UNIVERSIDAD NACIONAL AUTONOMA DE MEXICO

Facultad de Ingeniería

Reconocimiento Automático de voz Usando LPC y Transformada KLT, Aplicando Técnicas de Ajuste Dinámico de Tiempo

T E S I S

Que