales puras en los dos dominios, una delta de Dirac y una sinusoidal de duración infinita, tal como queda reflejado en la Figura 11, donde tenemos dos tonos puros a frecuencias f1 y f2, y dos impulsos localizados en los instantes de tiempo t1 y t2. Es importante destacar que la resolución frecuencial del tono de mayor frecuencia es peor que la del tono más grave. Al mismo tiempo las localizaciones temporales de los impulsos son peores a medida que la frecuencia se hace menor en el plano tiempo-frecuencia: La Transformada Wavelet tiene la capacidad de adaptar su resolución conjunta tiempo-frecuencia.

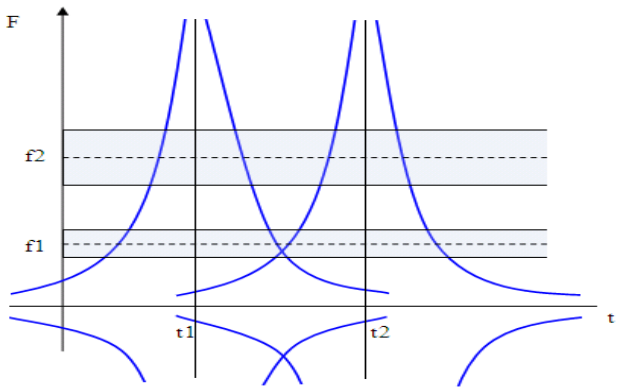


Figura 10 transformada wavelet continua para tonos puros e impulsos temporales

Regularidad. Esta propiedad es vital. Como veremos a continuación, el análisis wavelet permite asegurar la reconstrucción de funciones a partir de versiones a muy baja resolución. Ello solo es posible si las funciones wavelet madre y padre son continuamente derivables (Ecuación 2.38)[9]. El número de veces que puede llevarse a cabo la derivada sobre las funciones es su orden de regularidad. Imponer regularidad lleva a que los filtros que pueden derivar en un análisis en subbandas (o un análisis piramidal) Tendrán cierta suavidad, con lo que no introducirán discontinuidades artificiales en los coeficientes de la descomposición, si no están presentes en la señal analizada. Además imponer regularidad disminuye grandemente el efecto de los posibles errores de cuantificación en los esquemas de síntesis. Existen fórmulas iterativas para estimar el orden de regularidad de las funciones wavelets.

$$\varphi(t) = \sqrt{2}\sum_{n\epsilon Z} g_n \varphi(2t - n) \qquad [2.38]$$

2.3.3 La Transformada Wavelet Discreta.

Si tomamos la expresión de la transformada de Fourier enventanada en el tiempo, lo que hacemos es mirar la cantidad de energía contenida en una pequeña región del plano tiempo-

[9] Esta es la ecuación que representa la función wavelet padre, cuyo desarrollo lo podemos encontrar en la referencia [1] de la lista de referencias bibliográficas.

frecuencia alrededor del punto (τ,w). Normalmente se discretiza la transformada en ambos dominio:

$$S_n(m) = \hat{f}_g(n,m) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} \left(f(t) g(t - nt_0) \right) e^{-jmw0t} dt \qquad [2.39]$$

Como en toda teoría que desee conseguir una rápida y extendida aplicación "real" deberemos traducir las expresiones continuas que acabamos de ver a su versión discreta. El primer factor importante es que los dos parámetros que definen la transformada Wavelet, el de traslación "b" y el de dilatación "a" tomarán solamente valores discretos. Para la dilatación de la función Wavelet madre relacionaremos todos los valores como potencias "enteras" de una escala de referencia a_0, normalmente mayor que 1: $a = a_0^m$. Lo que tendremos será la posibilidad de estirar (estrechar) nuestros átomos de manera discreta. A medida que crezca el índice "m" iremos estirando la función, al ir aumentando la escala. Dado que estamos llevando a cabo un recubrimiento discreto del plano tiempo-frecuencia, y dado que el recubrimiento localizado es diferente a cada escala, la discretización del parámetro de traslación dependerá de la del parámetro de escala "a": Para escalas mayores, la traslación deberá ser mayor. Dado que el ancho de las funciones a cada escala es directamente proporcional con la misma, se toma una discretización del parámetro de traslación también directamente relacionado con la escala a la que se está trabajando: $b = nb_0a_0^m$. Los índices de nuestra descomposición serán siempre valores enteros, y el valor de referencia b_0 será siempre positivo por cuestiones de causalidad.

Con ello los átomos de nuestra descomposición atómica serán:

$$\psi_{m,n}(t) = a_0^{\frac{-m}{2}} \psi\left(a_0^{-m}(t - nb_0a_0^m)\right) = a_0^{\frac{-m}{2}} \psi(a_0^{-m}t - nb_0) \qquad [2.40]$$

En la Figura 13 podemos ver claramente la diferencia entre la discretización periódica y uniforme de la versión discreta de la STFT y la versión discreta de la Transformada Wavelet (WT) con traslaciones diferentes para cada una de las escalas de trabajo. En el

primer caso, por más que nos movamos por el plano tiempo-frecuencia siempre tendremos el mismo recubrimiento local del mismo; por contra, en la segunda solución, al movernos entre diferentes escalas, tendremos un recubrimiento diferente del plano, eso sí manteniendo constante el área recubierta.

Para la obtención de la transformada wavelet se irá llevando a cabo el producto escalar entre la función a descomponer y cada uno de los átomos. Al discretizar el plano tiempo-frecuencia ya no se consigue siempre la reconstrucción de la función analizada a partir de los coeficientes de la transformada. Al plantearse el problema de la discretización de la transformada y en base a la aplicación particular de cada problema, caben dos preguntas:

¿Podemos caracterizar nuestra función a partir de su transformada discreta? Este será el punto más relevante en la discusión de la participación de la transformada wavelet discreta y su funcionalidad en la extracción de información característica y diferencial en señales de voz a partir de su transformada discreta.

¿Podemos reconstruir la función de partida a partir de su transformada discreta? En cualquier aplicación de codificación de voz la respuesta afirmativa ante esta pregunta puede ser clave. A continuación veremos cuáles son las condiciones para que dicha reconstrucción sea posible.

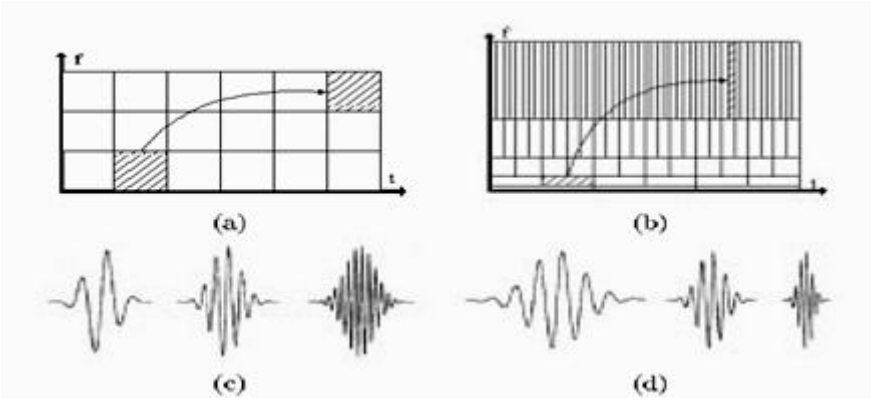


Figura 11 : Estructura del recubrimiento discreto del plano tiempo-frecuencia para la STFT y la TW discretas: (a) recubrimiento del plano STFT, (b) recubrimiento del plano WT, (c) bases STFT, (d) bases WT

El hecho de discretizar la Transformada Wavelet no lleva consigo eliminar la redundancia propia de la aplicación. Esta propiedad es utilizada en multitud de aplicaciones desde muy diferentes enfoques. Una de las ventajas de la redundancia reside en que los errores en el cálculo de la transformada se trasladan muy atenuadamente en la reconstrucción de la señal. En otras aplicaciones la redundancia se busca para extraer de ella lo estrictamente esencial para caracterizar aquello que se está descomponiendo.

Marcos. Si se analizan con detenimiento las dos cuestiones que se planteaban en la sección anterior veremos que son realmente dos casos de un único problema. Efectivamente, la respuesta positiva a ambas preguntas viene dada por la condición de que las funciones wavelets $\psi_{m,n}(t)$ constituyan lo que se denomina un marco. Bajo la condición de marco existirán funciones $\psi_{m,n}(t)$ en el espacio dual que permitirán la reconstrucción de una señal mediante la expresión:

$$f(t) = \sum_{m,n} \langle f(t), \psi_{m,n}(t) \rangle \tilde{\psi}_{m,n}(t) \qquad [2.41]$$

Si calculamos ahora el producto escalar de la Ecuación [2.40] con una función arbitraria g(t) de $L^2(R)$, tendremos:

$$\langle g(t), f(t) \rangle = \overline{\langle f(t), g(t) \rangle}$$

$$= \overline{\sum_{m,n} \langle f(t), \psi_{m,n}(t) \rangle \langle \tilde{\psi}_{m,n}(t), g(t) \rangle}$$

$$= \sum_{m,n} \langle g(t), \tilde{\psi}_{m,n}(t) \rangle \langle \psi_{m,n}(t), f(t) \rangle \qquad [2.42]$$

Vemos entonces, que cambiando g(t) por f(t) podemos expresar la fórmula de reconstrucción (Ecuación [2.40]) en su forma equivalente:

$$f(t) = \sum_{m,n} \langle f(t), \tilde{\psi}_{m,n}(t) \rangle \, \psi_{m,n}(t) \qquad [2.43]$$

Una forma intuitiva de entender la necesidad de imponer la condición de marco al conjunto de wavelets surge al tratar de contestar la siguiente pregunta:

¿Cómo podemos estar seguros de reconstruir nuestra función? Para ello deberíamos asegurar que dos funciones diferentes no pueden tener los mismos coeficientes en sus respectivas descomposiciones atómicas:

$$\langle f_1(t), \psi_{m,n}(t) \rangle = \langle f_2(t), \psi_{m,n}(t) \rangle \; Para\ todo\ m, n \in Z \rightarrow f_1(t) \equiv f_2(t) \qquad [2.44]$$

Además deberíamos pedir que

$$\langle f_1(t), \psi_{m,n}(t) \rangle = 0 \text{ para todo } m, n \in Z \rightarrow f(t) = 0 \qquad [2.45]$$

Para poder llevar a cabo cualquier tipo de estudio con los coeficientes discretos obtenidos de la transformada necesitaremos, inevitablemente, que el conjunto de todos los coeficientes de la descomposición sean en si mismos una secuencia perteneciente al espacio $L^2(Z^2)$, lo que equivaldrá a decir que los coeficientes son cuadráticamente sumables:

$$\sum_{m,n} \left| \langle f(t), \psi_{m,n}(t) \rangle \right|^2 \leq B \left| |f(t)| \right|^2 \qquad [2.46]$$

Por el otro lado, tenemos que ver bajo qué condiciones se produce la correlación de "tamaño" entre los coeficientes de la descomposición y la función descompuesta. Deberíamos ser capaces de encontrar una $\propto < \infty$ tal que para $\sum_{m,n} \left| \langle f(t), \psi_{m,n}(t) \rangle \right|^2 \leq 1$,

tuviéramos $||f(t)||^2 \leq \propto$ que reescribirse como una cota inferior $A||f(t)||^2 \leq \sum_{m,n}|\langle f(t), \psi_{m,n}(t)\rangle|^2$

Tenemos ya pues la condición de *marco.*

Definición 1. *Una familia de wavelets* $\{\psi_{m,n}(t),\ m \in Z, n \in Z\}$ *en* $L^2(R)$ *constituye un marco si existen* $A > 0$ *y* $B < \infty$ *Tales que para toda la función* $f(t) \in L^2(R)$

$$A\|f(t)\|^2 \leq \sum_{m,n}|\langle f(t), \psi_{m,n}(t)\rangle|^2 \leq B\|f(t)\|^2 \qquad [2.47]$$

Las constantes A Y B se denominan cotas del marco

Si el conjunto de funciones forma un marco, ya tenemos una condición suficiente para que podamos reconstruir cualquier función del espacio a partir de los coeficientes de la Transformada Wavelet. En el caso general A ≠ B, el cálculo de las funciones duales $\tilde{\psi}_{m,n}(t)$ puede obtenerse siguiendo una fórmula iterativa.

Cuando estamos cerca de la condición A ≈ B el conjunto de funciones duales $\tilde{\psi}_{m,n}(t)$ puede calcularse a partir de las funciones wavelets de partida $\psi_{m,n}$(t) como:

$$\psi_{m,n}(t) = \frac{1}{A}\psi_{m,n}(t) \qquad [2.48]$$

Sin embargo, aún cuando A ≈ B no tiene porque existir independencia lineal entre las funciones wavelets. Es importante destacar que debido a la dependencia lineal, que caracteriza a los marcos, si bien una función puede reconstruirse a partir de las funciones duales $\tilde{\psi}_{m,n}(t)$ la expresión no es única.

Sólo cuando A = B = 1 pasamos a tener una base ortogonal de funciones y, por lo tanto, una correspondencia uno a uno entre la Transformada Wavelet y la función.

Si tomamos un valor de a_0 muy cercano a la unidad y tomamos en consecuencia valores para el desplazamiento pequeños tendremos funciones que formarán un conjunto

sobredimensionado, aunque eso sí podremos reconstruir funciones sin ningún problema, y prácticamente ninguna restricción en la función wavelet madre. Si aumentamos el valor de la escala de referencia, entonces, tendremos muchas menos opciones para conseguir una reconstrucción de la función.

2.3.4 Bases ortonormales para análisis wavelet.

Para elecciones adecuadas de la función wavelet madre ϕ (t) y de los parámetros de referencia a_0 y b_0 la familia de funciones $\phi_{mn}(t)$ puede constituir una base ortonormal para $L^2(R)$. En particular, para la elección $a_0 = 2$ y $b_0 = 1$ se han obtenido funciones $\phi(t)$ con adecuadas propiedades de localización tanto en tiempo como en frecuencia que generan bases ortonormales:

$$\phi_{m,n}(t) = 2^{-m/2}\phi(2^{-m}t - n) \qquad [2.49]$$

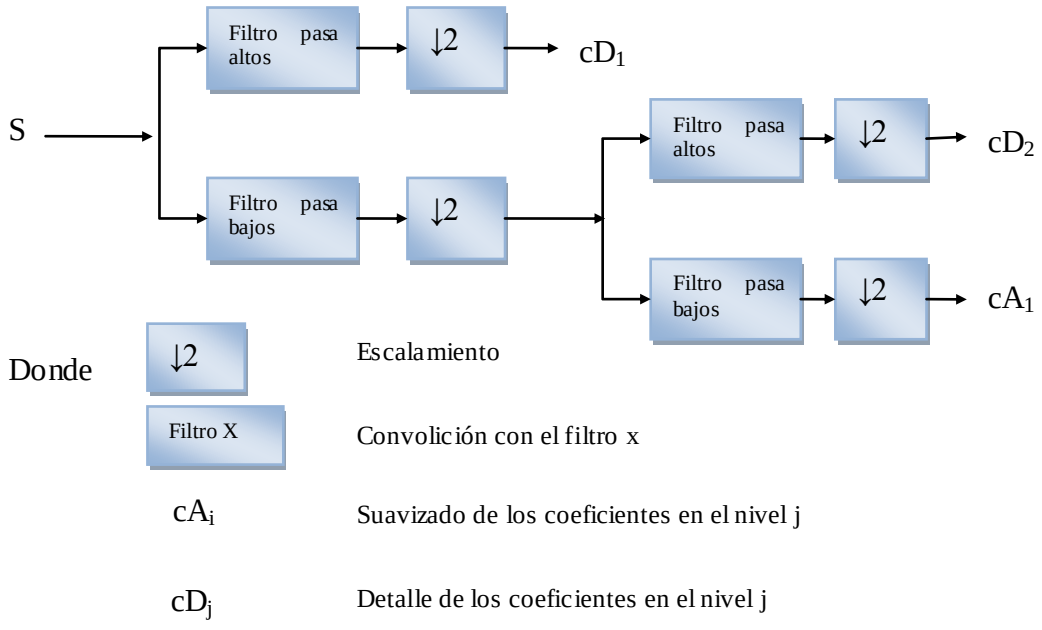


Figura 12 la transformada wavelet discreta. Estructura de proceso

2.3.5 Paquetes de onditas orientadas perceptualmente.

Se presenta a partir de un banco de filtros de manera que la base realice un análisis similar al que realiza el oído.

Generación de wavelets a partir de un banco de filtros. Viendo la relación entre la transformada wavelet discreta y el esquema de la codificación en subbandas, sólo nos queda preguntarnos si existe alguna diferencia entre ambas soluciones. La respuesta radica en la regularidad exigida para los filtros que convergerán en un análisis wavelet. Si tomamos de forma reiterada el filtro paso bajo y vamos buscando la función de transferencia asociada a la concatenación de filtros (con diezmado) tendremos:

$$G^i(z) = \prod_{l=0}^{i-1} G(z^{2l}) \qquad [2.50]$$

Donde G(z) es la transformada Z de la secuencia g [n] definida por:

$$G(z) = \sum_n g[n] z^{-n} \qquad [2.51]$$

Al aumentar hacia el infinito el número de iteraciones el tamaño del filtro resultante crecerá también hacia el infinito. Si definimos la función $g^i[n]$ como aquella que tiene por transformada Z $G^i(z)$, y consideramos una función $f^i(t)$ que es constante a trozos en intervalos de tamaño $1/2^i$ con valor $2^{i/2}g^i[n]$ en el intervalo $[n/2^i; (n + 1)/2^i]$. Puede demostrarse que la función resultante tiene soporte en el intervalo [0; L - 1]. Si hacemos tender a infinito las iteraciones, resultará que tendremos una función que tenderá a ser continua, (o quizás no consigamos la convergencia pasando por esquemas fractales). Para asegurar la convergencia a una función continua, deberá cumplirse la condición necesaria de que g [n] tenga un número adecuado de ceros en $z = \leftrightarrow 1$ (no es más que una condición de suavizado del espectro G(z) a su frecuencia crítica):

Proposición 1 *Para tener los K primeros momentos de las funciones wavelet madre nulos, su transformada Z debe tener K ceros en z = 1.*

Generación de un análisis wavelet. Con la presentación de la codificación en subbandas y su relación con el análisis wavelet queda pues claro que una posible manera de llevar a cabo un análisis wavelet (con posibilidades de esquemas rápidos) radica en el diseño de wavelets (ortonormales) para asegurar la reconstrucción de la señal con una escala de referencia $a_0 = 2$ con lo que habremos de obtener los coeficientes de los filtros paso alto y paso bajo asociados al análisis multirresolución e iterar hasta la precisión deseada. El problema de esta eficaz solución radica en la necesidad de tomar como escala de referencia $a_0 = 2$. Si miramos lo que esto significa desde el punto de vista de resolución en frecuencia de nuestra transformada, veremos que sólo dispondremos de una escala para cubrir la mitad superior de nuestro recubrimiento frecuencial, lo que puede resultar evidentemente escaso en numerosos problemas y planteamientos.

Soluciones Alternativas. Hemos visto como podía obtenerse una base ortonormal tomando como factor de escala de referencia a = 2. Lo que se presenta a continuación, son soluciones alternativas a las bases ortonormales que permitirán una síntesis por análisis de funciones, sin cumplir esta condición.

2.3.6 Paquetes de onditas *(Wave packets)*.

Una primera solución a este problema fue presentada por Wickerhauser, Coifman y Meyer[10].Con las “Wave-Packets". Básicamente se trata de llevar a cabo la iteración del análisis wavelet tanto por las bandas de baja frecuencia como por las de alta frecuencia, e ir profundizando hasta la resolución deseada. Una vez alcanzada la máxima profundidad se buscan cuales son los niveles (escalas) que minimizan la entropía introducida con la

[10] Mladen Victor Wickerhauser. Acoustic Signal Compression with Wave Packets. Informe técnico, Department of Mathematics, Yale University, agosto 1989.

codificación. Salvo por la regularidad de las funciones asociadas a este análisis, es exactamente la misma idea de un análisis en subbandas con profundidad variable para las diferentes bandas de acuerdo con la señal analizada.

Su idea reposa en el hecho de que cualquier señal genérica puede expresarse como la superposición (nuevamente la idea de descomposición atómica) de diferentes estructuras cada una definida, quizás, a una escala particular y en una posición temporal diferente. Debe buscarse en una tarea de análisis la extracción (y clasificación) de dichas estructuras básicas. Si pensamos que podemos expresar una nota musical (ejemplo de sonido elemental) en base a cuatro parámetros básicos: Intensidad o amplitud, frecuencia, duración temporal y posición temporal, definieron un conjunto de funciones elementales (wavelet packets) catalogadas en base a esos mismos cuatro parámetros conjuntamente con un quinto: La posibilidad de elección del conjunto de funciones básicas librería (es como si se escogiera el instrumento musical con el que vamos a interpretar una partitura). El sistema de análisis de señales, simplificado, que plantearon se asemeja a la comparación de los sonidos con cada uno de los elementos de la librería, con la que se ha decidido trabajar, buscando puntos (y pares) de máxima correlación. Sin plantear resultados reales en tareas de reconocimiento, en sus orígenes reconocían la importancia de esta capacidad de construcción de funciones por superposición del mejor conjunto de funciones posibles no solo en sistemas de compresión de datos sino también en aplicaciones de reconocimiento (extracción de partes significativas en funciones más complejas).

En la Figura 15a podemos ver cómo la solución propuesta permite llegar a la selección de una base wavelet: Si escogemos los coeficientes asociados a alta frecuencia en primer nivel, la parte resultante de filtrar paso alto el filtrado paso bajo en el segundo nivel y finalmente la salida de los dos filtros en el tercer nivel. Si deseamos llegar a un análisis en subbandas (Figura 15b), tendremos que seleccionar todos los coeficientes al mismo nivel. La aportación del trabajo se presenta cuando se selecciona cualquier otra solución (Figura 15c). Los trabajos de Herley y Vetterli[11] de bancos de Filtros ortogonales variantes en el tiempo son una solución alternativa con la misma Idea: Generación de estructuras de

[11] Martin Vetterli y Cormac Herley. Wavelets and Filter Banks: Theory and Design. IEEE Trans. Signal Processing, 40(9): 2207-2232, septiembre 1992.

bancos en subbandas dónde la forma, el número de filtros del árbol, e incluso la familia de filtros utilizados vaya variando a lo largo del tiempo, en función de la señal que se esté analizando en cada momento. Evidentemente, no podemos dejar pasar por alto la aportación desde el grupo de la Universidad de Stanford (en estrecha colaboración con S. Mallat), en sus trabajos de selección de bases para la detección de señales, la eliminación de ruido, y la eliminación en lo posible de la redundancia propia de la transformada Wavelet con los algoritmos de *Basis Pursuit*, y de *Matching Pursuit*. Dada su capacidad de descomponer señales a diferentes resoluciones, y su habilidad para la detección de señales impulsivas, la transformada wavelet tiene rápida aplicación en técnicas de eliminación de ruido (reconstrucción de señales a partir de versiones ruidosas)

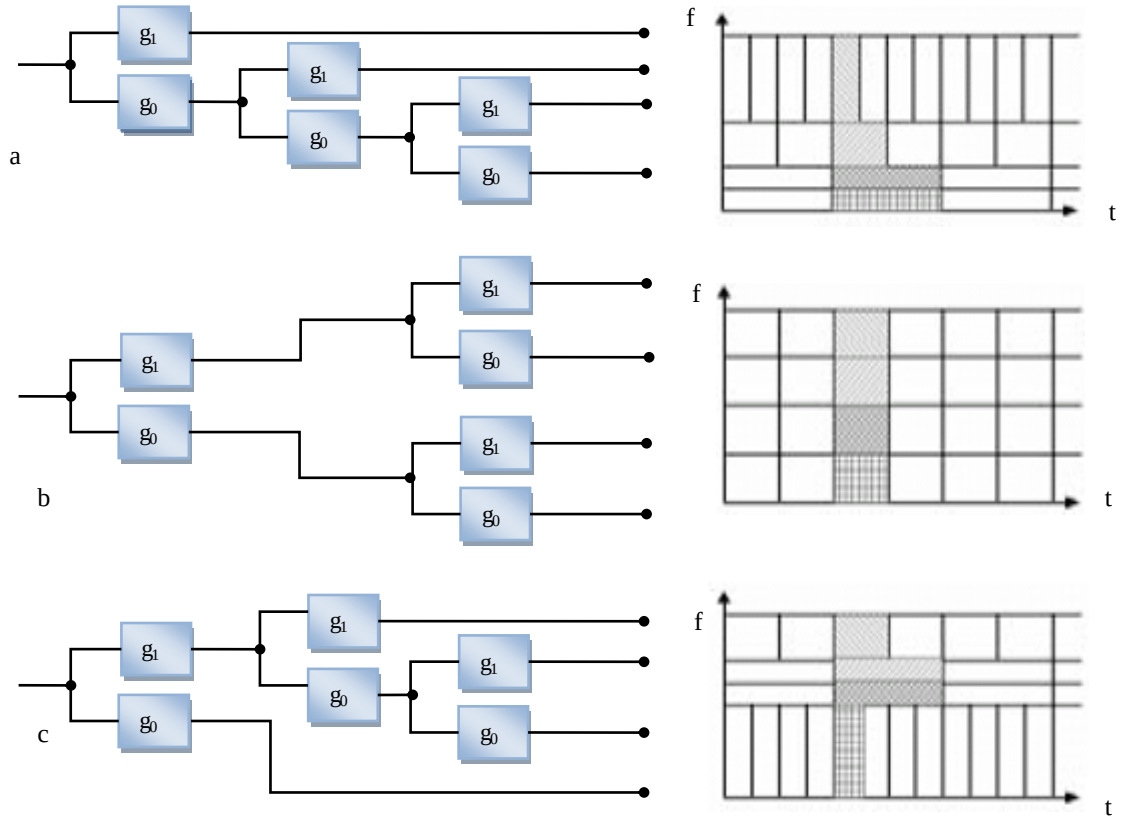


Figura 13 Posibles opciones del algoritmo de selección de la mejor base wave-packets: (a) Base Wavelet. (b) Análisis en Sub-bandas y (c) Cualquier otra base. Tenemos el esquema de análisis a utilizar, así como la división del plano tiempo-frecuencia.

2.3.7 Dilataciones Racionales.

Una segunda solución alternativa a las bases ortonormales busca escalas de referencia más pequeñas que 2 pero que permitan al mismo tiempo soluciones eficientes. Pascal Auscher[12] estableció las líneas básicas, para conseguir algo parecido a un Análisis Multirresolución, que derivaran en funciones wavelets trabajando con factores racionales (p.e. 2/3), con lo que podía llevarse a cabo una partición del eje frecuencial a diferentes velocidades en base a cada necesidad particular. De todas formas el mayor problema asociado a trabajar con factores de dilatación de referencia racionales está en la ausencia de la propiedad de traslación (en cada uno de los niveles de nuestro banco de filtros obtendremos una función wavelet que no resultará de la traslación de otra wavelet), que implica como demostraron Cohen y Daubechies[13] que no pueda construirse un análisis multirresolución con factores de dilatación racionales y filtros de respuesta impulsional finita (FIR), aunque en el límite se tienda a ello.

Proposición 2 *Dado cualquier número real M > 1, nos preguntamos si existe un conjunto finito de funciones $\phi_1(t)$, $\phi_2(t),\ldots,\phi_l(t)$, todas ellas pertenecientes a $L^2(R^2)$ tal que obtengamos una base ortonormal del espacio con la familia de funciones $M^{j/2}\psi_i\left(M^j t - k\right); j,k \in Z, 1 \le i \le l$*

Los Sistemas Diádicos. Si bien no todos los esquemas de estimación del periodo de pitch basados en transformada wavelet trabajan con estructura diádica, la mayoría sí que lo hacen. Incluso se han desarrollado esquemas rápidos para su evaluación.

[12] P. Auscher. Wavelet Bases for $L^2(R)$ with Rational Dilation Factor. En Mary Beth Ruskai (editor), Wavelets and Their Applications, pags. 439-452. Jones and Bartlett Publishers, 1992.
[13] Albert Cohen y Ingrid Daubechies. Orthonormal Bases of Compactly Supported Wavelets III. Better Frequency Resolution. SIAM J. Mathematical Analysis. To Appear In SIMAT.

2.3.8 La transformada wavelet Diádica.

Con la Ecuación [2.32] se definió la transformada Wavelet Continúa dependiendo de la función wavelet madre (ψ (t)), del factor de escala (a) y del parámetro de traslación (b). Después de esto se presentó el fundamento de las bases ortonormales y en particular el factor de escala a= 2. A partir de esta escala de referencia, todas las que se utilizan en el sistema son potencias negativas (habitualmente) de 2. Esta será la mejor manera de introducir la Transformada Wavelet Diádica de una señal x (t), en función del parámetro de traslación "b", del indicador de escala "j", de la función wavelet madre ψ (t) y de la señal a descomponer:

$$D_I WT_x(b, 2^j) = \frac{1}{2^{\frac{j}{2}}} \int_{-\infty}^{\infty} x(t)\, \bar{\psi}\left(\frac{t-b}{2^j}\right) dt \; ; j \in Z \qquad [2.52]$$

Una manera sencilla de obtener esta transformada es utilizando un banco de filtros de octava. Por el factor de escala particular de esta aplicación la relación entre los diferentes anchos de banda y las frecuencias centrales de cada una de las bandas es 2^{-j}. La propiedad más importante de la transformada diádica es su capacidad para detectar discontinuidades en la señal a analizar. Si una señal x (t), o sus derivadas, presenta discontinuidades, el módulo de su transformada Wavelet Diádica $D_I WT_x(b, 2^j)$ presenta máximos locales alrededor de dichas discontinuidades. Este punto es muy interesante para la estimación del periodo de pitch, dado que el momento de cierre de la glotis provoca cambios bruscos en la derivada del flujo de aire proveniente de los pulmones, que se refleja en transitorios en la señal de voz. Mallat[14], demostró que si la función wavelet madre era la primera derivada de una función de suavizado (con su energía concentrada en bajas frecuencias), entonces los máximos locales indicaban los puntos de transición, mientras que los mínimos locales del módulo de la transformada Wavelet Diádica corresponderán con los puntos de estacionariedad de la señal. Todos los esquemas de estimación del periodo de pitch con

[14] Stéphane G. Mallat. Multiresolution Aproximations and Wavelet Orthonormal Bases Of $L^2(R)$. Trans. Amer. Math. Soc., 315(1): 69-87, septiembre 1989.

transformada wavelet diádica tienen en cuenta esta propiedad, correlacionando luego la posición de máximos en sucesivas bandas (escalas).

2.3.9 Representaciones ralas.

Una de las actividades actuales es extender los resultados anteriores mediante WP e investigar sobre la posibilidad de encontrar una base, no necesariamente ortogonal, que maximice algún criterio (clasificación, dispersión, independencia, etc). Recientemente se han encontrado conexiones interesantes entre el análisis de señales mediante bases sobre-completas y la manera en la que el cerebro parece procesar algunas señales sensoriales. Este tipo de análisis puede dar representaciones que poseen muy pocos elementos activos o diferentes de cero, o sea sumamente ralas. Esta es una característica que comparten los sistemas sensoriales biológicos, y hace que la información sea fácilmente codificable en términos de trenes de pulsos o espigas. Existen varios métodos que permiten encontrar una representación rala de una señal si se posee una base sobre-completa adecuada. Así mismo se pueden diseñar los elementos de estas bases a la medida del tipo de "pistas" que se pretenden encontrar en la señal, o buscar automáticamente la base o diccionario que satisfaga algún criterio particular. Las representaciones ralas poseen también una robustez intrínseca al ruido aditivo, cuando este no pueda ser expresado fácilmente en términos de los elementos de la base. "Recientemente se han presentado estudios de las aplicaciones de estas técnicas para señales sintéticas"[15]. (Rufiner y Rocha), pero muy pocos trabajos con relación a señales del mundo real.

2.3.10 Filtrado optimo probabilístico.

Se trata de obtener una función que transforme la voz ruidosa en una estimación de la voz limpia en base a algún criterio de optimalizad como, por ejemplo, el de mínimo error

[15] RUFINER H. L., ROCHA L. F., GODDARD J. "Denoising of speech using sparse representations." Proc. of the International Conference on Acoustic, Speech & Signal Processing, 2002.

cuadrático medio entre ambas señales .La principal desventaja de estos métodos es que para cada nuevo entorno se debe calcular una nueva función de transformación, por lo que el entorno debe ser conocido a priori.

3. TÉCNICAS DE SELECCIÓN DE UNIDADES FONÉTICAS

Las técnicas de la etapa de clasificación de unidades fonéticas, son aquellas que están encaminadas a la representación de la señal de voz cómo una cadena de símbolos asociados con los eventos acústicos fonéticos. Para esta tarea existen varias alternativas de las cuales en este trabajo se incluyeron las siguientes:

3.1 Técnicas tradicionales

3.1.1 Cuantificación vectorial.

El objetivo de la cuantificación es reducir el volumen de los datos que vamos a tratar, perdiendo la mínima cantidad de información posible. Una vez que disponemos de los pesos que definen la distancia que vamos a utilizar y el codebook formado por un conjunto de centroides representativo del total de vectores que nos han servido para entrenar, estamos en disposición de realizar la cuantificación. Este proceso consiste en asignar a cada uno de los vectores (cepstrales…) el centroide más cercano, tal y como se indica en la figura 14. Cuando se trata del reconocimiento de locutores, para cada locutor se calcula un modelo, consistente en un codebook de vectores de características. El proceso seguido para realizar el codebook consiste en promediar aquellos vectores más semejantes, para formar un codebook de típicamente 6 bits (64 vectores). El vector resultante de cada uno de los promedios será el centroide (o centro de masas) de la agrupación (o clúster) de vectores usados para calcularlo, y será el que mejor los represente en media. Lógicamente con codebooks más grandes se puede obtener un mejor modelado, pero aumenta significativamente el número de vectores que deberá contener la secuencia de entrenamiento, y por tanto los tiempos necesarios para parametrizar y reconocer a los locutores. Esto es debido a que la estimación de cada codeword (o centroide) es equivalente a la estimación de una media, y la variabilidad en la estimación de una media es inversamente proporcional al número de vectores usado.

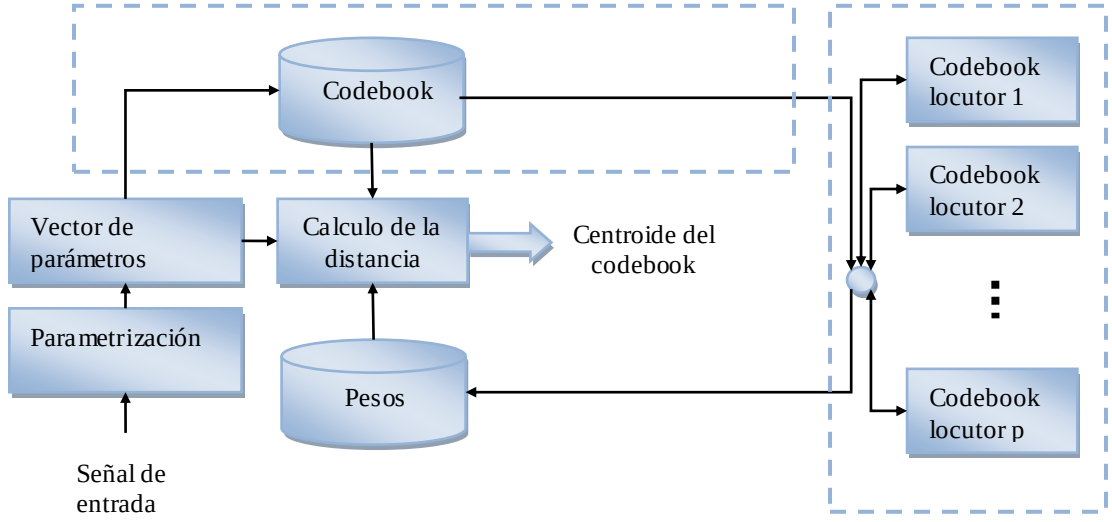


Figura 14 Cuantificación vectorial

Técnicamente, la cuantificación vectorial es el mapeo de un espacio Euclídeo k - dimensional en un conjunto *C* finito con N puntos de salida (llamados palabras de código o centroides). Es decir:

$$Q: \Re^k \rightarrow C$$

Donde $C = \{y_1, y_2, \dots, y_N\}$ Y $y_i \in \Re^k$ $\forall$ i =1, 2,..,N. El conjunto *C* es llamado *codebook* y tiene N vectores diferentes de dimensión k. Asociado con cada uno de los N puntos del *codebook* tenemos una región o celda. La i-ésima celda queda definida por:

$$R_i = \{x \in \Re^k : Q(x) = y_i\} \qquad [3.1]$$

De la definición de celda se deduce que $\cup_i R_i = \Re^k$ y $R_i \cap R_j \neq \phi \; \forall \; i \neq j$ o sea que las celdas forman una partición de $\Re^k$

Un cuantificador vectorial puede ser descompuesto en dos funciones más elementales como lo son: un codificador y un decodificador vectorial.

El Codificador Vectorial. El codificador *E* es el mapeo de $\Re^k$ en un conjunto de índices *J:* $E : \Re^k \rightarrow J$. Es importante notar que una partición dada de $\Re^k$, determina completamente la forma como el codificador asignará un índice a cada entrada dada. El codificador no necesita conocer el *codebook* para cumplir su función.

El Decodificador Vectorial. El decodificador D mapea el conjunto de índices *J* en el conjunto de representaciones C: $D: J \rightarrow C$.
Análogamente al caso del codificador, dado un codebook tenemos perfectamente determinado como el decodificador generará la salida a partir de un índice dado. El procedimiento de decodificación viene dado por una tabla de correspondencias, no es necesario conocer la geometría de la partición para llevarla a cabo.

Cuantificadores de vecino más cercano. Una clase especial de cuantificadores vectoriales, es la de los llamados cuantificadores Voronoi o de vecino más cercano. Veremos más adelante que un cuantificador vectorial debe pertenecer a esta clase para ser óptimo en el sentido de minimizar la distorsión promedio.
Se define un cuantificador de vecino más cercano como uno cuya partición en celdas viene dada por:

$$R_i = \{x / d(x, y_i) \leq d(x, y_j) \forall\, j \in J\} \qquad [3.2]$$

donde el *codebook* está dado por $C = \{y_i\} \; \forall \, i \in J$

Esto tiene como ventaja que durante el proceso de codificación no se necesita una descripción geométrica de las celdas de manera explícita. En lugar de eso, basta con conocer la distancia al *codebook* almacenado.

Condiciones de optimalidad. En esta sección se estudiarán las propiedades de optimalidad de los cuantizadores vectoriales. Estas propiedades son de gran ayuda para el diseño de cuantizadores pues proporcionan condiciones simples que un cuantizador vectorial debe

cumplir para ser óptimo y de ellas se deduce una técnica iterativa sencilla para mejorar un cuantizador dado
(*Algoritmo de Lloyd*).

La meta principal en el diseño de un cuantizador vectorial es encontrar un *codebook* y una partición que minimicen una medida de distorsión, considerando la secuencia completa de vectores a ser codificados.

Para el caso continuo, la distorsión *D* queda definida como: $D = E\left[d\left(x, Q(x)\right)\right]$ lo que es lo mismo que:

$$D = \int d\left(x, Q(x)\right) f_x(x)\, dx \qquad [3.3]$$

Donde $f_x(x)$ es la *pdf* conjunta del vector *x* y la integral es sobre todo el espacio k - dimensional.
Existen condiciones necesarias para que un codificador sea óptimo para un determinado decodificador y viceversa. El codificador queda determinado para la partición R_i y el decodificador queda determinado por el *codebook*. Entonces las condiciones de optimalidad son:

- Condición de vecino más cercano (*E* óptimo dado *D*)
- Condición de centroide (*D* óptimo dado *E*)
- Condición de probabilidad cero en los bordes

Condición de vecino más cercano. En primer lugar se considerará la optimización del codificador, dejando al decodificador fijo. Para un *codebook* dado, una partición óptima es la que satisface la condición de vecino más cercano: es decir a la región *Ri* se le asignan todos aquellos *i* que distan de *yi* menos que a cualquier otro vector del código.
Para un *codebook* la distorsión media puede ser acotada por:

$$D = \int d\big(x, Q(x)\big)\, f_x(x)dx \;\leq \int \min_{i\in I} d(x, y_i)\, f_x(x)dx \qquad [3.4]$$

y la igualdad se alcanza si *Q(x)* es la palabra de código que genera menor distorsión, es decir, el vecino más cercano. Entonces la partición óptima satisface: $Q(x) = y_i$ solo si

$$d(x, y_i) \leq d(x, y_i)\ \forall\, j \in J \qquad [3.5]$$

Condición de centroide. Ahora se estudiará la optimización del decodificador, dado al codificador. Podemos expresar la distorsión como:

$$D = \sum_{i=1}^{N} \int_{t_i}^{t_{i+1}} (x - y_i)^2 f_x(x)dx \qquad [3.6]$$

Basta con derivar la distorsión respecto a y_i para hallar el y_i óptimo. Finalmente cada región R_i será representada por su centroide definido como:

$$y_i = \frac{\int_{R_I} x f_x(x)dx}{\int_{R_I} f_x(x)dx} \qquad [3.7]$$

También es fácil ver que el centroide concuerda con la definición de centro de gravedad, de modo que el centroide es único. Para el caso discreto, la definición de centroide sigue siendo válida y en el caso en que cada vector tenga la misma probabilidad y la medida de distorsión sea la asociada al error cuadrático medio, el centroide coincide con el promedio aritmético.

Este resultado asume que todas las regiones tienen probabilidades distintas de cero de contener al vector de entrada. De no ser así se tiene el problema de *celda vacía*. Para una región con probabilidad nula el centroide no está definido, además tampoco tiene sentido

malgastar una palabra de código en semejante región. La solución que se toma generalmente, consiste en eliminar la celda vacía y partir la celda con mayor distorsión en dos. De esta manera el tamaño del *codebook* se mantiene constante y la distorsión disminuye.

Condición de probabilidad cero en los bordes. Existe una tercera condición necesaria para la optimalidad de un cuantificador que es útil en el caso discreto. Esta es la condición de probabilidad cero en los bordes o lo que es lo mismo que el vecino más cercano sea único.

Si un punto de la entrada, x_0, coincide con la frontera de las regiones R_i y R_j entonces dos particiones diferentes se pueden formar, asignando x_0 a R_i o bien a R_j En ambos casos la distorsión promedio es la misma, sin embargo, estamos cambiando de celda un punto de entrada con probabilidad distinta de cero, lo cual mueve los centroides de R_i y R_j, lo que implica que el codebook ya no es óptimo para la nueva partición.

Esta condición se cumple siempre cuando la entrada es una variable aleatoria continua pues el borde tiene volumen cero (y por lo tanto probabilidad cero). Esta condición es útil para el caso discreto pues la probabilidad puede ser colocada en puntos del borde, esto es, un vector de la secuencia de entrenamiento puede ser equidistante de dos vectores del código.

Diseño de un cuantificador vectorial. Las condiciones necesarias estudiadas para la optimalidad proporcionan las bases para un algoritmo iterativo que mejore un cuantificador vectorial dado. Se ha visto que las condiciones de optimalidad no aseguran que el cuantificador sea globalmente óptimo. Por lo tanto, la condición inicial torna un aspecto importante a tener en cuenta para obtener un buen resultado final: si se parte de un cuantificador bien diseñado la probabilidad de converger al óptimo será mayor.

Técnicas para diseñar cuantificadores. Se empezará estudiando algunas formas de obtener un buen codebook inicial. De hecho si éste es lo suficientemente bueno, no valdrá la pena correr algoritmos de mejora.

Random Coding. La idea más simple para encontrar un codebook de tamaño N es elegir aleatoriamente los vectores del código de acuerdo con la distribución de probabilidad de la fuente, lo que puede ser visto como un diseño Monte Carlo. La opción más simple cuando se diseña el codebook basándose en una secuencia de entrenamiento es elegir los primeros N vectores. Si la secuencia de entrenamiento es muy correlacionada es mejor elegir entonces, uno de cada K vectores. Desgraciadamente, este codebook no tendrá ninguna estructura útil y podrá comportarse bastante mal.

Pruning. Esta técnica se basa en la idea de comenzar con todos los vectores de la secuencia de entrenamiento como candidatos a integrar el codebook; y eliminarlos uno a uno según cierto criterio hasta que el conjunto final resulte ser el codebook.

Un método posible podrá ser el siguiente: se considera el primer vector de la secuencia como el primer vector del codebook. Luego se calcula la distorsión entre éste y el siguiente vector de entrenamiento. Si la distorsión es mayor que cierto umbral el vector siguiente se incluye también, sino es desechado. Con cada vector de entrenamiento nuevo se busca su vecino más cercano entre los vectores ya integrados al codebook, y, si la distorsión entre ambos es mayor que cierto umbral, el vector se integra al codebook sino se desecha. Este paso se repite hasta que se completa el codebook. Para una secuencia de entrenamiento finita puede resultar que no se encuentre el número de vectores necesario; en este caso se debe reducir el umbral de decisión y recomenzar.

Splitting. Inicialmente se elige el centroide de la secuencia de entrada y_0 como codebook de resolución 0 (con un solo elemento). Esta única palabra puede ser divida en dos palabras de código: y_0 y $(y_0 + \varepsilon)$ donde ε es un vector de módulo pequeño.

La resolución r se define como r = $log_2 N$

Este nuevo codebook tiene dos elementos por lo cual no puede ser peor que el original. El algoritmo iterativo de mejora puede ahora aplicarse a este codebook para obtener un buen codebook de resolución 1. Como siguiente paso se divide cada uno de los vectores del codebook en dos, repitiendo lo anterior. Se continúa de ésta manera, hallando un codebook de resolución r + 1 partiendo de un buen codebook de resolución r. Esta técnica provee un algoritmo completo de diseño de codebook desde la secuencia de entrenamiento.

El Algoritmo de Lloyd. Se discute ahora con detenimiento un algoritmo iterativo de mejora de codebook. Si la iteración continúa hasta la convergencia, un buen cuantificador vectorial (se desea que óptimo) se alcanza. La iteración comienza con un codebook que cumpla la condición de vecino más cercano hallado con alguna de las técnicas ya vistas. Entonces se encuentra un nuevo codebook (usando la condición del centroide) que sea optimo para la partición dada, luego encuentra una nueva partición para el nuevo codebook (usando la condición de vecino más cercano). Este nuevo cuantificador vectorial tiene una distorsión menor o igual al original. La aplicación repetitiva de este paso proporciona un algoritmo que reduce o no cambia la distorsión en cada paso. Si bien cada uno de los pasos es óptimo y directo, nada nos asegura que se hallará el cuantificador vectorial óptimo (el de menor distorsión para el conjunto de todas la posibles inicializaciones), ni siquiera uno que cumpla ambas condiciones a la vez.

1. m = 1; Se inicializa el *codebook Cm*
2. Dado C_m se halla la partición óptima mediante el vecino más cercano; Se resuelven los casos de igualdad.
3. Dada la partición se halla el *codebook* óptimo C_{m+1} mediante la condición de centroide. Se resuelven las celdas vacías generadas en el paso 2.
4. Se calcula la distorsión media para C_{m+1}, si el cambio desde la última iteración es menor que cierto umbral FIN, de lo contrario m = m + 1, y se vuelve al paso 2.

Muchos criterios de parada pueden ser usados. Uno de los más comunes es chequear si $1 - \frac{D_{m+1}}{D_m}$ es menor que cierto umbral adecuado.

Si el umbral se fija como cero, se tiene una sucesión de codebook cuyos valores de distorsión asociados son no crecientes. Si el algoritmo converge a un codebook en el sentido de que sucesivas iteraciones no producen cambios en él, entonces éste debe satisfacer las dos primeras condiciones necesarias de optimalidad.

Para un conjunto de entrenamiento finito, el Algoritmo de Lloyd converge en un número finito de pasos. Esto es fácil de ver dado que hay sólo un número finito de particiones, y la distorsión nunca crece, por lo cual, el algoritmo no puede volver a una partición que entregue un valor mayor de distorsión. De ahí que, la distorsión promedio asociada a la sucesión de cuantificadores vectoriales producidos por el Algoritmo de Lloyd debe converger en un número finito de pasos.

3.1.2 Alineación temporal dinámica (DTW *Dinamic Time Warpin*).

Los sistemas de reconocimiento de locutores basados en técnicas de DTW han sido los primeros que han alcanzado un nivel de fiabilidad suficientemente alto como para dar lugar al desarrollo de productos comerciales. Los sistemas de reconocimiento basados en DTW funcionan de la siguiente manera: primero se parametriza la señal de voz a reconocer; para ello se divide en pequeñas ventanas de análisis (unos 20 mseg), y sobre cada una de esas ventanas se realiza un proceso de análisis que extrae un conjunto de parámetros (que pueden ser acústicos o coeficientes espectrales). Ese conjunto o vector de parámetros se puede ver como un punto en un espacio n - dimensional. El conjunto de todas las ventanas de análisis se convertirá así en una secuencia de puntos en ese espacio, y esa secuencia de puntos es lo que se llama "patrón".

El sistema de reconocimiento dispone de un conjunto de patrones de "referencia" que se han calculado en la fase de entrenamiento, y que representan al conjunto de locutores que el sistema puede reconocer. De esta forma, una vez obtenido el "patrón", la tarea del sistema de reconocimiento consiste en compararlo con todos los patrones de referencia que el sistema tiene, calculando la "distancia" que lo separa de las referencias, y elegir como locutor reconocido aquel cuyo patrón de referencia de la menor distancia en la comparación.

Si bien DTW realiza la clasificación basado en la medición de distancias en el espacio de características de manera similar a VQ, utiliza el hecho de que durante el entrenamiento la frase dicha es la misma que en el test. DTW compara la secuencia de vectores de testeo $(\bar{x}_1; \ldots; \bar{x}_N)$ con la secuencia de entrenamiento $(x_1; \ldots; x_M)$, tomando en cuenta que ambas frases o palabras nunca son idénticas ya que los fonemas pueden pronunciarse de manera más larga o corta. Para esto, se encuentra una alineación temporal de las secuencias de entrenamiento y testeo la cual es óptima en el sentido de que no hay otra que de una distancia global menor (z) y que satisfaga ciertas restricciones.

$$z = \sum_{i=1}^{M} d\left(x_i, \bar{x}_{j(i)}\right) \qquad [3.8]$$

Donde los índices del vector de testeo *j(i)* son dados típicamente por el algoritmo DTW

Este método explica la variación, por tramos, de los parámetros que corresponden a la dinámica de las pronunciaciones y del tracto vocal. La figura 16 muestra que aparece una trayectoria de deformación cuando dos señales de energía (voz) se utilizan como rasgos de deformación. Si las señales fueran idénticas, la trayectoria de la deformación sería una línea diagonal y el alineamiento no tendría ningún efecto. En caso contrario el alineamiento procede en virtud de una distancia global menor.

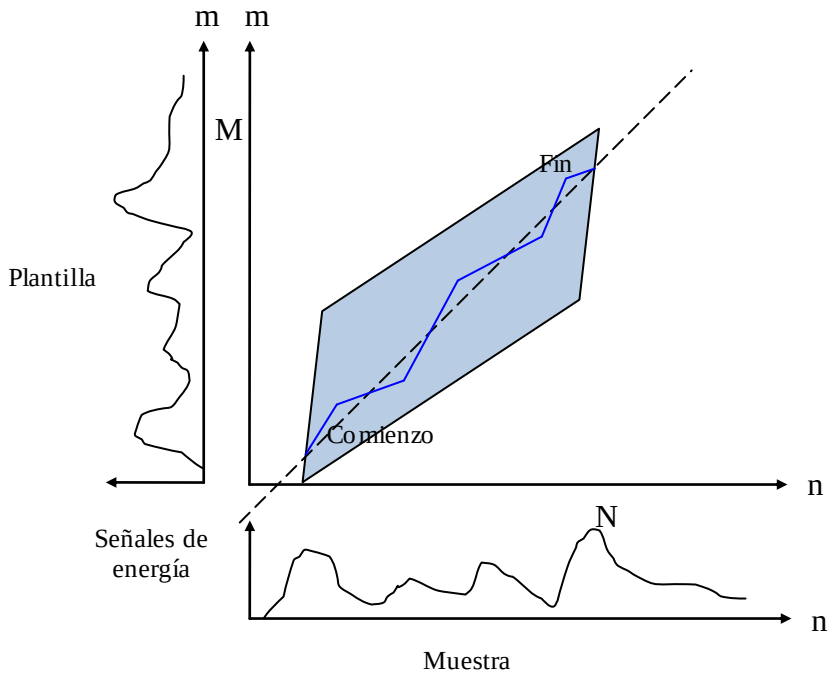


Figura 15 DTW de dos señales de energía.

3.1.3 Reconocimiento basado en modelos ocultos de markov (HMMs).

Los modelos ocultos de Markov (HMM) son modelos estadísticos que pueden representar procesos aleatorios paramétricos. Estos son el enfoque estocástico más popular y con mayor éxito en el ámbito de reconocimiento de voz.

Un modelo oculto de Markov está compuesto de dos elementos básicos:
Un proceso de Markov y un conjunto de distribuciones de probabilidad de salida. Los estados del proceso de Markov están "ocultos" pero son observables de una manera indirecta a partir de la secuencia de vectores con información espectral extraída de la señal de voz de entrada.

"Los HMMs se utilizan en reconocimiento de voz teniendo en cuenta dos hipótesis"[16]:

a) la voz se puede dividir en segmentos, estados, en los que la señal de voz se puede considerar estacionaria. Es decir, en la ventana de análisis la señal mantiene la estructura de principio a fin. Se asume que las transiciones entre segmentos contiguos son instantáneas.
b) La probabilidad de observación de que un vector de características se genere depende sólo del estado actual y no de símbolos anteriores. Esta es una suposición de Markov de primer orden, denominadas hipótesis de independencia.

Bases del reconocimiento de voz con HHMs. En un sistema de reconocimiento se asume disponible un conjunto de observaciones acústicas que representan una palabra, o en general una sentencia, a ser reconocida. ´Esta se puede denotar por O = $\{o_1, o_2, \ldots, o_t\}$, donde o_t es el vector de observación en el instante de tiempo t. Entonces, dado un vocabulario limitado de palabras, la tarea de reconocimiento se puede llevar a cabo a partir de la regla básica de decisión:

$$\hat{W} = arg \max_{w}\{P(W|O)\} \qquad [3.9]$$

Donde W representa la palabra, o cadena de palabras, bajo hipótesis. Utilizando la regla de Bayes se tiene que:

$$P(W|O) = \frac{P(W|O)P(W)}{P(O)} \qquad [3.10]$$

[16] Ninguna de estas hipótesis es cierta para la señal de voz. Sin embargo, hasta el momento, los HMM estándar son los que se utilizan en la mayoría de los reconocedores de voz actuales.

Por lo tanto, ya que P(O) se puede considerar constante para una entrada dada, la tarea de reconocimiento implica encontrar el modelo, o conjunto de modelos, que maximiza P(O|W)P(W) en lugar de P(W|O). Las probabilidades a priori P(W) se determinan normalmente a través de un modelo de lenguaje.

Para la tarea de reconocimiento se están utilizando, entonces, tres fuentes principales de información: el modelo acústico, el modelo de lenguaje y el léxico. El HMM, como modelo acústico, se necesita para determinar, junto con el modelo de lenguaje, la secuencia más probable de palabras (sentencia) dados unos datos observados de voz. Específicamente, dentro de este proceso el modelo acústico se necesita para dar la probabilidad de cada posible secuencia de palabras. Así, cada cadena de palabras se "mapea" al apropiado conjunto de modelos utilizando el léxico, y la tarea se reduce a encontrar P(O|M), donde M es el conjunto de HMMs asociado con la cadena de palabras W.

El léxico se utiliza para hacer corresponder cualquier forma abstracta que representan los modelos acústicos con las palabras presentes en el vocabulario y modelo de lenguaje. Para tareas de vocabularios pequeños el léxico suele ser muy simple con una correspondencia uno-a-uno del modelo acústico a la palabra. En tareas de grandes vocabularios, donde los modelos acústicos pueden representar unidades de tipo sub-palabra, el léxico dicta cómo se enlazan estas unidades de sub-palabras entre sí para formar palabras individuales. Por ejemplo, la palabra "tesis" se puede dividir como

$$\text{Tesis} = \{t\ e\ s\ i\ s\}$$

Donde los fonemas son las unidades de tipo sub-palabra

El modelo de lenguaje contiene la información sobre las secuencias de palabras permitidas. Puede incluir la probabilidad de tales secuencias. Este modelo reduce mucho la carga computacional y mejora la precisión del reconocedor limitando el espacio de búsqueda de palabras. Se han propuesto diversas aproximaciones para el diseño de los modelos de

lenguaje, por ejemplo gramáticas de pares de palabras, las cuales simplemente limitan el conjunto de palabras que pueden seguir a la palabra actual, o modelos de lenguaje estocásticos, tales como N-gramáticas, que asocian probabilidades con cada secuencia de palabras.

Asumiendo que la secuencia de observaciones correspondiente a una palabra ω_i se genera por un modelo oculto de Markov λ_i, para resolver el problema de reconocimiento se necesita calcular $P(O|\lambda_i)$, donde se asume que

$$P(O|\lambda_I) = P(O|\omega_I) \qquad [3.11]$$

Debido a la naturaleza oculta de la secuencia de estados en un modelo oculto de Markov, asumiendo una secuencia de estados S tal que $S = \{s_1, s_2,\ldots,s_T\}$, esta verosimilitud se puede calcular como

$$P(O|\lambda_I) = \sum_{s \in S} P(O|S,\lambda_I)\, P(S|\lambda_I) \qquad [3.12]$$

$$= \sum_{s \in S} a_{s_0 s_1} \prod_{t=2}^{T} a_{st\, st+1} b_{st}(O_t) \qquad [3.13]$$

Donde S es el conjunto de todas las posibles secuencias de estados de longitud T en el modelo λ_i y, S_o y S_{T+1} son los mencionados estados de entrada y salida del modelo.

Este cálculo, sin embargo, no es trivial, incluso para un número pequeño de estados (N) y tramas de tiempo (T), ya que en cada instante de tiempo se pueden alcanzar N−2 estados (sin contar una transición de salto al estado de salida), llevando a $(N - 2)^T$ posibles secuencias de estado, con aproximadamente 2T operaciones para cada secuencia.

El cálculo eficiente de la anterior verosimilitud se puede realizar utilizando el denominado procedimiento Forward-Backward, que es un algoritmo recursivo que se describirá en el entrenamiento. El paso forward de este algoritmo es suficiente para calcular $P(O|\lambda_i)$.

Especificaciones de los HMMs. Un modelo oculto de Markov es una máquina de estados finita probabilística, es decir, un conjunto de estados conectados unos a otros por arcos de transición, con probabilidades asociadas a cada arco. Figura 17. En cualquier instante de tiempo especificado se puede considerar que el sistema está en uno de los estados disponibles y que, a intervalos regulares de tiempo ocurre una transición a otro estado (o al mismo estado si este dispone de una transición a sí mismo) conforme a las probabilidades asociadas a los arcos de transición. Asociado a cada estado también existe una función de densidad de probabilidad que define la probabilidad de emitir un vector de observación una vez que se entra en dicho estado del HMM. Si una determinada señal de voz se parametriza en T vectores de observación, representados por $O = \{O_1, O_2,\ldots O_T\}$, entonces se puede calcular la verosimilitud de generar esta señal de voz utilizando una determinada secuencia de estados $S = \{s_1, s_2,\ldots,s_T\}$ dado el anterior HMM. Dicha verosimilitud está dada por el producto de la verosimilitud de que cada observación O_T sea generada por su estado asociado s_T, y la probabilidad de la secuencia de estados calculada a partir de las probabilidades de transición.

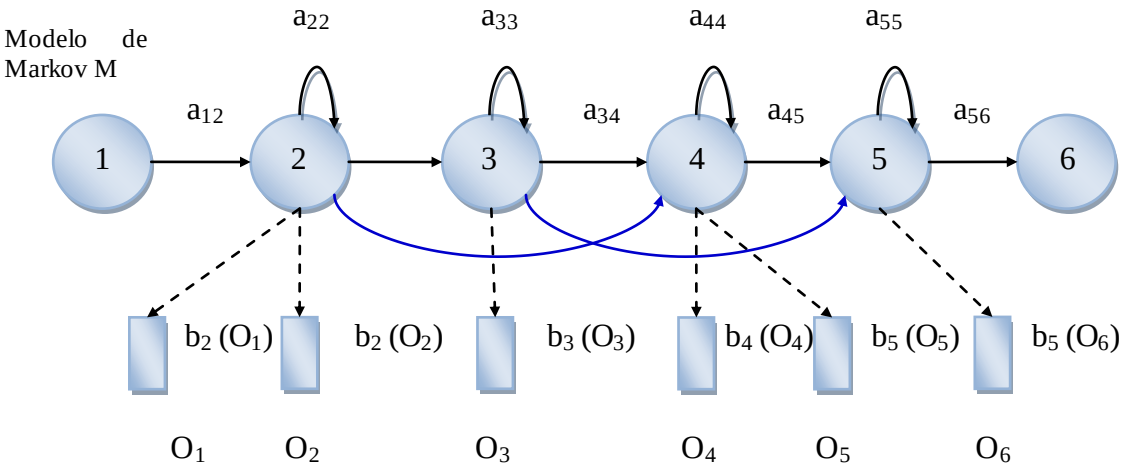


Figura 16 Generación de los modelos de Markov de izquierda a derecha (Bakis)

Arquitectura de los HMM. La arquitectura de un modelo oculto de Markov viene dada por el número de estados que lo componen y las transiciones permitidas entre dichos estados. Dependiendo de la aplicación a la que vayan a ser destinados los HMM interesará un tipo de arquitectura u otra.

Modelos de izquierda a derecha. En estos modelos los elementos de la matriz de probabilidades de transición A deben cumplir $A_{ij}= 0$, $j < i$

Es decir, si en un momento determinado el modelo está en el estado de índice i seguirá en dicho estado con probabilidad a_{ij}, o bien el modelo evolucionará a un estado con un índice j mayor que i con probabilidad a_{ij} . Esta configuración de la matriz A da lugar a modelos con arquitecturas como la que se muestra en la figura 17

Modelos ergódicos. El caso más genérico de modelos de Markov son los llamados modelos ergódicos. De manera estricta, un HMM ergódico será aquel en el cual se pueda evolucionar desde cualquier estado a cualquier otro en un número finito de transiciones, aunque el termino se suele utilizar habitualmente para referirse a modelos en los cuales todas las transiciones son posibles. Un ejemplo de un modelo ergódico se presenta en la figura 18.

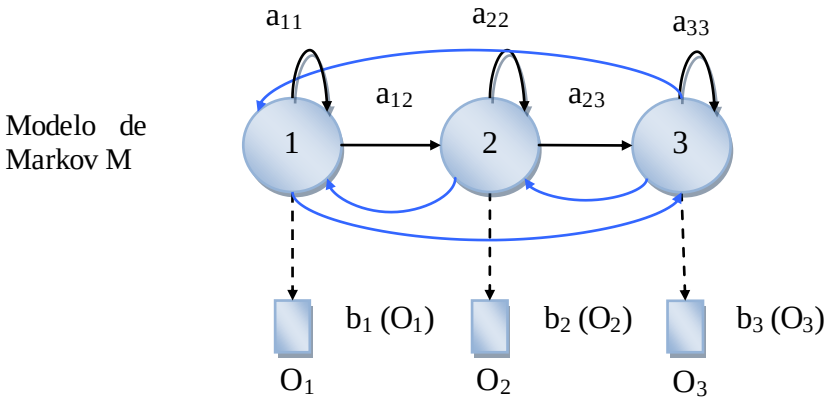


Figura 17 Ejemplo de HMM ergódico con tres estados.

Tipos de Modelos Ocultos de Markov. Un HMM se clasifica también en función de la matriz B de distribuciones de probabilidad de salida. Dependiendo de cómo sean los elementos de esta matriz los modelos se pueden clasificar en tres categorías:

Discretos: También denominados DHMM (*Discrete* HMM). En este caso, las observaciones son vectores de símbolos de un alfabeto finito con M elementos diferentes, cada uno denominado codeword, que se agrupan en el denominado codebook V = {v_1, v_2,…,v_M}. La distribución de probabilidad de un símbolo se define como B = {bj(k)} donde

$$b_j(k) = P\left(x = v_k | s_j\right), 1 \leq k \leq M$$

Es la distribución de símbolos en el estado j con j = 1, 2,…, N. En este caso basta especificar N y M, las observaciones de símbolos V y los conjuntos de medidas de probabilidad A y B que definen el modelo de Markov.

Continuos: También denominados CDHMM (*Continuous Density* HMM).

Se asume que las distribuciones son densidades de probabilidad de espacios de observación continuos. Se impone la restricción de que estas distribuciones presenten una determinada forma para obtener un número abordable de parámetros a estimar.
La forma más extendida de distribución es la de una "mezcla" de un conjunto de densidades base g de una familia G con una forma paramétrica simple.
Principalmente se utilizan densidades de base g ∈ G de tipo Gaussiano. Figura 20, para lo cual basta con definir la matriz de medias y la de covarianzas. Para la estimación de estos parámetros es necesario un gran número de vectores de entrenamiento, que aumenta de forma lineal con el número de modelos a entrenar.
Estas funciones permiten un modelado preciso y directo de los parámetros de la señal de voz y pueden llevar a mejores resultados de reconocimiento que el uso de densidades discretas.

Para un HMM de N estados la función de distribución de probabilidad se puede escribir como:

$$b_j(o_t) = \sum_{k=1}^{M} \omega_{jk} N(O_t; \mu_{jk}; \sum_{jk}), \quad 2 \leq j \leq N-1 \qquad [3.14]$$

Donde O_t representa un vector de observación, N una función normal multivariable caracterizada por unos valores medios μ_{jk} y unas varianzas $\sum_{jk}$, M el numero de mezclas por estado y ω_{jk} los pesos asociados con cada una de las funciones Gaussianas verificando que

$$\sum_{j=1}^{M} \omega_{jk} = 1 \qquad [3.15]$$

A pesar de que el uso de matrices de covarianza completas implica un menor número de Gaussianas que el uso de matrices de covarianza diagonales, para obtener prestaciones análogas, en la mayoría de las implementaciones, debido a la gran reducción en cálculo y memoria necesaria, se utilizan normalmente matrices de covarianza diagonales.

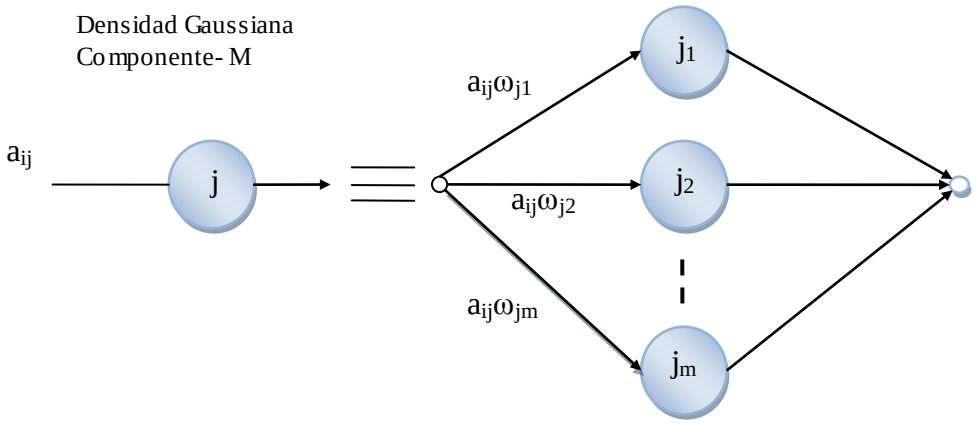


Figura 18 Representación de la densidad Gaussiana.

Semicontinuos: También denominados SCHMM (*Semicontinuous* HMM). Surgen de la necesidad de entrenar un gran número de modelos con bases de datos limitadas.
Los modelos semicontinuos, al igual que los continuos, se modelan a partir de un conjunto de "mezclas" de funciones de densidad de probabilidad Gaussiana.
La principal diferencia radica en que las funciones base son comunes a todos los modelos, como ocurre en el caso de los modelos discretos donde existe un codebook común a todos ellos. Por lo tanto, un modelo semicontinuo se define con los pesos asociados a cada una de las funciones base (Gaussianas).
La forma de la función bj(x) es idéntica a la del caso continuo, con la diferencia de que el conjunto de funciones normales es común a todos los modelos. Su expresión resulta ser:

$$b_j(x) = \sum_{k=1}^{L} \omega_{jk} N\left(x; \mu_{jk}; \sum_{jk}\right), \quad 1 \leq j \leq N-1 \qquad [3.16]$$

Donde L indica el número total de funciones normales en lugar del número de mezclas por estado.

Entrenamiento- reconocimiento usando HMM. Los modelos ocultos de Markov se caracterizan por tres problemas que hay que resolver para que resulten modelos útiles en aplicaciones reales.

Problema de evaluación. Dada una secuencia de observaciones y un modelo se busca cómo calcular la probabilidad de que la secuencia observada haya sido producida por dicho modelo.

Problema de estimación. Dada una secuencia de observaciones y un modelo, se busca cómo elegir una secuencia de estados que sea óptima en algún sentido.

Problema de entrenamiento. Dada una secuencia de observaciones de entrenamiento, se busca como obtener los parámetros de modelo de forma óptima.

Estos problemas se concretan en las dos fases, de entrenamiento y reconocimiento

El problema de entrenamiento. El problema de entrenamiento implica la estimación de los parámetros del modelo λ, dada la secuencia de observaciones $O = \{O_1, O_2 \dots O_T\}$ como datos de entrenamiento, tal que se maximice P(O |λ).

Típicamente el método utilizado para lograr esto es la estimación de máxima verosimilitud (ML – Máximum likelihood). Para el caso de datos incompletos (tales como la secuencia oculta de estados) la estimación ML se puede calcular utilizando el algoritmo Esperanza Maximización (EM).

El caso particular del algoritmo EM para los modelos HMMs se conoce como algoritmo de Baum- Welch. Desarrollado por Baum y sus colegas. El algoritmo EM es una aproximación iterativa para el cálculo de máxima verosimilitud que se utiliza para encontrar en cada paso una estimación del conjunto de parámetros λ, y luego intenta maximizar la verosimilitud de generar los datos de entrenamiento utilizando el modelo de tal modo que la nueva verosimilitud es mayor o igual a la previa. Al definir la secuencia de estados **s** como perteneciente a un espacio de secuencia S, que incluye todas las posibles secuencias de estados, la maximización de la anterior verosimilitud se puede realizar maximizando una función auxiliar dada por

$$Q(\lambda, \hat{\lambda}) \triangleq \sum_{s \in S} P(O, S|\lambda)\, log(O, S|\hat{\lambda}) \qquad [3.17]$$

La convergencia del anterior algoritmo fue aprobada en primer lugar por Baum.

La anterior función auxiliar se puede expandir y descomponer en funciones auxiliares separadas que se pueden maximizar de forma independiente, para obtener estimaciones para las probabilidades de transición entre estado, los pesos de las mezclas y los parámetros de las Gaussianas para el HMM propuesto. En concreto para la estimación de pesos y parámetros de Gaussianas se llega a las siguientes expresiones.

$$\mu = \frac{\sum_{t=1}^{T} L_{ik}(t) O_t}{\sum_{t=1}^{T} L_i(t)} \quad [3.18]$$

$$\hat{\Sigma} = \frac{\sum_{t=1}^{T} L_{ik}(t)(O_t - \mu_{ik})(O_t - \mu_{IK})^T}{\sum_{t=1}^{T} L_{ik}(t)} \quad [3.19]$$

$$\hat{w}_{IK} = \frac{\sum_{t=1}^{T} L_{ik}(k)}{\sum_{t=1}^{T} L_i(t)} \quad [3.20]$$

Donde L_{ik} *(t)* es la probabilidad a posteriori de estar en el estado i en el instante t. también se pueden obtener ecuaciones para la estimación de las probabilidades de transición de estado.

Para el cálculo de los anteriores parámetros se deben hallar las probabilidades a posteriori mencionadas. Ello se puede hacer de forma eficiente utilizando el algoritmo Forward-Backward que se describe a continuación.

Se define una variable "forward" $\alpha_i(t)$ como:

$$\alpha_i(t) = P(O_1, O_2, .. | O_t, s_t = i | \lambda) \quad [3.21]$$

Que es la verosimilitud de estar en el estado i en el instante t, habiendo visto los t primeros vectores de observación. Para su cálculo se puede seguir el siguiente procedimiento:

a) Inicialización. Dado que los estados de entrada y salida son no-emisores, las condiciones iníciales serán

$$\alpha_i(1) = 1$$

$$\alpha_i(1) = a_{1i} b_i(o_1), \qquad 2 \leq i \leq N-1 \qquad [3.22]$$

y también para la condición final

$$\alpha_N(T) = \sum_{j=2}^{N-1} \alpha_j(T) a_{jN} \qquad [3.23]$$

b) Inducción

$$\alpha_i(t) = \left[\sum_{j=2}^{N-1} \alpha_j(t-1) a_{ji}\right] b_i(o_t), \qquad 2 \leq i \leq N-1 \; y \; 2 \leq t \leq T \qquad [3.24]$$

c) Terminación

$$P(O|\lambda) = \alpha_N(T) \qquad [3.25]$$

La probabilidad forward se puede calcular de forma recursiva para todos los estados del HMM utilizando el paso de inducción en cualquier instante t, comenzando en t = 2 y continuando hacia t = T, donde se alcanza el paso de terminación.

La probabilidad "backward" se puede calcular de forma similar. Considerando la variable backward definida como:

$$\beta_i(t) = P(O_{t+1}, O_{t+2}, \ldots, O_T | S_t = i, \lambda) \qquad [3.26]$$

que es la verosimilitud de observar la secuencia desde t+1 hasta el final, estando en el estado i en el instante t. También se puede utilizar para su cálculo un procedimiento recursivo:

Inicialización

$$\beta_i(T) = a_{iN} \quad 2 \leq i \leq N-1 \qquad [3.27]$$

Inducción

$$\beta_1(1) = \sum_{j=2}^{N-1} a_{1j} b_j(o_1)\beta_j(1) \qquad [3.28]$$

De las anteriores definiciones se puede deducir que

$$\alpha_i(t)\beta_i(t) = P(O, S_i = i|\lambda) \qquad [3.29]$$

Como,

$$L_i(t) = P(s_t = i|O, \lambda) \qquad [3.30]$$

Entonces

$$L_i(t) = \frac{1}{P(O|\lambda)}\alpha_i(t)\beta_i(t) \qquad [3.31]$$

De forma análoga se puede obtener $L_{ik}(t)$ para el caso de utilizar "mezclas" Gaussianas.

El problema de reconocimiento. En la práctica es preferible basar el reconocimiento en la secuencia de estados de mayor probabilidad que genera un conjunto dado de observaciones, ya que esto se generaliza de forma sencilla a la tarea de reconocimiento de voz conectada, mientras que el uso de de la posibilidad total no lo hace.

El problema es encontrar

$$\arg\max_I \{P(w_i|O)\} \qquad [3.32]$$

Donde de acuerdo a Bayes:

$$P(w_i|O) = \frac{P(0|w_i)P(w_i)}{P(O)} \qquad 3.33$$

Asumiendo que la secuencia de observaciones correspondiente a una palabra w_i se genera por un modelo oculto de Markov λ_i, para resolver el problema de reconocimiento se necesita calcular $P(0|\lambda_i)$. En este caso se asume que:

$$P(O|w_i) = P(O|\lambda_i) \qquad [3.34]$$

$$= \arg\max_x \prod_{xi} b_{xi}(O_i) \prod_{t=2}^{T} a_{xt-1} x_t b_{xt}(O_t) \qquad [3.35]$$

El cálculo de la anterior verosimilitud se puede realizar utilizando el denominado procedimiento fordward – Backward, que es un algoritmo recursivo. El paso forward es suficiente para calcular $P(O|\lambda_i)$.

Dado un modelo λ, se define un parámetro $\delta_i(t)$como la verosimilitud máxima de observar los vectores O_1 a O_t y estar en el estado i en el instante t. Esta verosimilitud parcial se puede calcular de forma eficiente utilizando la siguiente recursión

$$\delta_i(t) = \max_j \{\delta_i(t-1)a_{ij}\} b_i(O_t) \qquad [3.36]$$

Las condiciones iníciales en este caso son:

$$\delta_1(1) = 1$$

$$\delta_1(1) = a_{1i} b_i(O_i) \;\; 2 \leq i \leq N-1 \qquad [3.37]$$

El paso de terminación, la máxima verosimilitud es:

$$\delta N(t) = \max_j \{\delta_j(t) a_j N\} \qquad [3.38]$$

El mejor camino se puede obtener calculando la anterior mayor verosimilitud en cada instante de tiempo y manteniendo el camino de los argumentos que lo maximizan.

La anterior recursión forma la base del denominado algoritmo de Viterbi

3.1.4 Modelo de mezclas Gaussianas (Gaussian mixture models).

Para implementar el test de hipótesis mencionado en **(verificación del locutor)** se debe elegir la función de probabilidad $f(\lambda / x)$ que de aquí en adelante llamaremos $P(\lambda / x)$. Para las aplicaciones *dependientes del texto* HMM tiene una buena performance pero para aplicaciones independientes del texto GMM ha probado ser el más exitoso.

GMM es ampliamente utilizado para el modelado de la voz a partir de los vectores de características (**MFCC** u otros) adquiridos de cada locutor. Una vez obtenida cierta cantidad de estos vectores por cada locutor, se crea un modelo probabilístico que lo representa de forma singular.
Dado un vector de características: $\vec{x}$, la mezcla de densidades Gaussianas está dada por:

$$P(\vec{x}/\lambda) = \sum_{i=1}^{M} w_i b_i(\vec{x}) \qquad [3.39]$$

Que no es más que la combinación lineal ponderada de M densidades Gaussianas b_i, y que representa la probabilidad de observar un determinado vector de características $\vec{x}$ de cierto locutor λ, en donde:

- $\vec{x}$ Es el vector de dimensión D a observar.

- w_i son los pesos de cada componente Gaussiana y cumplen $\sum_{i=1}^{M} w_i = 1$
- b_i ($\vec{x}$) son las densidades Gaussianas D – dimensionales, cada una con la forma:

$$b_i(\vec{x}) = \frac{1}{(2\pi)^{D/2}|\sum_I|^{1/2}} e^{-(1/2)(\vec{x}-\mu_i)^t \Sigma_t^{-1}(\vec{x}-\mu_i)} \quad con\ i = \cdots M \qquad [3.40]$$

- con $\vec{\pi}_i$ y Σi los vectores de medias y matrices de covarianza respectivamente.
- M es el orden del modelo o número de Gaussianas que tiene el modelo.

De esta forma cada locutor será representado por un modelo de muestras Gaussianas λ cuyos parámetros son:$\{w_i, \mu_i, \Sigma_i\}$ con i =1...M

Se debe ser precavido a la hora de elegir la cantidad M de mezclas Gaussianas con la cual se va a trabajar. Elegir un número elevado puede provocar que el modelo hallado sobre ajuste demasiado a los datos extraídos (*overfitting*). Por otro lado elegir un M pequeño puede llevar a que el modelo no sea lo suficientemente diferente a los demás modelos y no se pueda reconocer adecuadamente al locutor en cuestión. Experimentalmente se encuentra que generalmente dieciséis Gaussianas es un número apropiado.
El modelo general consta de matrices de covarianza Σi completas, pero lo más común en la bibliografía consultada es emplear modelos en los cuales las matrices de covarianza son diagonales. Esto reduce el número de parámetros que deben ser optimizados y además simplifica enormemente los cálculos a realizar. Sin embargo, esta limitación sobre las matrices de covarianza reduce las capacidades de modelado e incluso puede que se necesite incrementar el número de componentes empleadas.

Proceso de entrenamiento. A partir de una colección de vectores X = $\{x_1 \ldots\ldots\ldots\ldots, x_t\}$ de entrenamiento de una persona, se estiman los parámetros del modelo usando el algoritmo EM (estimación - maximización)

Partiendo de un modelo inicial, el algoritmo EM refina iterativamente el modelo GMM incrementando de manera monótona su verosimilitud. Esto es, en la k - ésima iteración se encuentra el modelo $\lambda(k)$ y se cumple: $P(X / \lambda^{(k)}) > P(X / \lambda^{(k-1)})$. Este es el nuevo modelo inicial para repetir el proceso hasta llegar a un nivel de convergencia predeterminado.

En general el conjunto de vectores de características es muy grande, y por tanto, los valores de *P* (...) son a menudo muy chicos. Por esta razón es común calcular el logaritmo de las verosimilitudes que viene dado por:

$$LogP\left(X/\lambda\right) = \frac{1}{T}\sum_{t=1}^{T} logP(\vec{x}_i/\lambda) \qquad [3.41]$$

A este valor lo llamaremos Logl (Log - *Likelihood*) y es la medida que nos dice que tan probable es que los vectores X pertenezcan al modelo λ

La condición para detener la iteración puede ser: $LogP(X / \lambda^{(k)}) - LogP(X / \lambda^{(k-1)}) < \varepsilon$, o se puede imponer un número máximo de iteraciones.

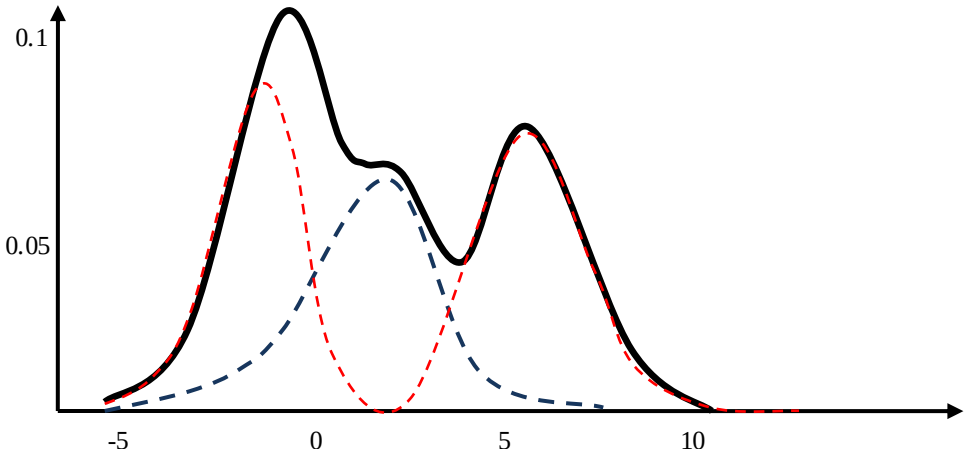


Figura 19 Modelo GMM (línea sólida) con tres mezclas Gaussianas (línea punteada)

Proceso de testeo. El *logl* es comúnmente usado para medir que tan bien un modelo se ajusta a los datos experimentales. Veamos a continuación como es que se lleva a cabo esta medida.

La *Identificación* de locutores se asocia con un problema de cercanía, de qué modelo se acerca más a los datos de entrada El sistema supone que los vectores $\vec{x}$ de entrada pertenecen a un locutor que ya tiene su modelo correspondiente λ creado en la base de datos. Simplemente se deben evaluar los *logl* para cada modelo, aquel con mayor argumento es el que tiene mayor probabilidad de que los vectores de entrada pertenezcan a ese modelo. Cabe acotar que el *logl* con mayor argumento será el de $P(x / \lambda)$ más cercana a uno.

En *Verificación*, en cambio, se debe decidir si los vectores $\overrightarrow{x}$ de entrada de un locutor desconocido, pertenecen o no a un locutor buscado, cuyo modelo llamaremos: λ_i Aquí la decisión se debe tomar sin tener en cuenta los otros modelos existentes en la base de datos.
Para esto se crea el modelo universal o mundial (**UBM**) λ_M a partir de voces de diferentes personas que pueden o no estar incluidas en la base de datos.
El sistema acepta o rechaza la hipótesis de que los vectores x_i son de la persona en cuestión. Para eso se deben evaluar los *logl* para el modelo de la persona buscada λ_i y para el modelo universal: λ_M compararlos y decidir de acuerdo a:

$$\Lambda(X) = LogP(X/\lambda_I) - LogP(X/\lambda_M) \;\; Si \begin{Bmatrix} \Lambda(X) > \Theta \rightarrow Aceptación \\ \Lambda(X) > \Theta \rightarrow Rechazo \end{Bmatrix} \qquad [3.42]$$

Si $\Lambda(X)$ es mayor que cero, significa que X tiene más probabilidad de pertenecer al modelo del locutor buscado que al modelo Universal, lo contrario sucede si $\Lambda(X)$ es menor que cero. A partir de esta deducción se podría fijar el umbral Θ en cero, pero experimentalmente se comprueba que esta no es una conclusión del todo acertada. Si bien el umbral es muy cercano a cero no es exactamente cero, más aún, puede ser negativo.

3.1.5 Arboles de decisión.

Los árboles de decisión son clasificadores muy conocidos en el campo de las estadísticas. Un árbol de decisión representa una colección de reglas, qué se organizan jerárquicamente, para conformar una estructura de decisión. Entre los árboles de decisión más populares se incluyen: el C4 e ID3 de Quinlan, CART de Breiman, y El árbol de decisión Bayesiano de Buntine.

En un árbol de decisión, cada nodo del árbol representa un punto de decisión, y cada terminal una clase (hoja). Las hojas representan las particiones exclusivas de los datos de entrada. Un vector de prueba se evaluará en cada nodo y dirigido a uno de dos nodos subsecuentes basado en una decisión. Este proceso de decisión continúa hasta que el vector de prueba venga a una hoja, en ese punto, se asignará la etiqueta de la clase de esa hoja.

En general, los árboles de decisión sólo consideran un elemento al tiempo, al tomar una decisión. Esto restringe la partición del espacio del rasgo para usar discriminantes que son perpendiculares al los ejes del rasgo. Ésta puede ser una severa limitación para los problemas que requieren más flexibilidad en el posicionamiento del discriminante, es decir, un discriminante diagonal. Los árboles de decisión ID3 y C4 están sujetos a este constreñimiento. El árbol de decisión de CART permite los discriminantes que son una función de todos los elementos del rasgo. Sin embargo, el algoritmo usado para encontrar estos discriminantes es heurístico e involucra una búsqueda exhaustiva; por consiguiente, es una alternativa poco atractiva computacionalmente.

La arquitectura de un árbol de decisión se encuentra por la inducción de la regla y puede determinarse recursivamente como sigue. Si todos los datos en un nodo pertenecen a una clase, entonces, designe este nodo como una hoja, y asigne a él la etiqueta de esa clase. Por otra parte, seleccione el rasgo que proporcionará la mayor ganancia de información por los datos en la subsecuente división. Una decisión se lleva a cabo basado en este rasgo y los datos son divididos acordadamente. Cada partición de los datos se envía a un nuevo nodo, y el algoritmo anterior se repite. Un árbol de decisión extendido a partir de este algoritmo clasificará completamente los datos de entrenamiento. Sin embargo, es bien sabido que los clasificadores que tienen un 100% de actuación sobre los datos de entrenamiento son

típicamente sobreentrenados y tienen rendimiento inferior con los datos de prueba. Este problema puede reducirse recortando el árbol[17]. Recortarlo consiste en quitar subárboles que contribuyeron a un excesivo crecimiento del árbol durante la clasificación de unos cuantos datos de entrenamiento. Aquí, la ejecución de los datos de entrenamiento se reordenará para mejorar la ejecución de los datos de comprobación. Los árboles de decisión tienen algunas ventajas sobre los MLPs. que incluyan arquitecturas de la misa organización y que no tienen que ser especificado a priori como con MLP. Por otro lado, MLP es ventajoso sobre los árboles de decisión en que la división espacial del rasgo no se restringe a un discriminante que es perpendicular a los ejes del rasgo. La actuación de los árboles de decisión y MLP se ha comparado y evaluado sobre la tarea de procesamiento del discurso, como el reconocimiento vocal. Lo que se quiere destacar aquí es que los MLPs. (con la selección de una arquitectura apropiada) pueden comportarse como arboles de decisión.

Un árbol de decisión puede usarse para el reconocimiento de locutor como sigue. Primero, los vectores de características se obtienen a partir de los datos de entrenamiento con todos los locutores. Estos datos se etiquetan entonces de una misma manera. Como se explico al comienzo. Un árbol de decisión binario es entrenado para cada locutor. Las hojas del árbol de decisión binario contendrán la etiqueta de la clase, dónde un uno corresponderá al locutor verdadero y un cero corresponderá al locutor rechazado. Además de las etiquetas de la clase, las hojas de decisión también requieren de un árbol que contenga una probabilidad asociado con la etiqueta. Para la identificación del locutor, todo el vector de características de la pronunciación se aplica a cada árbol de decisión. La medida de verosimilitud de cada locutor basada en la probabilidad del árbol de decisión es usada para determinar el locutor.

[17] En esta parte es valido utilizar el enfoque llamado *reduced-error pruning*, consiste en considerar cada nodo del ´árbol como candidato a ser podado. En este caso, la poda consiste en eliminar todo el sub árbol que tiene como raíz el nodo en cuestión, convirtiéndolo así en una hoja, cuya clase corresponde a valor más común de los ejemplares asociados a ese nodo.

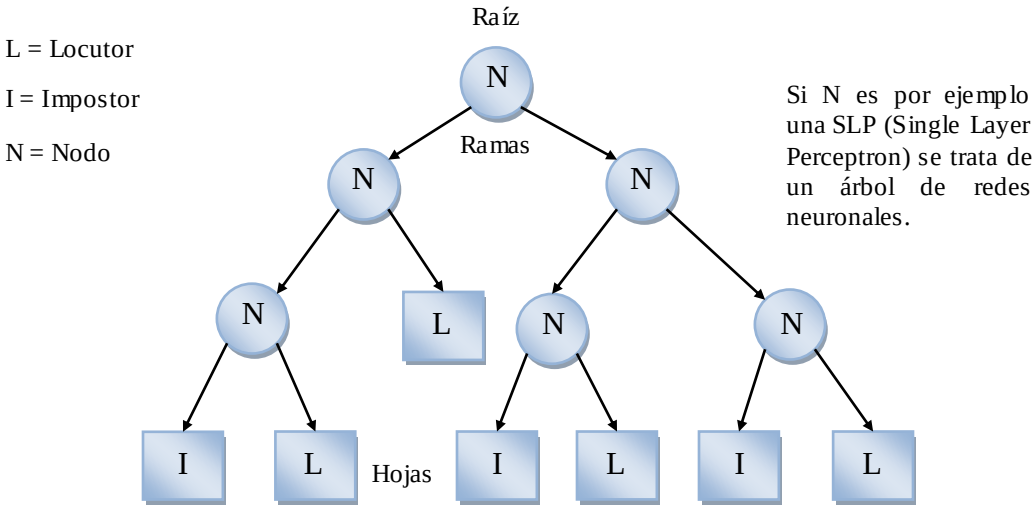


Figura 20 Árbol de decisión binaria

3.2 Las redes neuronales en el reconocimiento de locutor

Son numerosos los trabajos realizados empleando distintos modelos de redes neuronales y/o comparando el rendimiento de estas entre sí, ya que el ámbito de aplicación, con respecto a las distintas opciones de estudio que presenta el reconocimiento del locutor, es total, es decir, han sido utilizadas tanto en verificación, como en identificación, pudiendo ser estas tanto dependientes, como independientes de texto. Generalmente relacionadas con la etapa de clasificación, el papel jugado dentro de los sistemas de reconocimiento del locutor es más amplio, y aunque en menor medida, también son utilizadas en la etapa de extracción de características de la señal de voz. En estos casos, han sido usadas para realizar transformaciones no lineales sobre el vector de características, buscando espacios de representación que mejoren la eficacia de la etapa de clasificación.

Centrándonos en esta etapa, indicar que las estrategias de clasificación en las que intervienen, de una u otra forma, redes neuronales son múltiples. Lo más clásico es utilizar para la clasificación de la señal de voz un determinado tipo de red neuronal: MLP, RBF.

Son numerosos los estudios basados en este tipo de sistemas, pudiendo diferenciar, fundamentalmente, dos enfoques distintos: en el más habitual las redes tienen un papel discriminante, es decir, su salida proporciona directamente información acerca de la pertenencia o no de una muestra de voz a un determinado hablante. El otro enfoque consiste en usar las redes como predictores, pudiéndosele considerar como una extensión no lineal de los modelos autoregresivos: la entrada a la red es la secuencia de vectores de características $x_{t-i}, \ldots, x_t$, siendo la salida la predicción del vector x_{t+1}, la información acerca del propietario de la señal de voz se obtiene de la comparación entre el vector predicción y el vector real.

Casi simultáneo con el estudio de los sistemas que se acabaron de presentar, aparecen otros que buscan mejorar la etapa de clasificación, usando sistemas más complejos. Una gran línea de trabajo en este sentido, es la basada en el uso de modelos híbridos HMM-ANN, de forma que el resultado es un clasificador que usa lo mejor de cada uno para corregir las limitaciones de ambos. Dentro de esta línea podemos incluir, aunque no sean exactamente sistemas híbridos, aquellos trabajos que tras obtener mediante un modelo HMM la probabilidad a priori P(x/l): dado el modelo l, que la muestra de voz x pertenezca al modelo, utilizan ésta como entrada a una red neuronal para obtener la probabilidad a posteriori P(l/x): dada la muestra x, que pertenezca al modelo l, siendo esta probabilidad la salida final del sistema. La otra gran línea de trabajo es la basada en el uso de sistemas modulares, de forma que la decisión se basa en integrar la salida de múltiples clasificadores, pudiendo ser estos de igual naturaleza (por ej., mezcla de expertos), o de distinta naturaleza (por ej., reconocimiento de personas mediante la integración de reconocedores basados en distintos parámetros biométricos o integración de los resultados de distintos clasificadores ante una misma muestra de entrada). En este caso de sistemas modulares, la etapa de clasificación la podemos dividir en dos partes: respuesta individual de cada clasificador, integración de todas estas para lograr una salida única del sistema; pues bien, las ANNs pueden intervenir tanto en una tarea, como en la otra.

Como se puede observar, el campo de estudio es amplio. Buscando una mayor claridad en la presentación, es pertinente dividir las aplicaciones de las redes neuronales en la tarea del reconocimiento de un locutor (clasificación, reconocimiento):

- **Redes neuronales en la etapa de clasificación: sistemas simples**. Las redes forman parte de la etapa de clasificación. Aquí se analizan sistemas en los que la clasificación se realiza mediante un único módulo, bien compuesto por un determinado tipo de red neuronal, o bien por un modelo híbrido HMM-ANN.
- **Redes neuronales en la etapa de clasificación: sistemas modulares**. Nuevamente las redes forman parte de la etapa de clasificación. Incluiremos en este apartado referencias tanto al papel jugado en la parte de clasificación, como en el de integración.

3.3 Redes neuronales en la etapa de clasificación: sistemas simples

Redes neuronales no recurrentes. Cuatro son los modelos principalmente usados: Perceptron Multicapa (MLP), red neuronal de Funciones de Base Radial (RBFNN), Learning Vector Quantization (LVQ) y Mapa autoorganizado de Kohonen (SOM), aunque el rendimiento y aplicación de este último es menor. Vamos a estudiar cada uno por separado, incluyendo las referencias oportunas a los estudios comparativos con el resto.

3.3.1 Perceptron multicapa (MLP).

Aunque en los estudios realizados a tal fin, el rendimiento comparativo con otros modelos, por ejemplo RBF, no haya sido superior, si se puede considerar el modelo más frecuentemente utilizado y de manera más versátil. Dada esta amplitud de estudio, vamos a dividir la utilización del MLP en el reconocimiento del locutor en tres grandes grupos, atendiendo a su función:

- **Como clasificador**. La salida de la red es directamente una estimación de la probabilidad de pertenencia de la entrada a una determinada clase.
- **Como predictor**. Dada una secuencia de vectores de características correspondientes a los instantes de tiempo $t_n, t_{n-1}, \ldots, t_1$ la red trata de predecir el valor del vector de características en el instante siguiente, o sea, t.
- **Como transformada**. Se busca una transformación sobre los datos de entrada tal que sea una característica del hablante.

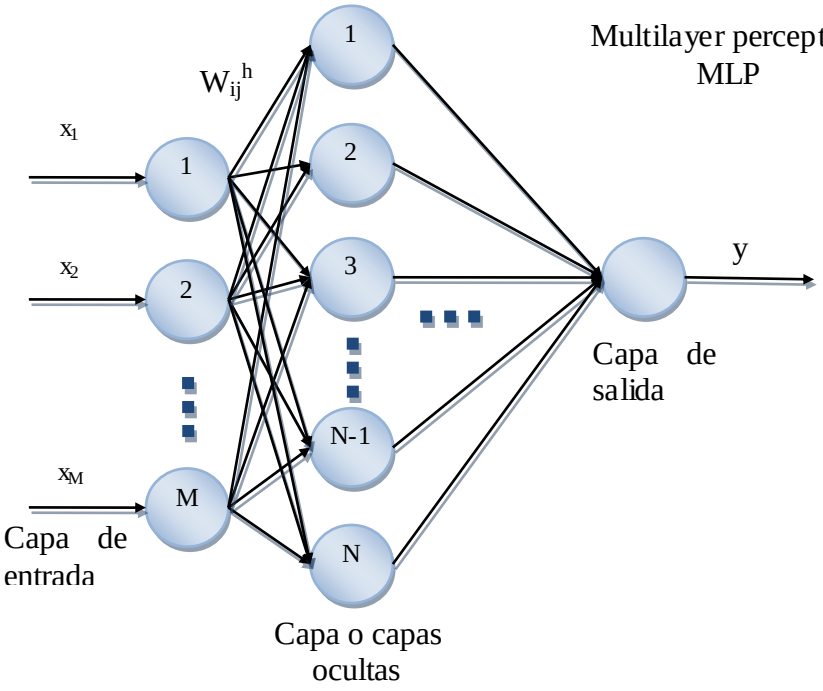


Figura 21 Esquema general de una red percepton multicapa

MLP como clasificador. Una de las tareas en las que ha demostrado un alto rendimiento es como generador de *funciones discriminantes:* la salida se puede considerar como la probabilidad de que la muestra de voz pertenezca o no a una determinada clase (hablante, en nuestro caso). Aunque conceptualmente el papel de la red es similar en todos ellos, vamos a ver algunos sistemas propuestos por diversos autores. En sus primeros trabajos,

Oglesby y Mason[18] abordan la tarea de identificación dependiente de texto, basada en una secuencia de 10 dígitos. La población de hablantes era de 10, entrenando para cada uno una red neuronal distinta, mediante una única muestra de la secuencia de 10 dígitos. Una vez concluido el entrenamiento la asignación muestra de voz-hablante se realiza siguiendo el criterio de red con valor de salida más alto. El sistema se probó con 4 muestras de la secuencia de 10 dígitos por hablante (distintas, obviamente, a las de entrenamiento). Se experimento con distintas arquitecturas, obteniendo los mejores resultados: 8% de error en la identificación, con una de tres capas, siendo el tamaño de la oculta de 128 neuronas. Las características usadas fueron 10 coeficientes cepstrales, obtenidos a partir de los coeficientes de predicción lineal (LPCC). Los autores compararon el sistema propuesto con otro basado en VQ, obteniendo para éste un rendimiento similar al mostrado cuando el número de centroides es de 64.

Vivaracho (1994) une el poder discriminante del MLP, con la compactación que, en cuanto al número de vectores de características a procesar, supone el uso de parámetros estadísticos para representar la muestra de voz. Concretamente, se utiliza la media por banda de frecuencia, calculadas estas mediante escala de Mel sobre el espectro de frecuencia, como entrada a la red. La perdida de información temporal que supone el promedio, se ve compensada al utilizar muestras de voz de corta duración, como son los dígitos, para el reconocimiento. Se realizaron pruebas de verificación tanto dependiente, como independiente de texto, sobre una población de 21 hablantes. En verificación dependiente de texto, se entrena una red por dígito y hablante, usando 5 muestras del dígito para tal fin. La red es probada con las otras 5 muestras del dígito de que se dispone. En verificación independiente del texto, se entrena una red por hablante, usando para ello las 10 muestras de 8 de los 10 dígitos. La red se prueba con las 10 muestras de los 2 dígitos que no fueron usados en entrenamiento. Tanto en un caso como en otro, la red es entrenada para dar salida 1 si la muestra pertenece al hablante, y 0 en caso contrario, usando como criterio de convergencia que la diferencia entre la salida deseada y la real para todas las

[18] Oglesby J. y Mason J. S. (1990). "Optimization of Neural Models for Speaker Identification". Proc. IEEE ICASSP, S5-1, pp. 261-264, Albuquerque, New Mexico (USA), 1990.

muestras de entrenamiento está por debajo de un determinado valor umbral, fijado a 0.01. se registra, que el caso más favorable: red con tres capas, 32 neuronas en la de entrada y 4 en la oculta, comparando estos con los de un sistema basado en HMMs (trabajo realizado Silva, Vivaracho, Alonso y Cardeñoso en 1998). Un dato a tener en cuenta en la comparación es que el número de parámetros de la red es muy inferior al necesario en HMM.

Hennebert y Petrovska[19], abordan la tarea de verificación del locutor mediante texto pedido (text prompted). Como primera aproximación, en su trabajo estudian el comportamiento de un sistema de verificación dependiente de texto basado en fonemas individuales, de forma que esto les permite analizar el rendimiento de distintos grupos de estos en la tarea propuesta. Entrena una red por hablante y fonema con una arquitectura, nuevamente, de tres capas, con 20 neuronas en la oculta y, a diferencia de los otros dos trabajos presentados, 2 en la de salida: una para la clase cliente (hablante para el que se entrena la red) y la otra para la clase impostor (todo hablante distinto al cliente): en entrenamiento las salidas deseadas se fijan a 1 y 0 respectivamente si la muestra pertenece al hablante cliente, y a 0 y 1 en caso contrario. El coeficiente de aprendizaje h es actualizado tras cada época de entrenamiento, según el siguiente criterio:

- $\eta_{i+1} = \eta_i/2$ *si el error medido sobre un conjunto de validación (independiente del de entrenamiento) se ha incrementado en la época i con respecto al obtenido en la época i-1.*
- $\eta_{i+1} = \eta_i$ *si el error anteriormente indicado ha decrecido*

El hecho de que se incremente el error indica que la red se está sobreentrenando, para evitar esto la actualización de pesos de la red es descartada si ocurre la situación planteada en el primer punto (anterior). El entrenamiento se finaliza cuando η caiga por debajo de un valor

[19]Hennebert J. y Delacretaz D. P. (1998). "Phoneme Based Text Prompted Speaker Verification with Multi Layer Perceptrons". Proc. RLA2C Workshop on Speaker Recognition and Its Commercial and Forensic Applications, pp.55-58, Avignon (Francia), 1998.

umbral prefijado. En prueba la decisión (hablante aceptado/rechazado) se toma atendiendo al criterio de la clase de la neurona con el valor de salida mayor. Las muestras de voz utilizadas en los experimentos pertenecen a la "HER Swiss German telephone speech database", y fueron tomadas vía teléfono en una única sesión. Una vez aislados los fonemas, las muestras correspondientes a cada uno de estos son divididas en ventanas: porciones de 30 ms. de señal, extrayendo de cada una un vector de características compuesto por 12 LPCC. Para probar la influencia de la información contextual en el reconocimiento del locutor, experimenta el sistema con y sin esta información, o sea:

- Sin información contextual: tanto en entrenamiento como en prueba la entrada a la red es el vector de características extraído de cada una de las ventanas en que ha sido dividida la señal de voz correspondiente.
- Con información contextual: ahora la entrada a la red incluye además del vector de características de cada ventana, los de j ventanas anteriores y posteriores. Los valores de j probados fueron 1,2 y 3.

El rendimiento del sistema es mayor cuando se incluye información contextual, obteniéndose la mejor relación resultados/tamaño de la red para j=1. En este caso, el valor de la tasa de equierror varía entre el 9.7%, en el mejor de los casos, para fonemas nasales y el 22.5%, en el peor de los casos, para sonidos líquidos.

Para concluir con trabajos que usan MLP como funciones discriminantes, referenciar el realizado por Labanova y Raev[20]. En su estudio presentan un enfoque original en cuanto a los datos de entrada a la red, proporcionando a la salida de esta un significado también original. Su objetivo es la verificación dependiente de texto vía teléfono. Como parámetros representativos del locutor parten de los formantes, pero no de los valores estáticos que estos adquieren en cada una de las ventanas en que se divide la señal (concretamente, de un

[20] Labanova M. A. y Raev A. N. (1998). "Speaker Verification Accounting the Formant Behaviour and Phonetic Representation of Enrolled Speech". Proc. RLA2C Workshop on Speaker Recognition and Its Commercial and Forensic Applications, pp.37-39, Avignon (Francia), 1998.

tamaño de 256 muestras, con un solapamiento de 40 entre dos consecutivas), sino de la evolución de estos en porciones de como máximo 6 ventanas, con la característica añadida de que esa evolución debe ser suave (este concepto es implantado estableciendo un umbral en la diferencia entre un formante de una ventana y la siguiente); a las zonas así obtenidas las denominan CFR (*Chain of the Frame*). El reconocimiento de una muestra de voz de origen desconocido se hace como sigue: se realiza un alineamiento entre CFRs de la muestra de entrenamiento y CFRs de la desconocida, sobre estos se establecen una serie de medidas (concretamente 8 distintas), cuyos valores serán las entradas a la red. La salida de esta será, por los tanto, la probabilidad de que las dos muestras confronten, o sea, la probabilidad de que los hablantes comparados sean el mismo o no. Dado que varias de las entradas a la red se pueden considerar como medidas de similitud entre dos vectores, el papel de está también puede ser considerado como de integrador no lineal de todas estas medidas para obtener una respuesta única, algo similar a lo que veremos en el siguiente apartado. La arquitectura seleccionada vuelve a ser de tres capas, con 16 neuronas en la oculta y 1 en la de salida. Se realizaron pruebas con 25 hablantes clientes y 100 impostores, obteniendo una tasa de equierror de 2.9 %, para claves basadas en sentencias de duración de 3 a 5 segundos.

MLP como predictor. La eficacia del MLP como discriminante ha sido de sobra demostrada, sin embargo, es conveniente recordar que para que las superficies de separación entre clases puedan ser perfectamente definidas, es necesario presentarle a la red, en un entrenamiento, un conjunto de ejemplos de cada una lo suficientemente representativo. Particularizando al caso del reconocimiento del locutor, esto, en principio, es totalmente factible en identificación, donde tenemos muestras de todas las clases a diferenciar. Sin embargo, la tarea de verificación es diferente: hemos de diferenciar entre un hablante (llamémosle cliente) y el resto (impostores), donde el "resto" es cualquier hablante. Esto plantea dos cuestiones a tener en cuenta: primero, la de poseer un conjunto de muestras de hablantes impostores, y, segundo, como escoger éste para que sea suficientemente representativo. Los enfoques que veremos a continuación obvian este doble

problema: incluso en verificación, la red es entrenada con tan solo las muestras del hablante cliente.

Aparte de su capacidad como clasificadores, algunos autores han explotado el potencial del MLP como generador de funciones, en este caso para la ***predicción*** de series temporales. Los sistemas presentados pueden ser considerados como una extensión no lineal de los modelos autorregresivos lineales (AR) estudiados por autores como Montacié y Le Floch (1992) o Bimbot y otros (1992).

Uno de los principales autores que han profundizado en este campo ha sido Hattori[21]. En el trabajo presentado en ICASSP´92, estudia 2 enfoques distintos para un sistema de identificación independiente del texto. En el primero de ellos entrena una red por hablante de forma que sea capaz de predecir una ventana de voz a partir de las dos anteriores. Durante la fase de prueba se mide la diferencia entre los valores pronosticados y los reales, asignando la muestra de voz a aquella red (hablante, por tanto) con un error de predicción menor. El enfoque presentado está suponiendo que el proceso que se intenta modelar es estacionario, pero en realidad no lo es. Para tratar este problema se plantea un sistema con M estados, donde cada estado es un modelo predictivo (MLP) diferente, especializado en una determinada parte de la muestra de voz del locutor. Este sistema puede ser considerado como un HMM, con una probabilidad de transición fija entre estados, y donde cada uno de estos ha sido sustituido por un MLP; estamos hablando de un sistema híbrido HMM-MLP, que será tratado en el apartado correspondiente. El mismo autor presentó en el congreso especializado en el reconocimiento del locutor, celebrado en Martigny (Suiza) (1994), un sistema para la verificación del locutor independiente del texto, que es una modificación del de un solo estado anteriormente descrito (llamado sistema base). La modificación afecta a la salida de la red, a la que aplica lo que denomina una normalización del error de predicción. Para la realización de esta normalización parte de la siguiente hipótesis: el error de predicción aumenta debido a dos factores, uno son las propias limitaciones de la red en su proceso de aprendizaje, le llama factor inherente, y el otro son aquellas características

[21] Hattori H. (1992). "Text Independent Speaker Recognition Using Neural Networks". Proc. IEEE ICASSP, Vol. 2, pp. 153-156, San Francisco (USA), 1992.

que no pueden ser aprendidas de los datos de entrenamiento, denominándole factor no aprendido. Ejemplos de factores inherentes son el ruido no causal: la red asume causalidad entre el ruido y las señales de voz vecinas, y el hecho de usar para la predicción un número de ventanas inferior al de la causalidad en la señal de voz. El factor no aprendido es la diferencia entre los datos de entrenamiento y los de prueba. Este último factor es difícilmente evitable, por lo que la normalización pretende paliar de alguna forma el primero. La idea es dividir el error de predicción obtenido del sistema base Eb, entre un valor Ei que, careciendo de la información acerca del hablante, tenga el mismo error inherente. Este segundo error se logra mediante una red con la misma arquitectura y entrenada de la misma forma que la del hablante cliente, pero con muestras de N hablantes.

El sistema fue probado para 12 hablantes hombres (con similares características), usando 150 palabras para entrenamiento y el resto para prueba. Las muestras de voz fueron obtenidas mediante micrófono en una habitación preparada. Como características se utilizaron 10 coeficientes cepstrales obtenidos a partir de la división del espectro de frecuencia mediante escala de Mel (MFCC). La arquitectura de las redes neuronales utilizadas es tres capas, con 20 neuronas en la de entrada, 20 en la oculta y 10 en la de salida.

Se entrenó una red por hablante, siendo la usada para la normalización común a todos. Los resultados muestran una mejora considerable con la introducción de la normalización, así por ejemplo, al usar un conjunto de 10 palabras para probar el sistema, pasaron de un EER del 41% sin normalización al 2.7% con ella. Comparando el rendimiento del modelo propuesto con otros, tales como VQ y HMM, se llega a resultados similares, pero con un menor número de parámetros en el sistema basado en redes neuronales (casi la mitad).

Otros autores que han utilizado modelos predictivos en el reconocimiento del locutor son Artieres y otros (1993) realizando una comparación extensiva entre distintos modelos, Levin (1990) proponiendo una modificación del modelo de M estados presentado por Hattori: el modelo de control oculto (Hidden control model, HCNN), que ha demostrado su efectividad en determinadas situaciones, pero con un mayor número de neuronas en la capa oculta, en general, que con el modelo inicial, y Sorensen y Hartma (1993) proponiendo un HCNN auto estructurado, donde el número de neuronas va incrementándose durante el

entrenamiento para adaptarse a la dificultad del problema; pruebas realizadas en identificación muestran unos resultados peores que con HCNN normal.

MLP como función transformada. En este caso se busca realizar transformaciones sobre los vectores de características extraídos de la señal que sean características de cada hablante.

Un enfoque similar al planteado en el caso del MLP como predictor, es el presentado por Lastrucci, Gori y Soda[22], que proponen el uso de MLPs como ***autoasociadores:*** se fuerza a la red a reproducir la entrada en la salida. Es una situación similar a la planteada en el uso del MLP para la compresión de datos. En prueba la decisión está basada en la distancia euclídea entre la entrada y la salida a la red: si la distancia medida supera un umbral prefijado ese hablante es rechazado, siendo aceptado en caso contrario.

Este enfoque es aplicado a un sistema de verificación del locutor dependiente de texto, basado en fonemas (concretamente, solo se probó con 2: /ae/ y /aa/ y de forma separada), utilizando como base de datos la DARPA-TIMIT. Como características extraídas de la señal de voz usan 20 LPCC. Con las secuencias de vectores de este tipo extraídos de las muestras de aprendizaje se entrena una red por hablante, con una arquitectura de tres capas, con 20 neuronas en la de entrada, 6 en la oculta y, obviamente, 20 en la de salida. Para la verificación de la identidad del locutor prueban inicialmente dos criterios: basar la decisión en la distancia de medida sobre una única ventana de voz, o utilizar un conjunto N de estas, concretamente de 5. En este segundo caso, la distancia es promediada de la siguiente forma:

$$D = \frac{1}{N}\sum_{i=1}^{N}(e^{Cd_i} + K) \qquad [3.43]$$

Donde C y K son constates para optimizar el efecto de la función exponencial. La tasa de equierror obtenida al usar el fonema /ae/ es de 7.9% al usar una sola ventana, bajando al 6% al usar 5. Los resultados son peores con el otro fonema. Tanto con un criterio como otro,

[22] Lastrucci L., Gori M. y Soda G. (1994). "Neural Autoassociators for Phoneme-Based Speaker Verification". Proc. Workshop on Automatic Speaker Recognition Identification and Verification, pp. 189-192, Martigny (Suiza), 1994.

aunque más con el primero, la decisión está basada en un tamaño de señal de voz excesivamente pequeño, por lo que introducen un tercer criterio establecido, en este caso, a nivel de fonema: agrupan todas las ventanas de voz correspondientes al fonema en conjuntos de 5, y sobre cada uno calculan D; es suficiente con que una distancia promediada D esté por debajo del umbral para que el fonema sea asignado al hablante bajo prueba. Los resultados del sistema mejoran considerablemente con este último criterio. Por ejemplo, en las pruebas realizadas con el fonema /ae/, y con un umbral de decisión fijo, se pasa de unos porcentajes del 46% en falsas aceptaciones y 1.3% en falsos rechazos con el segundo de los criterios indicados a nivel de ventana, a unos porcentajes del 0% y el 3.5% respectivamente. Similares mejoras se observan con el fonema /aa/.

Otra referencia a un sistema similar al anterior lo podemos encontrar en el trabajo de Gong y Haton[23]. Los autores prueban tres configuraciones distintas para entradas I_a y las correspondientes salidas deseadas I_b de la red:

- I_a e I_b pertenecen a distintos fonemas, pero pronunciados por el mismo hablante. Se busca que la red capture lo que comparten pronunciaciones de distintos sonidos por un mismo hablante.
- I_a e I_b son secuencias de vectores de características del mismo fonema pero emitidas Ia por el hablante cliente e Ib por un hablante de referencia común a todos los hablantes clientes. Se supone que la transformación obtenida es particular a cada hablante cliente.
- I_a es igual a I_b, o sea, pertenecen al mismo hablante y al mismo sonido.

Los autores señalan que esta última configuración, que es la similar a la presentada por Lastrucci, Gori y Soda, es la que mejores resultados da. El sistema es probado tanto en verificación como en identificación dependiente del texto: la palabra clave es, tan sólo, un fonema. Para ambas tareas se emplean las mismas redes (tres capas, con 4 neuronas en la oculta) entrenadas para cada hablante con 4 muestras de cada uno de los 4 fonemas usados

[23] Gong Y. y Haton J.-P. (1994). "Non-Linear Interpolation Methods for Speaker Recognition and Verification". Proc. Workshop on Automatic Speaker Recognition Identification and Verification, pp. 23-26, Martigny (Suiza), 1994.

como posibles palabras clave. Como referencia, en identificación, asignando la muestra al hablante con menor distancia acumulada entre entrada y salida de la red, el error obtenido es del 3.9%, para una población de 72 hablantes. Resaltar, tal y como indican los autores, la corta duración de las muestra de voz usadas tanto para entrenar las redes, como para el reconocimiento del locutor.

En una línea de trabajo similar a la de autores anteriores, Narendranath y otros (1994), muestran la eficacia del MLP para transformar características de voz de un hablante en características de voz de otro hablante. Concretamente, entrenan y prueban MLPs para transformar formantes vocálicos de un hablante, en formantes vocálicos de otro. Hermansky y Malayath[24] presentan en su trabajo un enfoque totalmente diferente. Parten del hecho de que cualquier muestra de voz contiene 3 tipos de información diferentes: información lingüística (qué se dice), información específica del hablante (quién lo dice) e información acerca del canal y/o el entorno en que se produce la comunicación. En el reconocimiento del locutor el interés se centra en aislar la segunda de las tres. Para intentar modelar esa información, los autores proponen un método con los siguientes tres pasos:

- Extraer de la señal de voz del hablante, vectores de características, I, con información exclusivamente lingüística.
- Extraer de la misma muestra de voz, otro conjunto de vectores de características, D, pero ahora con información tanto lingüística como del hablante.
- Estimar una función transformación, M, entre D e I, tal que la distancia entre D y M(I) sea mínima. Una vez obtenida M, en prueba, esa distancia será la utilizada para la clasificación de la muestra de voz de origen desconocido.

Notar que como D e I tienen el mismo contenido lingüístico, M transportará la información que está presente en I y no en D, es decir, la información específica del hablante. Para la obtención del vector D, dependiente del hablante, los autores prueban tres tipos de

[24]Hermansky H. y Narendranath M. (1998). "Speaker Verification Using Speaker-Specific Mappings". Proc. RLA2C Workshop on Speaker Recognition and Its Commercial and Forensic Applications, pp.111-114, Avignon (Francia), 1998

características: coeficiente cepstrales PLP de orden alto (concretamente de orden 14), coeficientes de predicción lineal (LPCs) y energía por banda de frecuencia usando escala de Mel. Para la obtención del vector I, independiente del hablante, los autores proponen dos tipos de características: coeficiente cepstrales PLP de orden bajo (concretamente de 7º orden) y las extraídas de la aplicación de una técnica que denominan "Oriented Principal Component Analysis", OPCA (Malayath, Hermansky y Kain 1997). Para implementar la función transformación se propone el uso de un MLP, con una única capa oculta con 30 neuronas. Comparando el sistema propuesto con el más clásico basado en mezcla de gaussianas (GMM) en verificación independiente del texto, bajo las mismas condiciones experimentales, los autores llegan a la conclusión de que el rendimiento de ambos es similar, pero con la ventaja del modelo propuesto de emplear menos parámetros (750 frente a los 4864 del GMM). También se observa una ligera pero consistente mejora en los resultados del nuevo modelo, cuando se aumenta la duración de la muestra de voz empleada para probar el sistema. En cuanto a las características usadas para D e I, indicar la ventaja de la energía por banda de frecuencia y las basadas en OPCA, respectivamente.

3.3.2 Red neuronal de Funciones de Base Radial (RBFNN).

Han sido generalmente utilizadas como una alternativa al Perceptron multicapa en clasificación directa de patrones: tienen similares propiedades discrimantes. Así por ejemplo, Oglesby y Mason en el trabajo presentado en ICASSP'90 (ya comentado en la parte referida al Perceptron), identifican como principales problemas del MLP su excesivo tiempo de entrenamiento y el excesivo tamaño que alcanza la red cuando la complejidad del problema aumenta. Para intentar aliviar estos problemas los mismos autores proponen en su trabajo del 1991 (ICASS'91) el uso de RBFs. Las condiciones experimentales son similares a las ya presentadas anteriormente para el MLP, en cuanto a que se entrena una red por hablante. El sistema fue probado en verificación, para 40 hablantes, obteniendo, según los autores, un rendimiento mayor que con sistemas basados en MLP y VQ. El mejor resultado, para una secuencia de prueba compuesta por 4 dígitos, y con una red con 384 neuronas en la capa oculta, es de un 8% en falsos rechazos y un 1% falsas aceptaciones.

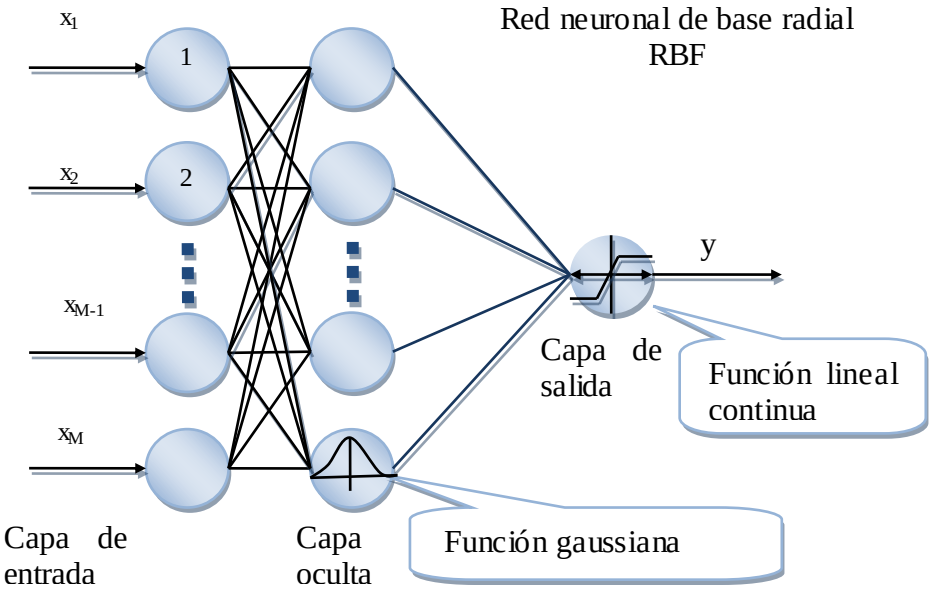


Figura 22 Topología particular de la RBF.

Fredrickson y Tarassenko[25] plantean un sistema de identificación dependiente del texto, basado en letras aisladas del alfabeto. Como funciones de activación de base radial de la primera capa de la RBFNN, se usan gaussianas de matriz de covarianza diagonal; la salida de cada una de estas neuronas será:

$$\phi(\|\vec{x}_p - \vec{C}_j\|) = exp\left(-\sum_{i=1}^{N_I} \frac{(x_i - C_{ji})^2}{2\sigma_{ji}^2}\right) \qquad [3.44]$$

Donde $\vec{x}_p$ es el p-ésimo vector de entrada a la red, N_I es el número de componentes de estos vectores, $\vec{C}_j$ es el centroide correspondiente a la neurona j de la capa oculta y σ_j es su

[25] Fredrickson S. E. y Tarassenko L. (1994). "Radial Basis Functions for Speaker Identification". Proc. Workshop on Automatic Speaker Recognition Identification and Verification, pp. 107-110, Martigny (Suiza), 1994.

matriz de covarianza diagonal. La segunda capa, la de salida de la red, realiza una aplicación lineal sobre las salidas de la capa anterior:

$$y_{kp} = w_{k0} + \sum_{j=1}^{N_C} w_{kj}\phi\left(\left\|\vec{x}_p - \vec{C}_j\right\|\right) \qquad [3.45]$$

Donde y_{kp} es el k-esimo nodo de salida de la red, w_k es su vector de pesos y N_C es número de neuronas en la capa anterior.

Los primeros experimentos son un estudio comparativo entre MLP y RBF, obteniendo una mayor eficacia con este segundo tipo de redes. La base de datos utilizada para estas pruebas está compuesta por emisiones aisladas de las 10 primeras letras del alfabeto, realizadas por 6 hablantes distintos; la grabación se realizó mediante un micrófono. La señal de voz correspondiente a cada letra es dividida en ventanas de 256 muestras, extrayendo de cada ventana 8 coeficientes cepstrales vía FFT; mediante una normalización temporal lineal el número de vectores de características se reduce a 8 para todas las muestras de voz: todas serán representadas, por tanto, mediante un vector de características final de 64 componentes (entradas a la red). Tanto en el caso MLP, como RBFNN, para realizar la identificación se entrena una única red con 6 salidas, 1 para cada hablante, usando para ello 10 muestras de cada letra del alfabeto (600 patrones de entrenamiento). El mejor resultado en el caso MLP se obtiene con una red con 24 neuronas en la capa oculta y es de un 29.5% de error. Para el caso RBF el peor de los resultados referenciados es de un 25.8% de error, para 60 neuronas en la capa oculta, que baja a un 21% para 300, límite máximo impuesto siguiendo la norma de que para que la red sea capaz de generalizar, el número de neuronas en la capa oculta tiene que ser menor que el número de patrones de entrenamiento; los autores fijaron el límite en la mitad de esta cantidad. La primera capa de la RBFNN fue entrenada mediante el algoritmo de los kmedios (k-means), y los pesos de la segunda mediante la técnica de la inversión de la matriz.

A raíz del resultado del estudio comparativo, los autores deciden centrar sus esfuerzos en un estudio más profundo del rendimiento de las RBFNN en la tarea propuesta. Tras un análisis del comportamiento individual de cada letra por separado, realizan la identificación usando conjuntos de tamaño r variable de estas. La salida final es el producto de las salidas individuales para cada letra: estas están siendo interpretadas como una probabilidad; el índice de la salida con un valor final mayor, identifica al hablante. Para una población de 50 hablantes (50 neuronas de salida en la red), usando 1000 patrones de entrenamiento (2 ejemplares por letra, 10 letras, para cada hablante), 500 patrones de prueba (1 ejemplar por letra para cada hablante) y con una red con 300 neuronas (dendroides) en la capa oculta, los resultados obtenidos varían entre un 23.8% de error para r=1, 0.84% para r=5 y 0% para r=10, usando las 10 letras con mejores resultados, y unos errores de 31.38%, 2.83% y 0.2% respectivamente, usando 10 letras escogidas aleatoriamente.

3.3.3 LVQ y SOM.

El LVQ es, quizás, junto al MLP uno de los primeros modelos conexionistas empleados en el reconocimiento del locutor. Surge como un refinamiento de una técnica más clásica como es el VQ. También es conveniente comentar que no ha tenido un uso muy extendido. Una de las primeras referencias de utilización la encontramos en Bennani, Fogelman y Gallinari[26]. Más concretamente, en su trabajo intentan comparar el rendimiento de dos parametrizaciones diferentes de la señal de voz: 12 coeficientes LPC (*Linear Predictive Coding*) y 8 coeficientes MFCC (*Mel Frequency Cepstral Coding*), usando para ello un sistema basado en LVQ. La tarea abordada en la identificación dependiente de texto, mediante una sentencia en francés de duración 2.5s., y para una población de 10 hablantes. El conjunto de vectores de características extraídos de la señal de voz correspondiente a una sentencia es representada mediante su media y el primer autovector de la matriz de covarianza: viendo el conjunto de vectores como una nube de puntos en el espacio de características, la media representa su posición, y el autovector su forma. La base de datos

[26] Bennani Y., Fogelman F. y Gallinary P. (1990). "A Connectionist Approach for Automatic Speaker Identification". Proc. IEEE ICASSP, S5.2, pp. 265-268, Albuquerque, New Mexico (USA), 1990

contiene 10 repeticiones de la sentencia usada en la identificación, de las cuales 9 son usadas para entrenar el modelo LVQ y la restante para prueba; se hacen 10 experimentos distintos variando el contenido del conjunto de entrenamiento y prueba (técnica "leave-one-out"). El LVQ es inicializado mediante un "k-means" con K=2 o 3, para cada hablante. Los autores muestran unos resultados superiores con la parametrización MFCC, alcanzando tasas de error en la identificación inferior al 3%.

Muy poco se ha escrito sobre la utilización de los mapas autoorganizados de Kohonen en el reconocimiento del locutor. Una de esas pocas referencias la podemos encontrar en el trabajo realizado por Anderson y Patterson[27], que utilizan una red de Kohonen para inicializar un LVQ. El objetivo del trabajo indicado es comparar el rendimiento de parámetros auditivos, basados en los modelos de Patterson (*Auditory Image Model*, AIM) y de Payton (PAM), frente a parámetros más clásicos en el tratamiento de la voz como son los cepstrales, concretamente, MFCC. La tarea abordada es la identificación dependiente de texto, mediante sonidos vocálicos extraídos de sentencias de la base de datos TIMIT, y para una población de 37 hablantes. Entrenan una red, o mejor, un modelo LVQ por hablante, de la siguiente forma: con vocales extraídas de 7 sentencias correspondientes a una misma persona, se entrena un SOM de 8x8 neuronas, con disminución tanto del parámetro vecindad, como del de aprendizaje. Una vez entrenada se etiqueta cada nodo, asignándole a aquel fonema cuyas muestras más veces le hayan activado, para, a continuación, mediante un LVQ (concretamente LVQ3) realizar un ajuste más fino de los centroides: el resultado final es un modelo con 64 centroides. El proceso de identificación del hablante al que pertenece una determinada sentencia se realiza extrayendo las muestras correspondientes a las vocales, y midiendo la distorsión Ds media mínima para cada modelo entrenado **s**:

$$D_s = \frac{1}{N}\sum_{i=1}^{N} \min_{jek} \left\| x_i - m_{zj} \right\|^2 \qquad [3.46]$$

[27] Anderson T. R. y Patterson R. D. (1994). "Speaker Recognition with Auditory Image Model and Self Organizing Feature Maps: A Comparison With Traditional Techniques". Proc. Workshop on Automatic Speaker Recognition Identification and Verification, pp. 153-156, Martigny (Suiza), 1994.

Donde N es el número de vectores de características x_i, y el índice sobre los centroides m_{sj} es k=1,…,64. La muestra de voz se asigna al hablante cuyo modelo tenga una distorsión menor. Las pruebas se realizaron con una sentencia por hablante (37 pruebas en total), obteniendo unos errores de verificación del 6% para los parámetros MFCC, del 9% para los basados en AIM y del 33% para los basados en PAM.

3.3.5 Redes recurrentes.

Nuevamente nos encontramos con un tipo de redes muy poco utilizadas. Dentro de esos pocos trabajos presentados, referenciar el realizado por Tsoi, Shrimpton, Watson y Back[28]. Estos autores prueban el rendimiento de un modelo recurrente: arquitectura Frasconi-Gori-Soda (AFGS), comparando el rendimiento de éste frente al de un MLP en la misma tarea. La arquitectura indicada es una extensión del modelo más simple de Jordan-Elman, y pertenece a la clase de modelos recurrentes denominados "*locally recurrent globally feedforwad*" (LRGF), que se caracterizan porque la recurrencia es local a la neurona.

Sobre un modelo de neurona clásico, modelo de McCulloch-Pitt, la recurrencia se puede establecer en tres puntos distintos:

- En las conexiones de entrada a la neurona: se establece una realimentación en cada una de las conexiones sinápticas.
- Sobre la salida de activación a(t): a las entradas a la neurona se añaden valores anteriores de salidas de activación de dicha neurona, o sea, se produce una realimentación (con los retardos correspondientes) de la salida de activación de la neurona a la entrada.

[28] Tsoi A. C., Shrimpton D., Watson B. y Back A. (1994). "Application of Neural Network Techniques to Speaker Verification". Proc. Workshop on Automatic Speaker Recognition Identification and Verification, pp. 143-152, Martigny (Suiza), 1994.

- Sobre la salida de la neurona y(t): es similar al caso anterior, pero ahora la realimentación se realiza después de haber aplicado la función no lineal sobre la salida de activación.

La AFGS se corresponde a este último caso, y en ella la salida de la neurona en un instante t será:

$$y(t) = f\left(\sum_{j=1}^{m} k_j y(t-j) + \sum_{i=0}^{n} w_i x_i(t)\right) \qquad [3.47]$$

Donde f va a ser una función sigmoide.

Descrito el modelo, pasamos a describir los experimentos realizados. Realiza una verificación dependiente del texto mediante dígitos aislados, y para una población de 10 hablantes. Compara el rendimiento de la red recurrente presentada, frente a un MLP, ambas con una misma arquitectura de tres capas, con 20 neuronas en la oculta, y 2 en la de salida. Como características extraídas de la señal de voz utilizan 10 LPCC, su derivada primera y su derivada segunda (tamaño del vector de características: 30).

Estos valores de entrada a la red son normalizados entre -1 y 1, para después multiplicarles por un valor de ganancia de 50. Cada nodo de salida de la red se hace corresponder a la clase cliente y a la clase impostor respectivamente, de forma que cuando la muestra de entrenamiento pertenece al hablante cliente, la salida deseada se corresponderá con una secuencia exponencial creciente, pasando a ser ésta decreciente en el caso de que la muestra de entrenamiento pertenezca a un impostor. Se entrena una red distinta para cada dígito y cada hablante, para lo que utiliza 6 muestras distintas del cliente y 6 de tres hablantes impostores distintos. En el proceso de aprendizaje se van alternando las muestras del cliente y de impostores, por los que cada muestra del primero se repite tres veces en cada época de entrenamiento. En prueba, la salida para una muestra de voz de origen desconocido será la

media de las salidas obtenidas en el nodo perteneciente al hablante cliente. La decisión final se realiza combinando salidas de ese tipo obtenidas para conjuntos de 10 dígitos.

Utilizando como criterio de finalización del proceso de aprendizaje un número fijo de 40 épocas, las tasas de equierror obtenidas son del 7.5% para la red recurrente y del 6% para el MLP. Se puede observar que el uso de la recurrencia no mejora el rendimiento del sistema. Otra referencia de uso de redes recurrentes las podemos encontrar en el trabajo Shrimpton y Watson (1992) con resultados y configuración de experimento muy similares al anterior.

3.4 Otras

Se incluyen en esta sección aquellas técnicas que resultan de la conbinacion de otras. En este caso se hace referencia al modelo hibrido sobre el que ha hecho un mayor numero de publicaciones.

3.4.1 Modelos híbridos HMM-ANN.

Son técnicas que intentan aunar la capacidad de los HMMs para modelar secuencias temporales, mediante una estructura multiestado, con la capacidad tanto discriminante, como de síntesis de funciones de las ANNs. Han sido aplicadas en el caso de clasificación basada en modelos predictivos, como una extensión natural de los modelos de un solo estado, ya referenciados anteriormente. Como indicábamos allí, el problema de los modelos de un solo estado es que están suponiendo una naturaleza estacionaria en el fenómeno a estudiar, que no es cierta. La solución planteada es utilizar a un modelo con M estados, donde cada estado es un predictor (MLP) diferente, especializado en una determinada parte de la muestra de voz del locutor. Este sistema puede ser considerado como un híbrido entre HMMs y MLPs. En este caso, dado un modelo, la probabilidad de que genere la secuencia de vectores de características $(x_1,\ldots,x_L)$ vendrá dada por:

$$P_i(x_1^L/S_1^L) = \prod_t P_I(x_t/x_T, S_t) \qquad [3.48]$$

Donde x_1^L es la secuencia óptima de estados (error de predicción menor), x es el contexto de predicción, o sea, x_{t-1} y x_{t-2}. En entrenamiento se optimizan tanto los parámetros de las redes, como la segmentación, es decir, $P_i(x_1^L/S_1^L)$. Han sido probados tanto modelos de izquierda-derecha, como ergódicos. Con estos últimos, se ha logrado alcanzar un error de identificación del 0%, con un modelo de 4 estados. Los experimentos se realizaron con 24 hablantes de la base de datos TIMIT, usando 14 segundos de voz para entrenamiento y 3 para prueba. De pruebas comparativas realizadas, el sistema presentado mejora los resultados obtenidos con modelos como VQ, HMM y ANNs discriminantes.

Un sistema similar es el presentado por Hassanein, Deng y Elmasry[29], al que denominan modelo oculto de Markov predictivo neuronal (*Neural predictive* HMM). Éste consiste en un modelo de izquierda-derecha donde, al igual que antes, cada estado es sustituido por un MLP que funciona como predictor para un segmento de la señal de comportamiento cuasi-estacionario. El entrenamiento es un proceso iterativo de dos pasos: segmentación y estimación. La segmentación consiste en la asignación de porciones de señal a cada MLP, y se realiza mediante un procedimiento de programación dinámica normal. Una vez segmentada la señal, se aplica el algoritmo de retro-propagación del error para ajustar los pesos de las redes asociadas a cada segmento de voz. Este proceso iterativo se repite hasta lograr la convergencia, punto que se alcanza si el resultado de la segmentación se estabiliza o si el error de predicción cae por debajo de un determinado valor umbral. Los autores prueban con un modelo de 4 estados. Las redes neuronales tienen una arquitectura de 3 capas, con 7 neuronas en la capa de entrada, 5 en la oculta y 7 en la de salida. La representación de cada ventana de señal de voz se realiza mediante un vector de características compuesto por 7 MFCC. De la arquitectura de la red se deduce, entonces, que la "historia" utilizada en la predicción se reduce a la ventana anterior. El sistema se

[29] Hassanain K., Deng L. y Elmasry M. I. (1994). "A Neural Predictive Hidden Markov Model for Speaker Recognition". Proc. Workshop on Automatic Speaker Recognition Identification and Verification, pp. 115-118, Martigny (Suiza), 1994.

prueba mediante una base de datos cuyo contenido son 6 sílabas distintas, del tipo consonante-vocal, repetidas cada una 22 veces (8 para entrenamiento y 14 para prueba), por 11 hablantes diferentes (6 hombres y 5 mujeres); las muestras de voz se obtuvieron mediante un micrófono en una única sesión. En identificación dependiente del texto, siendo el "texto" la sílaba /di/, se obtuvo un error del 0.7%, para una población de 10 hablantes (5 h. y 5 m.), entrenando un modelo para cada hablante. En verificación dependiente del texto, siendo el "texto" también sílabas, pero ahora todas: se entrenó un sistema por hablante y sílaba, se obtuvieron unos resultados de 0% en falsas aceptaciones (el valor del umbral se fijo para obtener este valor) y 1.6% en falsos rechazo; ahora la población de hablantes se redujo a 3 (todos hombres), utilizando las muestras de los otros 3 hombres de la base de datos para las pruebas sobre impostores.

Un enfoque diferente lo encontramos en las denominadas "alpha-nets" (Bridle, 1990). La técnica propuesta aplica las habilidades discriminantes del MLP a los HMM. Concretamente, una vez que el HMM está entrenado, se realiza un ajuste de las medias y las desviaciones de las gaussianas definidas en cada estado, aplicando un algoritmo de gradiente descendiente en derivadas parciales basado en el de retro-propagación de los MLP. Carey, Parris y Bridle (1991) aplican esta técnica a la verificación del locutor dependiente del texto, mediante palabras clave. Por cada hablante se entrenan dos modelos, uno con muestras del hablante cliente, y otro con muestras de hablantes distintos a este. En prueba se resta la salida de ambos, si el resultado es mayor que el umbral el hablante es aceptado, siendo rechazado en caso contrario. Los experimentos realizados indican una mejora global del rendimiento del sistema tras el "ajuste discriminante" de los parámetros: empeora la eficacia en falsos rechazos, pero en menor medida de los que mejora en falsas aceptaciones. El sistema fue probado en condiciones reales de uso: intento de acceso telefónico, mediante una secuencia de 5 digitos. Para cada hablante y dígito se crea un modelo como el descrito, de forma que es probado con 50 secuencias de 5 dígitos del propio hablante y 10 de cada uno de los otros. Cada dígito es valorado por separado, de forma que el criterio final de decisión es "el más votado", o sea, si los resultados para tres o

más dígitos dicen que las muestras de voz correspondientes pertenecen al hablante cliente, éste es aceptado. De las 600 pruebas realizadas, el sistema solo erró en 1.

Zeek[30] presenta en su trabajo una técnica que podríamos denominar a caballo entre un sistema híbrido y un sistema modular. Como sistema clasificador principal utiliza un HMM, de forma que la salida de éste: P(X/l) (con X la muestra de voz y l el modelo perteneciente a un determinado hablante), probabilidad no necesariamente discriminante, es procesada posteriormente por un MLP para convertirla en probabilidad a posteriori: P(l/X). La inclusión del post-procesamiento indicado mejora notablemente el rendimiento del sistema. Así, por ejemplo, en verificación dependiente de texto, para una población de 32 hablantes, creando un modelo para cada palabra de la frase clave, se pasa de un error del 19.5% usando sólo HMMs, a un error del 6.5% con la técnica propuesta. Un problema interesante que se plantea es el de la normalización de la entrada a la red, o sea, normalizar el vector de salidas de los HMMs. El autor indica como la más eficaz de las probadas la denominada normalización estadística: el vector de entrada a la red se normaliza de forma que tenga media 0 y varianza unidad:

$$a_{ij}^{n} = \frac{a_{ij} - \mu_j}{\sigma_j} \qquad [3.49]$$

Donde el subíndice i hace referencia a cada neurona de entrada a la red y el "j" al vector de entrada que se está normalizando. El superíndice n indica el valor normalizado.

3.4.2 Redes neuronales en la etapa de clasificación: sistemas modulares.

Uno de los primeros intentos de "modularizar" la etapa de decisión lo podemos encontrar en el trabajo de Rudasi y Zahorian (1991). Estos autores plantean una técnica a la que denominan "binary-pair neural networks", como alternativa a los sistemas de identificación

[30] Zeek E. J. (1996). "Speaker Recognition by Hidden Markov Models". Tesis presentada en la escuela de ingeniería del instituto de tecnología del ejército del aire (USA), 1996.

del locutor basados en una red neuronal única. El problema de este último tipo de sistemas se plantea cuando la población de hablantes se incrementa: cada vez que se añade un nuevo usuario al sistema, ha de reentrenarse la red, implicando un continuo aumento en la complejidad de ésta, lo que afecta su rendimiento. La alternativa propuesta consiste en dividir los N hablantes a identificar en grupos de 2 (N(N-1)/2 grupos), entrenando una red distinta para cada pareja. Estas redes son del tipo MLP, con una única capa oculta de 6 neuronas, y dos salidas, cada una asociada a un hablante de la pareja. Una vez entrenadas, para la asignación de una muestra de voz de origen desconocido, se plantean dos métodos distintos:

"Global soft decisión search": la muestra de voz, o mejor, los vectores de características de ésta extraídos, sirven de entrada a todas las redes. Los valores de salida asignados a cada hablante son sumados, de forma que la asignación se realizará al locutor con el valor de la suma mayor.

"Binary tree search": las N clases (hablantes) son pareados (N/2 pares), de cada pareja se elimina al hablante cuya salida de la red asociada sea menor, de forma que a la siguiente "ronda" solo pasan la mitad de los N locutores. Esta operación se va repitiendo hasta que sólo quede uno.

El segundo método no solo es más rápido, sino que, además, proporciona mejores resultados. El sistema se probó en comparación con la técnica de red única. Como características extraídas de la señal de voz se usaron 15 coeficientes cepstrales. El criterio de convergencia para ambas técnicas es el mismo: sobrepasar un porcentaje de aciertos umbral en el conjunto de datos de entrenamiento (este umbral varió del 30% al 50% en el caso de red única y del 65% al 75% en el enfoque de partición binaria, dependiendo del número de nodos en la capa oculta y de N). En las pruebas realizadas para identificación independiente del texto, sobre la base de datos DARPA-TIMIT, usando 5 sentencias diferentes para entrenar la red, con ambas técnicas se logró un 100% de aciertos, para una muestra de voz suficiente larga y los parámetros de las redes optimizados. La ventaja del método propuesto la encontramos cuando hay que actualizar el sistema ante la incorporación de nuevos hablantes: el tiempo necesario para entrenar la red única es

superior. En contraposición a la ventaja que acabamos de exponer, el método propuesto por Rudasi y Zahorian "supone un alto coste en cuanto a número de redes, éste es proporcional al cuadrado del número de hablantes"[31]. La propuesta de Bennani (1992) pretende aunar las ventajas de la división del trabajo de clasificación entre "expertos", implícito en el trabajo anterior, con un coste no tan alto en recursos. La idea básica es simple: en vez de crear parejas de hablantes, hagamos grupos más numerosos de acuerdo a algún criterio de similitud entre ellos, entrenando una red para cada uno de estos grupos, a los que los autores se refieren como "tipologías".

Como características de la señal de voz se usan vectores de LPCCs, agrupando estos mediante la técnica de los k-medios. Cada hablante se asigna a la agrupación donde se encuentren la mayoría de sus vectores de características, formando así las distintas tipologías. Con los datos así obtenidos, se entrenan dos tipos de redes: una para cada tipología (la llamaremos Red de Tipología RT), encargada de diferenciar entre los distintos hablantes que la componen, y otra encargada de identificar la tipología a la que pertenece una determinada muestra de voz (la llamaremos Red Identificadora de Tipología RIT). El tipo de red neuronal empleada es el TDNN, con tres capas. Para la identificación de la muestra de voz de origen desconocido se prueban dos métodos distintos:

- "Todo para el ganador". En este caso la RIT actúa a modo de puerta, o conmutador entre las distintas RTs: sólo la salida de la red correspondiente a la tipología detectada es considerada, siendo ésta la salida final del sistema.
- "Salida ponderada". Para cada ventana de voz, los valores de salida de cada RT, es multiplicada por la salida correspondiente de la RIT (el mismo para todas las salidas de una determinada RT).

Tras sumar los valores así obtenidos para todas las ventanas de la muestra de voz, ésta es asignada al hablante asociada a la salida con un resultado de la suma mayor.

[31] Rudasi L. y Zahorian S. A. (1991). "Text Independent Talker Identification with Neural Networks". Proc. IEEE ICASP, S6.6, pp. 389-392, Toronto (Canada), 1991.

Utilizando este segundo método para la identificación, los autores comparan el rendimiento del sistema modular propuesto, con el basado en MARMs "Multi-variate Auto-Regressive Models"[32]. De las pruebas realizadas sobre la misma base de datos (los 2 primeros dialectos de la TIMIT, 102 hablantes), los resultados referenciados son del 100% en el sistema modular y del 95.6% en el basado en MARMs. Los autores concluyen que el método propuesto es discriminante y rápido en la fase de identificación. Además, para ésta solo se requieren muestras de voz de corta duración: 0.75s. Por contra, el tiempo de entrenamiento es muy elevado.

Haciendo referencia a la división de tareas indicada en la introducción del apartado al hablar de sistemas modulares: clasificación e integración, en los sistemas presentados las redes forman parte exclusivamente de la tarea de clasificación, realizándose la integración de los diversos resultados de formas diferentes, pero nunca con la intervención de redes neuronales. Sharma, Vermeulen y Hermansky (1998) presentan en su trabajo la situación inversa: la red neuronal, concretamente un MLP, se ocupa únicamente de la integración de las respuestas proporcionadas por dos técnicas de clasificación distintas, frente a una misma muestra de voz. La ventaja del uso de redes neuronales en esa tarea es que permite una integración no lineal de las distintas salidas. La tarea abordada es la verificación del locutor independiente de texto. Prueba el rendimiento del sistema modular en tres situaciones distintas:

- Cuando los sistemas cuyas salidas se inregran tienen por separado un rendimiento similar: mezclar ambos mejora los resultados.
- Cuando uno de los dos sistemas mejora al otro: el rendimiento del sistema propuesto es similar al del mejor de ambos.
- Cada sistema a integrar funciona mejor que el otro bajo determinadas condiciones de operación.

[32] Artieres T., Bennani Y., Gallinari P. y Montacie C. (1991). "Connectionist and Conventional Models for Free Text Talker Identification". Neuro-Nimes, Francia, 1991.

En este caso a la red que realiza la integración se le añaden nuevas entradas para las "condiciones de operación". El sistema combinado mejora el rendimiento de ambos por separado, de forma que su eficacia se aproxima, sean cual sean las condiciones de operación, a la del mejor en cada caso.

Un enfoque distinto a los presentados, donde se diferencian claramente clasificación e integración, es el presentado más recientemente, por Hadjitodorov, Boyanov y Dalakchieva [33], para la tarea de identificación independiente del texto. En su trabajo presentan un sistema clasificador con dos niveles, basados ambos en redes neuronales: SOMs en el primero y MLPs para el segundo, de forma que la entrada del segundo nivel es obtenida a partir de las salidas del primero. Intentan aunar la capacidad de estimación de las funciones de densidad de probabilidad de los vectores de entrada, incluso con señal ruidosa, del SOM, con la capacidad discriminante del MLP. El proceso de entrenamiento nos permite tener una visión clara del funcionamiento y características del sistema. De las muestras de voz para entrenamiento correspondientes a cada hablante se extraen las secuencias de vectores de características,15 LPCC por ventana en este caso, que servirán de entrada al SOM. Una vez entrenado éste, se le presentan nuevamente, pero ahora con los pesos fijos, las L muestras de voz de entrenamiento de cada hablante, evaluando, para cada hablante y muestra de entrenamiento, la frecuencia de activación de cada una de las neuronas de la red: f_{ij} , siendo i y j las coordenadas de la neurona. Se obtiene, de esta forma, para cada hablante y muestra de entrenamiento, una matriz de valores, a la que se denomina PDM (*Prototype Distributing Map*). Con el conjunto de PDMs obtenidos, se entrena, en el segundo nivel, un MLP por hablante. Cada una de estas redes se entrena para diferenciar a un hablante del resto, por lo que la salida deseada se fija a 1 si el PDM pertenece al hablante, y a 0 si pertenece a alguno de los restantes.

En el procedimiento descrito aparecen dos problemas. El primero es el escaso número de vectores que se tiene para entrenar el segundo nivel. Se aumenta dividiendo el conjunto de entrenamiento en T grupos disjuntos distintos, y entrenando para cada uno de estos un SOM diferente. De esta manera el número de PDMs por hablante será, ahora, de TxL El

[33] Hadjitorov S., Boyanov B. y Dalakchieva N. (1997). "A Two Level Classifier for Text Independent Speaker Identification". Speech Communication, Vol 21, No. 3, pp. 209-217, Abril 1997

segundo problema es la posible influencia de las frecuencias menos significativas. Para evitarlo los valores de cada PDM son filtrados de la siguiente manera:

$$si\ 0 \leq f_{ij} \leq Kf_{max} \quad entonces\ f_{ij} = 0$$

$$si\ Kf_{max} \leq f_{ij} \leq f_{max} \quad entonces\ f_{ij} = f_{ij}$$

Donde $0 < K < 1$ es un coeficiente de filtrado ajustado experimentalmente, y f_{max} es el valor máximo de f_{ij}.

En prueba, de la muestra de voz de origen desconocido se obtienen T PDMs, cada uno de los cuales será procesado por cada una de las redes neuronales del segundo nivel, sumándose cada una de las salidas así obtenidas. La muestra de voz se asignará al hablante cuya red tenga la mayor salida acumulada. Se realizó un estudio comparativo del rendimiento del sistema presentado, frente al obtenido por sistemas basados en SOMs sólo: SOM+LVQ3 con una red única de 15x15 neuronas para todos los hablantes, basados en MLPs sólo: una red por hablante con dos capas ocultas de 64 y 4 neuronas respectivamente y 1 en la de salida, y frente a sistemas basados en modelos autoregresivos (AR-models). Las condiciones experimentales, las mismas para todos los sistemas, son:

- Pruebas con señal limpia: 68 hablante (35 hombres y 33 mujeres), 2 sentencias de 4 a 7 segundos por hablante para entrenamiento, también dos sentencias de prueba por hablante, distintas a las de entrenamiento y obtenidas en sesiones distintas a las de éste.
- Pruebas con señal telefónica: 92 hablantes (48 hombre y 44 mujeres), 3 sentencias de 4 a 7 segundos por hablante para entrenamiento, también 3 sentencias por hablante para prueba, con las mismas características que las del punto anterior.

Con señal limpia los errores de identificación fueron del 3.67% con MLP solo, del 2.94% con SOM+LVQ solo, del 3.67% con los modelos auto-regresivos y del 2.2% con el sistema

en dos niveles propuesto. Para la señal telefónica los errores obtenidos, puestos en el mismo orden que antes, fueron: 6.15%, 5.43%, 5.79% y 2.17%

3.4.3 Máquinas de soporte vectorial (*SVM, Support Vector Machines*)

Las Máquinas de Soporte Vectorial constituyen el estado del arte en tareas de clasificación no lineal. Sus ventajas frente a otros clasificadores tradicionales han llamado la atención de numerosos investigadores en diversas áreas, entre ellas la de reconocimiento automático de habla. Fundamentalmente, hay tres razones que han conducido a plantear el uso de las SVMs como posible alternativa a los Modelos Ocultos de Markov (HMMs) en reconocimiento de voz: a) el modelado acústico es fundamentalmente un problema de clasificación de patrones, mientras que los HMMs son modelos generativos y, por tanto, lo que buscan es estimar las funciones de densidad de probabilidad correspondientes a cada uno de los modelos; b) la función de coste definida por las SVMs establece un compromiso entre la minimización del riesgo empírico y del riesgo estructural, lo que produce máquinas con una mayor capacidad de generalización frente a otros clasificadores tradicionales; c) el entrenamiento de las SVMs busca la maximización del margen, definido como la distancia de las muestras de entrenamiento a la frontera de decisión, lo que a priori les hace más robustas frente a condiciones ruidosas, uno de los principales problemas en reconocimiento de habla.

No obstante, la aplicación de las SVMs en reconocimiento de habla no es inmediata, debido principalmente a que estas máquinas trabajan con vectores de entrada de dimensión fija. Por el contrario, los métodos más comunes de parametrización de la voz producen secuencias de parámetros de longitud variable, dependiendo de la duración de la locución. En los últimos años han aparecido diversas formas de abordar este problema. El sistema de reconocimiento de habla continua mediante SVMs que se presenta en [2] lo evita trabajando trama a trama. El reconocedor propuesto emplea una SVM multiclase para determinar a qué clase pertenece el segmento de voz considerado y, a continuación, se usa

el algoritmo de Viterbi para obtener una secuencia de palabras a partir de las decisiones acústicas que proporciona la SVM.

Como es sabido, los HMMs contribuyen al proceso de reconocimiento de la secuencia de parámetros espectrales de entrada aportando la verosimilitud de que la muestra actual haya sido generada por cada uno de los modelos que constituyen el sistema de reconocimiento de voz (palabras, fonemas, trifonemas...). Las SVMs, en cambio, proporcionan la etiqueta de la clase asignada al vector de entrada. Para aquellas aplicaciones en las que el clasificador solamente se encarga de una parte de la decisión global, se han propuesto diversas formas para estimar probabilidades multiclase a partir de las salidas blandas de las SVMs. Su planteamiento depende principalmente de la estrategia multiclase que se adopte en la SVM (1-contra-1 ó 1-contra-el resto), las cuales se detallarán en la siguiente sección. Una de las más usadas, se basa en el cálculo de la probabilidad de Platt, para cada SVM binaria (en la aproximación 1-contra-1) y el empleo de una variante del método de Refregier-Vallet para obtener las probabilidades multiclase. No obstante, el fundamento teórico de los métodos citados no está suficientemente justificado, tal y como muestran los experimentos realizados. Su principal debilidad consiste en la hipótesis de gaussianidad en las funciones de densidad de probabilidad condicional de la salida de las SVMs binarias, dada una cierta clase ($p(f|y = +1)$ y $p(f|y = -1)$, donde f denota la salida blanda de la SVM binaria para la muestra actual y +1 y −1 son las etiquetas asociadas a las dos clases consideradas). En esta seccion se exploran diversas alternativas basadas en el uso directo de las salidas blandas de las SVMs, evitando en lo posible hipótesis como la señalada anteriormente.

Fundamentos de las SVMs. Una SVM es un clasificador binario que asigna una etiqueta $Y \in \{+1, -2\}$ al vector de entrada x conforme al signo de la siguiente expresión:

$$f(x) = w^t \cdot \phi(x) + b, \qquad [3.50]$$

donde $\phi(x) : R^d \rightarrow R^H$ ($d << H$) es una transformación del espacio de entrada a un espacio de características de mayor dimensión (incluso infinita), en el que se supone que las clases

son linealmente separables. El vector w define el hiperplano de separación en dicho espacio y b representa el sesgo respecto al origen de coordenadas.

La razón que hace a las SVMs más robustas que otros clasificadores es su criterio de entrenamiento, consistente en un compromiso entre la minimización del riesgo empírico y del riesgo estructural. Éste último evita un posible sobreajuste de la máquina al conjunto de entrenamiento.

La solución viene dada por el siguiente problema de minimización cuadrática:

$$\min_{w,b,\xi_i} \frac{1}{2}\left|\left|w\right|\right|^2 + C\sum_{i=1}^{n}\xi_i, \qquad [3.51]$$

$$\text{Sujeto a} \quad y_i(w^t.\phi(x) + b,) \geq 1 - \xi_i, \qquad [3.52]$$

$$\xi_i \geq 0; i = 1, \ldots, n,$$

Donde $x_i \in R^d$ $(i = 1, \ldots, n)$ son las muestras de entrenamiento con etiquetas $y_i \in \{+1, -2\}$. Las variables ξ_i miden el error de entrenamiento de cada muestra y C es un factor de ponderación entre el riesgo empírico y el riesgo estructural.

El problema de programación cuadrática en [3.52] se suele resolver empleando el dual de Wolfe, en el que los multiplicadores de Lagrange α_i se calculan maximizando:

$$\max_{\alpha_i} \sum_{i=1}^{n}\alpha_i - \sum_{i=1}^{n}\sum_{j=1}^{n} y_i y_j \alpha_i \alpha_j \phi^T(x_i)\phi(x_j) \qquad [3.53]$$

$$\text{Sujeto a} \sum_{j=1}^{n} \alpha_j y_i = 0, 0 \leq \alpha_i \leq C$$

Este problema es cuadrático y convexo, por lo que la convergencia al mínimo global está asegurada. Una vez resuelto, el vector de pesos w se puede expresar de la siguiente forma:

$$w = \sum_{i=1}^{n} \alpha_i y_i \phi(x_i) \qquad [3.54]$$

Sólo aquellas muestras cuyo multiplicador asociado α_i sea distinto de 0 contribuyen a la definición de la frontera de decisión, razón por la que reciben el nombre de vectores soporte.

Normalmente, la función $\phi(x)$ no se conoce de forma explícita o es imposible de evaluar. No obstante, esto no plantea ninguna dificultad, ya que si nos fijamos en las expresioes [3.51 y [3.53] veremos que únicamente se precisa calcular los productos escalares $\phi^T(x_i) \cdot \phi(x_j)$, los cuales, empleando lo que se ha denominado truco del kernel, se pueden evaluar mediante la función de kernel $K(x_i, x_j)$. De esta forma, la salida blanda de la SVM adopta la siguiente expresión:

$$f(x) = \sum_{i=1}^{n} \alpha_i y_i K(x_i, x) + b \qquad 3.55$$

Los kernels más comunes son el lineal (KL):

$$K_L(x_i, x) = x_i^T . x_j \qquad [3.56]$$

Y el gaussiano (K_{RBF}):

$$K_{RBF}(x_i, x) = exp\left(-\frac{\|x_i - x_j\|^2}{2\sigma^2}\right) \qquad [3.57]$$

Al comienzo de este apartado se indicó que una SVM es, en principio, un clasificador binario. Existen algunas aproximaciones al problema multiclase que reformulan las

expresiones anteriores para considerar todas las clases a la vez. Su elevado coste computacional ha llevado, no obstante, a emplear combinaciones de SVMs binarias para abordar el caso multiclase. Existen básicamente dos versiones. La primera, denominada 1-contra el resto, consiste en comparar cada clase con todas las demás, mientras que en la segunda versión cada clase se compara con las restantes de forma separada (1-contra-1).
Aunque en este último caso el número de SVMs binarias es mayor k(k-1)/2 frente a k SVMs, siendo k el número de clases), en [3.52] se adopta esta solución dado que el menor número de vectores de entrenamiento en cada SVM conduce a un menor coste computacional con prestaciones similares. A continuación se aplica un proceso de votación entre todas las SVMs binarias para decidir la etiqueta correspondiente a la muestra de entrada.

Estimación de probabilidades en SVMs. Dependiendo de la solución multiclase que se adopte, las SVMs binarias pueden comparar dos clases entre sí o bien una de ellas con todas las demás. No obstante, por lo general el proceso de obtención de probabilidades a posteriori a partir de las salidas de las SVMs consta en ambos casos de dos pasos: 1) obtener la probabilidad de que la muestra pertenezca a cada clase en todas las SVMs binarias, y 2) transformar estas probabilidades binarias a probabilidades multiclase. En la aproximación 1-contrael resto, este último paso puede reducirse a una simple normalización para que la suma de las probabilidades a posteriori sea uno, ya que el número de SVMs binarias coincide con el número de clases.
La forma más comúnmente empleada para transformar la salida de una SVM en probabilidades binarias consiste en el uso de una función sigmoide, para la que se recomienda ajustar mediante sendas gaussianas las funciones de densidad de probabilidad condicional de la salida de la SVM $\left(p(f|y=+1)\ y\ p(f|y=-1)\right)$. Aplicando la regla de Bayes.

$$P(y=1|f)=\frac{p(f|y=1)P(y=1)}{\sum_{i=-1}^{} p(f|y=i)P(y=i)} \qquad [3.58]$$

y sustituyendo dichas expresiones para las funciones de densidad de probabilidad condicional se llega a:

$$P(y = 1|f) = \frac{1}{1 + exp(af^2 + bf + c)} \qquad 3.59$$

Para simplificar esta función y evitar que no sea monótona, se asume que las gaussianas mencionadas están centradas en los márgenes (±1) y tienen la misma varianza, la cual debe estimarse. En este caso, la expresión de la probabilidad a posteriori se simplifica en una sigmoide, cuya pendiente en la zona lineal se calcula a partir de la varianza de las gaussianas. El sesgo se calcula de forma que P (y = 1|f) = 0,5 en f = 0.

Basándose en este trabajo y asumiendo que en la zona comprendida entre los márgenes las funciones de densidad de probabilidad condicional son aproximadamente exponenciales, Platt propone un modelo paramétrico para la probabilidad binaria a posteriori:

$$P(y = 1|f) = \frac{1}{1 + \exp(Af + b)} \qquad 3.60$$

La diferencia respecto al trabajo anterior consiste en que los parámetros A y B se estiman de manera discriminativa maximizando la verosimilitud.

Esta expresión proporciona directamente probabilidades multiclase en el caso 1-contra-el resto. Si se tienen k clases distintas, la probabilidad a posteriori de la clase i-ésima se puede obtener como:

$$P(y = i|x) = \frac{1}{1 + exp(A_I f_i(x) + B_i)}, \qquad [3.61]$$

siendo f_i (x) la salida de la SVM binaria que clasifica la clase i contra el resto. En este caso no se garantiza que la suma de las probabilidades sea uno; una forma de obtener probabilidades a posteriori normalizadas consiste en usar la función *softmax*:

$$P(y = i|x) = \frac{exp(\gamma f_i(x))}{\sum_{j=1}^{k} exp(\gamma f_j(x))} \quad 3.62$$

donde el parámetro γ se estima maximizando la verosimilitud. En el caso 1-contra-1, en primer lugar se debe calcular la probabilidad de Platt para la muestra de entrada en cada SVM binaria $(i, j), \forall i, j \in \{1, \dots, k\}$.. La probabilidad de Platt de que x pertenezca a la clase i en la SVM binaria (i, j) se calcula como:

$$r_{ij} = P(y = i|y = i \text{ ó } j, x) = \frac{1}{1 + \exp\left(A_{ij} f_{ij}(x) + B_{ij}\right)}$$

$$r_{ij} = P(y = j|y = i \text{ ó } j, x) = 1 - r_{ij}(x), \quad [3.63]$$

siendo $f_{ij}(x)$ la salida de la SVM binaria (i, j). A continuación, se emplea una modificación del método de Refregier-Vallet para transformar estas probabilidades binarias r_{ij} $\forall\, i, j$ en probabilidades multiclase P(y = i|x) $\forall\, i$. Existen referencias [12] donde se resuelve un sistema lineal formado por k − 1 ecuaciones del tipo:

$$r_{ij} P(y = i|x) = r_{ij} P(y = j|x), \quad [3.64]$$

junto con otra que fuerce que la suma de las probabilidades sea uno. La solución que se obtiene depende en gran medida de las ecuaciones seleccionadas, por lo que se propone como alternativa el siguiente problema de minimización, que considera todas las ecuaciones posibles:

$$\min_{p} \frac{1}{2} \sum_{i=1}^{K} \sum_{j:j \neq 1}^{n} \left(r_{ij} P(y = i|x) - r_{ij} P(y = j|x)\right)^2$$

Sujeto a

$$\sum_{i=1}^{k} P(y=i|x) = 1, \quad P(y=i|x) \geq 0 \ \forall \ i, \qquad [3.65]$$

$$\text{Con } P(x) = [P(y=1|x), \ldots, P(y=k|x)].$$

Finalmente, una forma alternativa para obtener las probabilidades a posteriori multiclase cuando la probabilidad a priori es la misma para todas las clases, es:

$$P(y=i|x) = \frac{\prod_{j;j\neq m} r_{mj}}{\sum_{m=1}^{k} \prod_{j;j\neq m} r_{mj}} \qquad [3.66]$$

4-APLICACIONES DE LOS SISTEMAS SIV

El objetivo principal de esta sección es proporcionar un marco representativo de las aplicaciones y tendencias de los sistemas SIV (***Speaker Identification and verification***) desde una perspectiva técnica. En la estructura de esta sección podemos encontrar una indicación de las industrias y los perfiles donde se hace factible y complementario el uso de las tecnologías SIV. Para un mejor entendimiento de este enfoque se han considerado un conjunto de aplicaciones existentes que se han organizado por categorías y ejemplos.

Esta sección se encuentra dividida en las siguientes partes:

- Introducción
- Mercados y perfiles
- Descripción de aplicaciones existentes (Ejemplos).
- Tendencias y aplicaciones del futuro
- Productos y tecnologías

4.1 Introducción

Los ejemplos presentados en esta sección podrían ser usados para acercarnos al contexto de las aplicaciones SIV desde diferentes perspectivas, pues además, se puede comparar y estimar la validez de las prestaciones de los sistemas SIV en la esfera cotidiana y productiva. Por esta razón, las descripciones, de las aplicaciones, que se hacen en este trabajo se validan desde la naturaleza realista de las mismas al mismo tiempo que se citan los factores más relevantes de los PROCESOS SIV.

Las descripciones de las aplicaciones desarrolladas en esta sección se enfocan bajo tres criterios principales y necesarios para el entendimiento de los sistemas SIV. Para una aplicación cualquiera, se describe la designación, autenticación (verificación o identificación) y la inscripción. La parte de autenticación por lo general incluye el manejo de un locutor rechazado. La parte de inscripción incluye la evolución del modelo de voz durante el proceso de entrenamiento y reconocimiento.

Para cada aplicación, se presentan los beneficios y alcances del sistema SIV en la aplicación dada.

Se presenta también una etapa de clasificación donde asignamos el perfil general de las industrias, la categoría de aplicación, la escala de despliegue y el nivel de seguridad en cada aplicación.

En la sección de características técnicas se resaltan rasgos especiales de cada aplicación y los confronta con otro grupo de aplicaciones similares.

Esta sección también incluye aquellas aplicaciones, que según los expertos de este campo, marcarán la tendencia y el futuro de las aplicaciones SIV.

Las aplicaciones aquí consideradas, se han tratado de ordenar jerárquicamente de acuerdo con el grado de importancia de las mismas. Esta valoración se ha establecido de acuerdo con los siguientes criterios.

- SIV sobre sistemas telefónicos
- SIV integrado con unidades de dialogo IVR.
- Aplicaciones simultaneas de SIV con ASR
- Aplicaciones basadas en voiceXML

Si dos o más de estos criterios son verdaderos para una aplicación, se considera de alta importancia, si sólo un criterio es verdadero, la aplicación tiene una calificación media y si ninguno de los criterios es verdadero para una aplicación, entonces a este se le da una calificación, de importancia, VoiceXML baja

4.2 Clasificaciones de las aplicaciones SIV

Hoy podemos encontrar aplicaciones SIV para muchos de los perfiles industriales existentes. La mayoría de estas aplicaciones encajan en una de las categorías principales que se estiman a continuación. No todas estas están basadas en telefonía. Encontramos una variedad de arquitecturas que son la base de las aplicaciones SIV.

Los diferentes componentes de una aplicación SIV pueden ser o estar interconectados sobre una o varias redes.

4.3 Industrias y perfiles

- Telecomunicaciones
- Servicios financieros
- Call centers (*outsourced*)
- Sector penitenciario
- Medios de comunicación
- Sector militar
- Gobiernos (ciencias forenses y ramas del poder)
- Servicios con valor agregado
- Tecnologías de información
- Mercado consumidor
- Asistencia medica
- Industrias de trasporte
- Industria privada
- Grandes organizaciones

4.3 Categorías de aplicación

Las principales categorías y tipos de aplicación son los siguientes

- **Autenticación de transacciones:**

a) Autenticación de auto-administración por teléfono basada en IVR
b) Reset de contraseña por teléfono (un caso especial de autoservicio)
c) Centro de autenticación de llamadas
d) Uso del teléfono en la prisión
e) Compras por teléfono con tarjetas de crédito (autoservicio)
f) Red de tiendas caseras (autoservicio)
g) Centro de autenticación de llamadas para transacciones bancarias.

- **Personalización del diálogo IVR:**

a) Tecnologías de voz dependientes del locutor (p.ej. detección de emociones)
b) Servidores de información dependiente del locutor

- **Recuperación de información:**

a) Información del Cliente para el centro de llamadas
b) Rasgos generales (edad, género)
c) Identificación de individuos específicos
d) Segmentación de discurso

- **Control de Acceso:**

a) Cerraduras manipuladas por voz para acceder a medios físicos (puerta de automóvil)
b) Control de acceso a computadores y redes de información
c) Control sobre fronteras

- **Registro de tiempo y asistencia remota:**

a) Tiempo de registro
b) Casa por cárcel (ajustada a una libertad condicional por verificación)

- **Exploración De audio:**

a) Indexación Automática de una transmisión de audio
b) Locuacidad de llamadas interceptadas

4.4 Aplicaciones existentes (ejemplos)

Esta sección contiene las aplicaciones reales más reconocidas a nivel mundial[34]. La mayor parte de estas, se han registrado recientemente y se han organizado de acuerdo con las categorías anteriores**.** Cada uno de los ejemplos que se describen a continuación se ha enmarcado desde los aspectos fundamentales y útiles para entender todo proceso SIV.

Cuadro 2. Acceso a Detalles del Contrato de Teléfono móvil

Autenticación de transacciones	**Acceso a Detalles del Contrato de Teléfono móvil** **Escenario A (Un equipo un usuario) "Nuance 8.0"**
Definición	Esta aplicación permite a los usuarios de teléfono móvil acceder a los detalles del contrato, con su respectivo operador, de manera automática haciendo uso de la voz.
Designación	En el momento de una llamada, el sistema demandara un PIN que corresponde al móvil desde el que se hace la llamada. Si este se envía y es válido procede con la autenticación si no es rechazado y debe rectificar su marcación
Autenticación	El usuario repite secuencias fijas de dígitos un determinado número de veces. Para evitar fraudes por parte de personas allegadas al usuario el sistema se vale de una comprobación de longitud variable. Para invalidar tentativas de entrar por la fuerza el sistema utiliza un contador que después de un número de fracasos inhabilita todos los intentos provenientes de este número de teléfono.
Inscripción	Para la inscripción el usuario repite secuencias fijas de dígitos un determinado número de veces con el propósito de que el sistema construya un modelo de referencia valido. Existe la opción de poner al sistema en modo adaptación automática, y esto es útil en la actualización de los modelos de referencia.
Características técnicas	• Identidad única (ninguna identificación) • El sistema opera con la concurrencia de SIV+ASR • Opera en modo text promted • El sistema utiliza secuencias de dígitos al azar como frases • El SIV implementado es multisesión • Aplica verificación de longitud variable • Maneja acontecimientos SIV-ESPECÍFICOS: frase inválida, fin de sesión: aceptado, fin de sesión: rechazado • Adaptación automática opcional (sin requerir cambios en las frases) • Inscripción con un número determinado de frases fijas • Acceso a inscripción controlado por PIN • El numero de tentativas fracasadas es limitado

[34] En algunos de los encabezados de los cuadros aparece * el acrónimo de algunas de las compañías o tecnologías directamente relacionadas con la aplicación.

Cuadro 3. Autenticación de la voz para uso directo de Servicios bancarios

Autenticación de transacciones	Autenticación de la voz para uso directo de Servicios bancarios *"Banco de ISRRAEL LEUMI"*
Definición	Los clientes del un Banco son autenticados por verificación de locutores durante conversaciones con un funcionario del centro de servicios. Para esta aplicación se usa la alternativa de preguntas de conocimiento en la autenticación. Los usuarios no se percatan durante una llamada que hay un proceso de verificación o de inscripción, pero a todos los clientes del banco se les notifica por escrito que se harán efectivos tales procedimientos.
Designación	Un Sistema IVR pide al usuario introducir su identificación (única) con tonos DTMF. Después de esto, la respuesta a tal solicitud es transferida al funcionario del centro de llamadas, que se ocupará de sus necesidades bancarias. A través de este procedimiento, el funcionario observa en su pantalla el nombre del usuario, su número de cuenta y estado actual de la muestra.
Autenticación	Cuando una muestra de voz de un usuario se encuentra en la base de datos, entonces el proceso de verificación se realiza durante la conversación entre el usuario y el funcionario del centro de llamada. Siempre que la autenticación sea necesaria durante la llamada, el sistema de Libre-discurso se activa para inferir la cuenta. Si la verificación no es exitosa, y por ende el usuario procesado es rechazado entonces se procede con la alternativa de autenticación basada en conocimiento.
Inscripción	El sistema comprueba automáticamente si el usuario ha terminado la inscripción. Si el formulario de preguntas (de conocimiento) es contestado correctamente, la información de esta llamada será validada para la inscripción. El sistema limita la longitud del segmento para la inscripción. Sin embargo, generalmente son necesarias dos o más llamadas subsecuentes al servicio bancario para tener información suficiente y confiable para crear el modelo de voz. El sistema crea automáticamente un modelo de referencia cuando hay suficiente información. Los modelos de referencia en esta aplicación no se adaptan automáticamente.
Características técnicas	• Industria: de actividades bancarias • Tipo: centro de autentificación de llamadas • Nivel de seguridad: alto • Escala de despliegue: nacional • VoiceXML-Importancia: media
Beneficios	❖ Mayor seguridad en comparación con los sistemas de códigos fijos de verificación. ❖ Sistema más amigable para los usuarios (menos preguntas obligatorias de verificación) ❖ Menor tiempo durante la operación en comparación con las verificaciones de preguntas obligatorias.
Características técnicas	○ Inscripción y verificación a partir del libre discurso ○ Detección de grabaciones a través de operadores ○ Control de acceso a inscripción con preguntas de conocimiento ○ Procesos de inscripción multillamadas ○ Autentificación SIV con retraso en procedimientos de conocimiento ○ sistemas independientes del lenguaje

Cuadro 4. Manejo integral y automático de contraseña

Autenticación de transacciones	**Manejo integral y automático de contraseña. *"Diaphonics"***
Definición	El manejo de la contraseña se ha convertido en una carga significativa para muchas organizaciones, en la práctica, esto se ha venido reflejando en un aumento de los costos y la vulnerabilidad de los sistemas de seguridad. Con la tecnología de verificación de locutores, los usuarios pueden verificar o hacer una nueva inscripción para una nueva contraseña por teléfono, sin involucrar al personal de oficina de la empresa.
Designación	Cuando un usuario requiere una nueva contraseña, él/ella marca el número de teléfono para un reajuste de contraseña, el sistema pide un código de identificación única. El sistema SV utiliza la identidad demandada para recuperar la muestra de voz del usuario. Cuando un usuario inicia el proceso de contraseña automática, el sistema notará que para este usuario no hay muestras de voz en la base de satos y por ende inicia un proceso de inscripción en vez de autenticación.
Autenticación	Una pronunciación es almacenada para la identificación y verificación durante una llamada. Dependiendo de la configuración del sistema, la autenticación se puede hacer a través de frases inducidas dependientes de texto. Las elocuciones se comparan con las muestras de voz del usuario, almacenadas en la base de datos. Si las muestras de voz concuerdan y por ende el locutor es aceptado la contraseña del usuario se reajusta en el sistema de identificación y la nueva contraseña la recibe el usuario por teléfono.
Inscripción y adaptación	Este es uno de los procesos más delicados de esta aplicación, pues, aquí se debe tener certeza sobre la correspondencia entre individuo e identidad. Hay un número de opciones que se pueden emplear para orientar este proceso y el rigor es determinado en gran parte por los requisitos sugeridos por la política de seguridad de la empresa. Varias medidas desde la autentificación se pueden entonces tomar para verificar inicialmente la identidad demandada, incluyendo la comprobación de la línea de teléfono del usuario. Para la inscripción una unidad de dialogo u operador, induce al usuario hablar una serie de dígitos y/o frases dependientes del texto. Con las muestras de voz aducidas se crea una plantilla de voz (voiceprint) que es única para ese usuario. La adaptación se da generalmente después de cada verificación acertada. La otra opción es acercarse al centro de servicios para una renovación de inscripción.
Clasificación	• Industria: aplicación horizontal útil para organizaciones con número de empleados >1000) • Tipo: centro de llamadas/oficina de servicios • Nivel de seguridad : media • Escala de despliegue: media (organizaciones y empresas de gran personal) • VoiceXML-Importancia: alta
Beneficios	❖ Los usuarios no tienen necesidad de acercarse a los centros de servicio de la empresa, pues este es un auto servicio para contraseñas ❖ Disminución de costos y trámites desde los centros de servicio de la empresa ❖ Incremento de la seguridad debido a que en la mayoría de los casos no hay intermediarios en la creación de contraseñas
Características técnicas	○ Identificación automática a partir de pronunciaciones ○ Autentificación sobre frases almacenadas en la designación ○ Verificación dependiente texto ○ Adaptación automática

Cuadro 5. Cerradura de puertas controladas por tecnologías SIV, Acceso a urbanizaciones y conjuntos residenciales control remoto de tiempo y asistencia, Usos personalizados

Control de acceso	**Cerradura de puertas controladas por tecnologías SIV** **Acceso a urbanizaciones y conjuntos residenciales *"Graphco"***
Definición 1	Como su nombre lo indica es una aplicación de la tecnología SIV para el control de dispositivos físicos que permiten el acceso a un espacio específico.
Control de acceso	**control remoto de tiempo y asistencia**
Definición 2	El administrador puede supervisar el estado de residentes por la frecuencia de accesos. Si un residente no hace uso de su voz para tener acceso a una puerta durante un periodo determinado, estos registros pueden ser utilizados por los agentes de seguridad para hacer un seguimiento al residente; esto es útil para verificar si un residente está enfermo, está teniendo un problema o se encuentra fuera de la localidad.
Control de acceso	**Usos personalizados**
Definición 3	Las voces de los miembros individuales de la economía doméstica se pueden marcar con etiquetas circunscritas a las órdenes de trabajo de modo que cuando un ama de casa llega y es autenticada por voz, una orden de trabajo aparezca en el monitor.
Designación	Cada puerta tiene un botón específico para la autentificación del usuario del sistema. Cuando se presiona, el número de identificación del apartamento se transmite a un servidor central que agrupa las muestras de las personas autorizadas con respecto a la hora y fecha de acceso. Si se ingresa un smartcard desde una puerta que está dentro del perímetro, el servidor central comprueba, si el dueño del smartcard tiene derecho o autorización para tener acceso a la puerta específica del perímetro.
Autenticación	Este procedimiento se lleva a cavo mientras el usuario habla libremente aproximadamente 1.5 segundos mientras que mantiene presionado el botón de la puerta. Si la autentificación del locutor falla se puede reintentar o utilizar una llave
Inscripción	Se hace en una oficina especial, no automáticamente en las puertas, pero si con la misma clase de arreglo y micrófono que se encuentra en las puertas. El sistema construye el modelo de referencia a partir de pronunciaciones libres dentro de un tiempo estimado de aproximadamente 3 minutos. Un administrador del sistema, observa la identidad de la persona que es evaluada y confirma los derechos de acceso.
Clasificación	• Industria: urbanizaciones y conjuntos residenciales • Tipo: control de acceso con dispositivos físicos • Nivel de seguridad: medio • Escala de despliegue: grupos restringidos de personas) (residentes, visitantes, encargados del servicio, entre otros) • VoiceXML-Importancia: baja
Beneficios	❖ Mayor comodidad ❖ Mayor seguridad
Características técnicas	○ Inscripción y verificación con pronunciaciones libres (gramática fonética) ○ Verificación dependiente de texto ○ Multiverificación de grupos pequeños ○ El grupo de usuarios autorizados es variable. Es dependiente del tiempo y controlado a través de un servidor central

Cuadro 6. Protección contra robo de vehículos

Control de acceso	Protección contra robo de vehículos *"VOCALSCW1"*
Definición	Es un dispositivo de voz contra el hurto de vehículos y en general el parque automotor. Reconoce hasta 5 conductores autorizados por contraseña de voz. El producto consiste en 2 componentes: el micrófono que es la entrada de la unidad de dialogo o sistema de caracterización, y una unidad de salida que se ubica en cualquier parte de la cabina de manejo. En esta se encuentra una unidad de control con un procesador de voz y actuadores para hacer efectiva la señal de interrupción. Esto Puede ser adaptado en cualquier vehículo. El sistema desconecta 3 circuitos eléctricos que a su vez inhabilitan tres componentes mecánicos fundamentales para el funcionamiento
Designación	No es necesaria una designación para la inscripción, puesto que se realiza sobre todas las muestras de voz (voiceprints) almacenadas. La designación para la inscripción se realiza ingresando uno de los cinco códigos válidos a través del sintonizador. El código es también un identificador del (voiceprint).
Autenticación	Cuando el conductor enciende el automóvil, el sistema emite una señal que el conductor interpreta como la exigibilidad de una contraseña. Si el primer intento de autentificación falla, el locutor (conductor) tiene tres oportunidades adicionales para una verificación correcta.
Inscripción y adaptación	Cuando se enciende el sistema, el conductor ingresa el código asignado. Luego el sistema pide el segundo código, que es el código de entrenamiento, el cual consiste en una palabra de dos a tres silabas que debe repetir en un determinado número de veces en un espacio limitado de tiempo. De esta manera se construye un modelo que es almacenado y utilizado en una verificación inmediata. Para inscripciones posteriores es necesario que la verificación del primero acumule un mínimo de aciertos para la adaptación. Si se está utilizado constantemente el modelo de referencia es conveniente que el grupo de conductores sea el mismo para que no haya necesidad de reinscripción.
Clasificación	• Industria: parque automotor • Tipo: control de acceso con dispositivos físicos • Nivel de seguridad: medio • Escala de despliegue: grupos pequeños • VoiceXML-Importancia: baja • Ventajas: mayor protección contra robo
Beneficios	❖ Mayor protección contra robo
Características técnicas	○ Contraseña de inscripción definida por el locutor (conductor) ○ Inscripción con gramática de fonemas ○ Inscripciones dependientes del texto-) ○ Verificación dependiente texto ○ Códigos para inscripción y reinscripción ○ Identificación por SIV y DTMF ○ Identificación sobre grupos pequeños

Cuadro 7. Control de acceso al computador personal

Control de acceso	Control de acceso al computador personal. *"MAC OS 9 S"*
Definición	El acceso al computador se encuentra limitado por una clave mecanográfica y una frase de paso. Una extensión de esta aplicación es la **Autenticación multibiométrica con cámara USB**
Designación	Cuando el usuario inicia una sesión, este puede iniciar la apertura del sistema con una clave que se puede digitar desde el teclado y luego procede con el procedimiento de autenticación por SV si ya hay un modelo en el sistema. En caso contrario, este puede iniciar un proceso de inscripción después de entrar al sistema por un código mecanografiado.
Autenticación	Una vez se digita la clave de usuario el sistema despliega una ventana que exhibe la frese de paso (según lo especificado durante la inscripción) para la autenticación y una sección que muestra el comportamiento de la señal de voz durante la pronunciación. Esto permite verificar el estado del micrófono y la fidelidad del sistema.
Inscripción	Para la inscripción el sistema despliega una ventana para la configuración o digitación de una frase de paso. Esto se puede configurar para que en cada inicio el sistema muestre la frase o la mantenga oculta. Después de inscribir la frase el sistema está en disposición de recibir las pronunciaciones de la frase un determinado número de veces (4), el comportamiento de la señal se puede observar en una sección de la ventana de inscripción. Después de esto, se hace una autenticación y si esta es exitosa en un mínimo número de veces (3) se finaliza el proceso de inscripción .
Adaptación	En esta aplicación no hay adaptación. Pues el sistema permite la continua reinscripción siempre que se tengan los derechos de usuario para hacerlo. Ya se dijo que después de tres intentos fracasados el usuario puede realizar una reinscripción.
Clasificación	• industria: de los computadores • tipo: control de acceso a computadores • nivel de la seguridad: medio • escala de despliegue: grupos pequeños • Importancia del VoiceXML: baja
Beneficios	❖ La frase de paso se puede exhibir en cada sesión. De este modo nunca se olvidará. ❖ Mayor seguridad. Alguien puede ver la clave del teclado pero no apoderarse de la frase de paso. ❖ Facilita el acceso por parte de personas discapacitadas y complementa el manejo de PCs. Por voz
Características técnicas	○ inscripción por medio de frases fijas ○ inscripción por la gramática del fonema ○ selección opcional de la frase de paso ○ inscripción dependiente de texto (las elocuciones de la inscripción deben ser constantes con respecto a los fonemas) • ○ la verificación del locutor es dependiente del texto en función del modelo. ○ la verificación puede configurarse a texto inducido ○ frases fijas o indirectas a través de una pregunta.

Cuadro 8. Control sobre fronteras

Control de acceso	Control sobre fronteras
Definición	Verificación de locutores para controlar el tránsito por las fronteras
Designación y autenticación	Sobre las fronteras se ejerce un control de entrada y salida a través de un teléfono. Este procedimiento se lleva a cabo a través de un peaje desde el que los conductores alcanzan un teléfono y se identifican. Un operador induce la pronunciación de una frase especifica un determinado número de veces y de esta manera se lleva un registro impreso de tiempo y fecha para cada persona que se inscribe en el sistema.
Inscripción	Este es un sistema que se encuentra vigente en Canadá y Estados Unidos. Para aquellas personas que transitan constantemente las fronteras es necesario un proceso de inscripción. Pues este procedimiento dura alrededor de 10 minutos en una de las oficinas de los peajes de frontera. La inscripción es a través de una frase seleccionada por el conductor y repetida hasta una autenticación exitosa por teléfono.
Clasificación	• Industria : gobierno • Tipo: control sobre frontera • Nivel de seguridad: alto • Escala de despliegue: grupo de tamaño mediano, solamente personas de “bajo riesgo” • Importancia de VoiceXML: baja
Beneficios	❖ Mayor comodidad para el control ❖ Mayor seguridad
Características técnicas	○ Ver protección contra robo de vehículos

Cuadro 9. SIV sobre llamadas interceptadas

Ciencias forenses y rastreo de voz	SIV sobre llamadas interceptadas
Definición	Las tecnologías de verificación e identificación de locutores constituyen en la actualidad una herramienta de gran importancia para los propósitos forenses de las agencias de inteligencia y para las investigaciones donde el seguimiento acústico de la voz de un individuo especifico puede conllevar a una aplicación oportuna de las instituciones y de la ley.
Verificación	Una serie de grabaciones, de llamadas interceptadas, se utiliza para crear el modelo de voz de un individuo específico y para posteriores verificaciones. Este procedimiento es útil para la verificación de un locutor sobre un gran número de llamadas interceptadas.
Características	• verificación sobre pronunciaciones libres • compara la muestra de voz con multiplex llamadas • uso de categorías y listas de posibles individuos
Identificación	La identificación se puede utilizar para identificar un individuo durante una llamada comparando el expediente con un grupo de muestras de voz alistadas.
Inscripción	La inscripción se realiza a través de grabaciones previas del blanco (individuo). La muestra de voz se almacena en una base de datos con la asignación de una identidad única. Esta aplicación generalmente se hace bajo el contexto de la independencia de texto debido a que las pronunciaciones son libres y bajo la integración de un sistema SIV (Identificación y verificación). Para los sistemas SIV de aplicaciones internacionales, en algunos casos se requieren las especificaciones de lenguaje y acento
Características	• Identificación sobre pronunciaciones libres • Identificación sobre grupos de muestras (voces) • Uso de categorías y listas de posibles individuos
Clasificación	❖ Industria: gobierno (policía) ❖ Tipo: forense ❖ Nivel de seguridad: alto ❖ Escala de despliegue: aplicaciones internas, individuos específicos en una nación, y posiblemente por todo el mundo ❖ Importancia VoiceXML: baja
Beneficios	○ Eficiencia en la exploración de grabaciones para determinar la presencia de un individuo especifico. ○ Mayor seguridad y versatilidad para los cuerpos de inteligencia

4.5 Aplicaciones del futuro basadas en autenticación de la voz

Las aplicaciones futuras podrían involucrar tecnologías que todavía no están en el mercado. Las tendencias e ideas, que en la referencia bibliográfica de este apartado se citan, combinan conceptos que necesitan de productos que aun no se encuentran en el mercado para su integración. Pero por la calidad de la fuente se puede asegurar que estas serán las aplicaciones del mañana.

- Combinación de la voz con otros rasgos biométricos, integrados por un teléfono especial o por otros dispositivos de entrada
- Tarjetas inteligentes para el almacenamiento de modelos de voz independientes del formato que utilizan los actuales dispositivos de lectura (ATM)
- Combinación con dispositivos de procesamiento dependientes del locutor (detector de emociones, sistemas de dictado)
- Identificación de locutores sobre grupos numerosos sin demanda de identidad.

Tendencias y aplicaciones factibles

Se incluyen aquellas aplicaciones que los expertos consideran dominarán muchos sectores de la economía y contribuirán al cambio de muchos paradigmas

4.5.1-Control de acceso

- ***Voice-Only-Access Control for ATM.*** En esta aplicación ficticia, propia de los bancos, los ATMs de los bancos pueden leer los modelos de voz que se encuentran en las tarjetas inteligentes de los usuarios para autenticar la transacción del individuo.

- **Control de acceso Multi-Modal para ATM.** En esta aplicación la tarjeta inteligente contiene la huella dactilar y unas muestras de voz del usuario. De esta manera, los ATMs de los bancos, por ejemplo; autentican la transacción del individuo.

- ***PIN-less Call-Center authentication***. Este ejemplo ficticio es similar al desarrollado en el cuadro [1.2] pero en este caso se utiliza la identificación sobre grupos numerosos para eliminar la contraseña de entrada.

- ***Prorotipo de un Elevador***

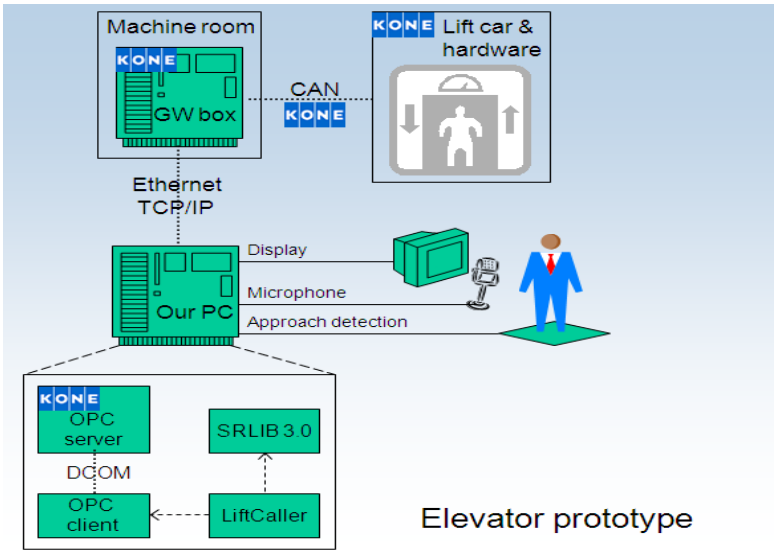


Figura 23 Prototipo de un elevador

4.5.2 Personalización

- *E-Mail sending service*

4.5.3-recuperación de información:

- **Correo de voz etiquetado.** Se trata de la identificación de personas a través de teléfonos móviles con el fin de emprender acciones automatizadas desde el teléfono frente a llamadas específicas. Es un procedimiento que aprovecha el identificador de

llamadas tradicional para realizar inscripciones y posteriormente identificación de personas por voz desde un teléfono móvil. Por ejemplo, cuando una persona esta a la espera de una llamada importante pero se ve impedido en un determinado momento para contestar la llamada, este puede grabar un mensaje que se transmita después de la identificación de la persona.

- **identificación de personas por centros de llamadas.** Esta es una aplicación que se puede implementar de diferentes maneras y bajo distintos propósitos. Por ejemplo, una compañía puede ofrecer información actualizada y confidencial de sus productos a sus clientes. Generalmente, se trata de una unidad de dialogo que está entrenadas para inducir las preguntas relacionadas con una dependencia especifica. Un proceso de autenticación se inicia. El sistema tiene la opción de comunicar con los operadores si no son satisfactorias las respuestas. De igual modo, si la persona que se encuentra al otro lado de la red es identificada y constituye un objetivo, entonces se emprenden otro tipo de acciones. Los expertos enfocan esta aplicación desde la recuperación de información porque se construyen listas de personas de las que se sustrae información cada vez que sean identificadas por el sistema. Por su puesto, que la naturaleza de la información depende del perfil de la aplicación.

- **Segmentación de discurso.** Es una extensión de los ya conocidos sistemas de subtitulado y transcripción pero para esta ficticia aplicación dependiente del locutor. Se proyecta como un sistema capaz de identificar un locutor dentro de una reunión y substraer de este o de varios información

4.6 Productos y tecnologías

Verbio Speaker ID: es un motor de verificación de locutor Agnitio KiVOX integrado con el sistema de reconocimiento del habla Verbio ASR, que permite, tras un breve proceso de

entrenamiento inicial, autenticar o verificar la identidad del locutor implicado en aplicaciones de control, acceso, identificación o seguridad.

ASIS-AGNITO: ASIS permite identificar locutores desconocidos comparando sus voces con un gran repositorio de voces.

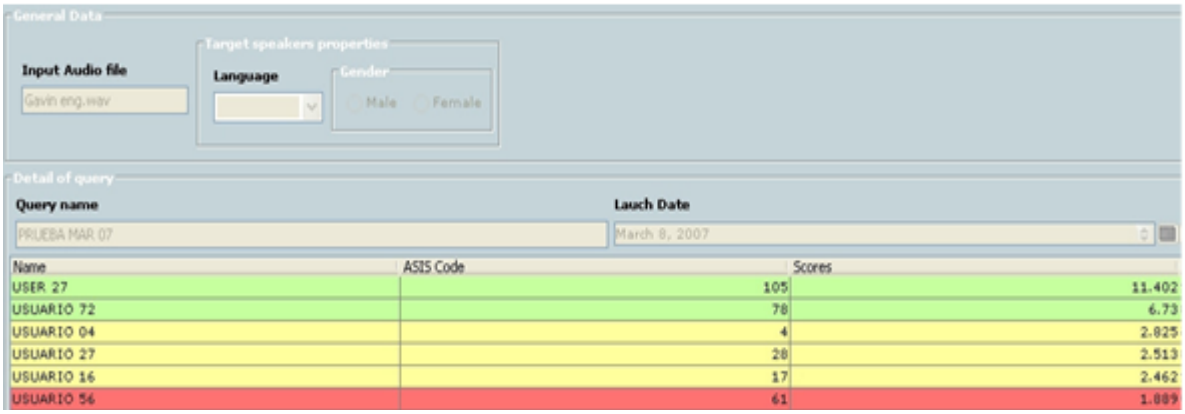
Con este sistema, como ya ocurre con otros datos biométricos, la voz puede ser usada durante una investigación criminal. De manera similar a como ocurre en el conocido sistema de huella AFIS, un modelo de voz de un sospechoso puede ser almacenado en ASIS para futuras identificaciones que se compararán con voces desconocidas grabadas durante la investigación de un nuevo caso criminal.

Algunas de las características que el sistema incluye son:

- Búsquedas masivas y específicas
- Creación y manejo de un número elevado de modelos de voz
- Informes para la administración y control de las operaciones realizadas en el sistema.
- Jerarquía de usuarios.

ASIS genera resultados rápidos y claros de las búsquedas, mostrando en pantalla una lista con los locutores más parecidos clasificados por un código de colores.

Cuadro 10. Representación de una búsqueda ASIS



ASIS BASIC.

- Ideal para pequeñas organizaciones y laboratorios con unos requerimientos de almacenamiento y búsqueda limitados.
- Una única búsqueda simultánea
- Búsqueda sobre 1.500 modelos de voz en un minuto.

ASIS PRO: PRO_{S2} PRO_{S4} & PRO_{S6}

- Ideal para organizaciones medianas y grandes con unos requerimientos de crecimiento y búsqueda altos.
- Escalable desde 2 a 6 búsquedas simultaneas.
- Búsqueda sobre 5.000 modelos de voz en un minuto.
- Se puede personalizar ASIS PRO para integrarlo con otras aplicaciones ya en funcionamiento o conectarlo a bases de datos externas.
- Todas las versiones del sistema incluyen EDIVOX, un sencillo software diseñado para ayudar a los usuarios de ASIS a generar grabaciones de audio listas para ser introducidas en ASIS.

Software White Paper.

Destinado para la verificación automática de locutores, "esta tecnología es de la compañía israelita SentryCom S.A" [35]. El desempeño de esta tecnología se encuentra descrito en el cuadro 4 (manejo integral de contraseña). Basado en plataformas abiertas.

El sistema no es afectado por la nasalización de los sonidos cuando hay un refriado, tampoco es afectado por pronunciaciones defectuosas propias de personas que tienen algún tipo de anomalía. El software, (SentryCom) provee una solución IVR tanto en ingles como

[35] SentryCom S.A. es un Miembro General del La Intel® *Comunications alliance*: una comunidad de las comunicaciones, los diseñadores y proveedores de soluciones.

en hebreo. El software corre sobre servidores que usan procesadores Intel y sistema operativo. Windows

Loquendo Speaker Verification. La tecnología **Speaker Verification** de Loquendo, utiliza el timbre vocal como una medida biométrica para verificar y comprobar la identidad de un individuo. "Las ventajas son múltiples: la autenticación de la voz es una tecnología versátil, fácil de utilizar y no es intrusiva, por lo tanto es aceptada fácilmente por los usuarios. Respecto a otras tecnologías biométricas, es exacta y no requiere el uso de equipamientos específicos. "Basta la voz"[36].

Funcionamiento. El proceso de autenticación de la voz está compuesto por dos fases distintas:

- Una primera **fase de registro** a los servicios, que tiene como objetivo capturar y memorizar el timbre vocal del locutor, se realiza solo una vez al inicio, pidiendo al usuario que pronuncie algunas palabras.
- **Verificación** - el usuario puede ser aceptado, rechazado o identificado. Con la tecnología de **Identificación** - la identidad del usuario se detecta automáticamente. La voz que está siendo verificada o identificada se compara con los modelos previamente adquiridos.

Características generales

- Loquendo Speaker Verification complementa la potencia de Loquendo ASR.

Las alternativas de diferentes modalidades operativas:

- Text-dependent: la comprobación se lleva a cabo con un texto específico, que puede ser sugerido por el sistema o escogido por el usuario;

[36] www.loquendo.com. Vocal technology and services. Loquendo es una compañía perteneciente al Grupo Telecom Italia situada en Turín, Italia. Es unas de las mejores compañías para las tecnologías de voz, tanto en Europa, como en EEUU y en Sudamérica.

- Text-prompted: el sistema proporciona un texto“aleatorio” a repetir por el usuario. Este modo excluye cualquier posibilidad de acceso fraudulento por medio de grabaciones, etc.
- Free Speech: el usuario es libre de decir todo lo que desee.

Comprobación del significado como seguridad agregada y personalización:

- Loquendo SV influencia los modelos acústicos del Loquendo ASR (Automatic Speech Recognition) para garantizar el matching semántico: el mecanismo de comprobación del significado verifica las características de la voz del usuario y el contenido de la contraseña vocal. Esta combinación aumenta considerablemente los niveles de seguridad generales del sistema.
- El campo de contraseñas vocales aplicables se especifica junto con las gramáticas pertinentes del reconocimiento, lo que permite la personalización sumamente precisa de la aplicación.

El sistema de control y los instrumentos de calibración cubren los requisitos específicos de seguridad de cada aplicación. Completamente flexible y personalizable ya que permite definir:

- El número de repeticiones y el contenido en la fase de la instrucción
- El número de repeticiones en la fase de comprobación.

NUANCE. Es una tecnología especializada en el reconocimiento de voz natural pero también a desarrollado otras unidades para el reconocimiento de locutores que se integran con las unidades de reconocimiento de habla. Para el reconocimiento de locutores utiliza tanto el perfil de identificación como el de verificación. En el cuadro 2 se describe una de las aplicaciones del mundo real en la versión de NUANCE 8.0

CONCLUSIONES

El reconocimiento automático de locutores constituye una tarea que ha evolucionado científica y tecnológicamente. Pues su enfoque sugiere la participación e integración de muchas áreas del conocimiento. Esta multidisciplinariedad hace de este problema un campo científico interesante y vemos que esta es quizás una de las justificaciones de los distintos enfoques desde los que se ha tratado de responder el interrogante del reconocimiento de locutores. Hoy quizás sea un poco temprano para afirmar que por mucho que se evolucione alrededor del tema siempre se estará pensar en encontrar una mejor manera de resolver el problema de identificación de una manera global. Aquí es importante resaltar que emular el comportamiento del ser humano en función del reconocimiento puede ser una salida importante. Ya se ha concluido que la voz es uno de los tantos rasgos biométricos que utiliza el hombre para la identificación natural de las otras personas que le rodean. Por esto, se piensa en la creación de sistemas integrados donde converjan reconocedores de distintos rasgos biométricos, el problema de reconocimiento es para los humanos un problema menor, precisamente por la integración biométrica de los rasgos de la entidad a reconocer. Aun para los seres humanos la voz como rasgo biométrico, es a veces insuficiente para el reconocimiento. Debemos recordar que este se vale de otros soportes para la validación de una entidad. En el caso de las personas puede ser el rostro, el tamaño y en realidad un sin numero de características biométricas. Por esta razón, el reconocimiento automático del locutor (automatic speaker recognition, ASR) representa actualmente una tarea clave dentro del reconocimiento biométrico. Además representa una amplia línea abierta de investigación en el tratamiento digital de señal que comparte numerosos elementos con la tecnología del habla.

En este trabajo se ha hecho un recorrido bastante amplio sobre las técnicas para la caracterización de la voz. Una vez más, se ha estimado que un análisis biológico puede brindar resultados positivos en pos de encontrar una técnica eficiente. El estado del arte en

este campo sigue abierto. Si bien se ha integrado el análisis con bancos de filtros orientados perceptualmente como respuesta al suscitado funcionamiento del oído es bien sabido que no hay una conformidad absoluta en cuanto al aporte de estas técnicas. Por otra parte, se siguen utilizando las técnicas tradicionales y en realidad sirven para resolver el reconocimiento pero bajo una carta de restricciones. Desde esta instancia es importante saber que el análisis biológico del reconocimiento no termina en el oído aun el cerebro sigue siendo una caja negra frente a la intensión de leer de este comportamientos específicos. Esto concuerda en este caso con el denominado sistema ASR que actúa como un clasificador de patrones. Cada patrón está formado por el conjunto de características o parámetros, extraídos de una determinada locución y es "enfrentado" o comparado con distintos modelos generados para cada locutor.

La salida del clasificador ofrece una verosimilitud o una medida de distancia, entre el patrón de entrada y el modelo; y en última instancia una decisión, basada en un umbral, que clasifica la locución como perteneciente a un determinado locutor.

Cada modelo de un locutor es generado mediante patrones extraídos de locuciones del mismo; siendo necesario que cada locutor involucrado en el sistema, disponga de su propio conjunto de datos de entrenamiento. Este conjunto será distinto del conjunto de datos sobre los cuales se pruebe el sistema. Este es el referente que nos permite concluir que en materia de reconocimiento hay mucho por explorar. Si suponemos que las técnicas de caracterización del estado del arte son suficientes seguramente es excluyente la suposición en relación con las técnicas de clasificación.

Cuando abordamos las técnicas de clasificación vemos que en su integración constituyen un recurso bastante limitado. El hecho de que un sistema debe ser capaz de expandirse en la medida que aparezcan nuevos individuos para el reconocimiento. Ósea otro que ingrese nuevos rasgos, implica que los sistemas de reconocimiento que se construyen en la actualidad se enfocan para un grupo restringido de personas. Esto lo podemos medir fácilmente toman como referencia técnicas de clasificación como los arboles de decisión, redes neuronales etc.

Estas técnicas sugieres que debe haber una unidad de clasificación por cada individuo a reconocer. De esta manera se concluye que la demanda computacional derivada de estas técnicas es alta y que una de las soluciones mas importantes de frente a los grupos extensibles concuerda con el desarrollo de sistemas modulares.

Para los seres humanos la suficiencia de la voz como rasgo biométrico para el reconocimiento es casi que absoluto. Pero para las maquinas de reconocimiento no. Por esta razón, la discriminabilidad de los sistemas artificiales se calibra de acuerdo a un umbral de decisión que compensa la salida de los posibles errores. Para mejorar el rendimiento de estos sistemas se proponen tres contextos alrededor del texto. Por ejemplo, el rendimiento de los sistemas que utilizan reconocimiento *dependiente del texto* es mayor, sin embargo son más vulnerables cuando la clave es conocida por personas ajenas a su propietario. El riesgo de la clave es evitado usando independencia del texto, pero el rendimiento de los sistemas que utilizan esta opción es menor, y además, el peligro radica en que cualquier grabación de la voz de una persona puede ser utilizada para un acceso no permitido. Los sistemas *independientes del texto* necesitan también más entrenamiento que los *dependientes del texto* ya que las características específicas de la frase no están disponibles. Una de las conclusiones del estado del arte es que una alternativa intermedia (texto solicitado) pude resolver las dificultades de los dos perfiles principales. Por otra parte se ha pensado que la información presente en diferentes niveles de información puede ser útil para enfrentar el problema de reconocimiento de una manera más global. Desde esta instancia la inclusión de rasgos físicos y aprendidos pueden acercar a los sistemas de reconocimiento artificiales a un mayor despliegue de experimentaciones y aplicaciones. En conclusión, la voz es un rasgo biométrico que ofrece muchos grados de libertad sobre el que se pueden hacer fusiones de datos provenientes de los diferentes niveles de información.

BIBLIOGRAFÍA

[1] JANER GARCIA, Leonard. Transformada Wavelet Aplicada a la Extracción de Características en señales de voz. Barcelona 1998. 215 h. Tesis Doctoral. Universidad politécnica de Cataluña. Departamento de teoría de señales y telecomunicaciones.

[2] DOCÍO FERNÁNDEZ, Laura. Aportaciones a los sistemas de reconocimiento. Vigo 2001, 294 h. Tesis Doctoral. Universidad de Vigo. Departamento de tecnología de comunicaciones.

[3] MACIAS GUARASA, Javier. Arquitecturas y métodos en sistemas de reconocimiento automático de habla de gran vocabulario. Madrid 2001, 232 h. Tesis Doctoral. Universidad politécnica de Madrid. Escuela técnica superior de ingenieros de Telecomunicaciones.

[4] MILONE HURTANI, Diego. Información acentual para el reconocimiento automático del habla. Granada 2003, 268 h. Tesis Doctoral. Universidad de granada. Departamento de electrónica y tecnología de computadores.

[5] FAÚNDEZ ZANUY, Marcos. Modelado predictivo no lineal de la señal de voz aplicado a codificación y reconocimiento de locutor. Cataluña 1998, 161 h. Tesis Doctoral. Universidad de Cataluña. Departamento de teoría de señales y telecomunicaciones.

[6] VILLAMIL ESPINOZA, Iván. Aplicaciones en reconocimiento de voz utilizando HTK. Santa Fe de Bogotá 2005. Trabajo de grado (maestría en ingeniería electrónica). Universidad Javeriana. Facultad de ingeniería electrónica.

[7] CONTRERAS ORTIZ, Sonia E. Detección activa de voz en tiempo real orientada a la clasificación de fonemas aislados. Bucaramanga 2003, 81 h. Trabajo de grado (magister en ingeniería). Universidad Industrial de Santander. Facultad de ingenierías físico-mecánicas.

[8] CZECH, Christyan; MIODOWNIK, Fabián y RABASCHIO, Alexis. Reconocimiento de locutores a partir de archivos en formato MP3. Madrid 2005, 93 h. trabajo de grado (ingeniería electrónica). Universidad politécnica de Madrid. Facultad de ingeniería electrónica.

[9] VIVARACHO PASCUAL, Carlos y ALONSO ROMERO, Luis. Redes neuronales en el reconocimiento de locutores. Valladolid 2003, 108 h. Monografía. Universidad de Valladolid. Facultad de ingeniería electrónica.

[10] FAÚNDEZ ZANUY, Marcos. State of the art in speaker recognition IEEE and E. systems Magazine, (Mayo 2005).

[11] JOSEPH P, Campbell, Speaker recognition: a tutorial. IEEE Vol. 85, No. 9 (septiembre 1997).

[12] REYNOLDS A. Douglas, an overview of automatic speaker recognition technology. IEEE (febrero 2002).

[13] ATAL S. Bishnu. Automatic recognitions or speakers from their voices. IEEE, Vol. 64, No. 4 (abril de 1976).

[14] W. M, Campbell. A SVM / HMM SYSTEM for speaker recognition. IEEE, (Marzo de 2003).

[15] SIEW CHAN, Woo; CHEEP ENG, Lim y R, Osman. Development of a speaker recognition system using wavelets and artificial neural networks. Proceedings of 2001 International Symposium on Intelligent Multimedia, video and Speech Processing (Mayo 24 2001 Hong Kong)

[16] SIEW CHAN, Woo; CHEEP ENG, Lim y R, Osman. Text dependent speaker recognition using the fuzzy ARTMAP neural networks. Proceedings of 2000 International Symposium (School of Distance Education. Universiti Sains Malaysia)

[17] RABINER L., JUANG B. H. Fundamentals of Speech Recognition. Signal Processing Series. Prentice-Hall, 1993.

[18] FARRELL, Kevin y MAMMONE, Richard. Speaker recognition using neural networks and conventional classifiers. IEEE Vol. 2, No. 1, (mayo de 1994).

[19] BOVES, Lou y ELS, Den Os. Speaker recognition in telecom applications IEEE, Vol. 5 No. 2, (junio de 1998)

[20] ZAMALLOA, Maidez; RODRIGUEZ, Luis J. y BORDEL, Germán. Selección y pesado de parámetros acústicos mediante algoritmos genéticos. IV jornada en tecnologías de del habla 8-10 (noviembre de 2006).

[21] GARCIA LÓPEZ, Francisco y FAUNDEZ ZANUY, Marcos. Reconocimiento de locutores independiente de texto en ambientes ruidosos. Universidad Politécnica de Mataró Vol. 3, No 2, (agosto de 2002)

[22] FERNÁNDEZ, Merlo. Reconocimiento de voz mediante una red neuronal de Kohonen. Universidad de buenos Aires 2003

[23] FAUNDEZ ZANUY, Marcos. Efectos de la extensión del ancho de banda en el reconocimiento de locutor. Proyecto Europeo Cost, 277, (mayo de 2000).

[24] DABOUL, Claudia y ECKERT, Martin. Speaker identification and verification applications. Voice XML-Forum, Speaker Biometric Committee, (mayo de 2006)

[25] FUENTES BUENO, Virginia; GONZÁLEZ CARRASCO, Israel y RUIZ MEZCUA, Belen. Subtitulado en tiempo real. Sistemas y tecnologías. Madrid, Universidad Carlos III, (Marzo 2006).

[26] MARKOWITZ, Judith. Hands on with the US Immigration and Naturalization Service, VoiceID Quarterly, Vol. 2, No 2, April 1998, p1-5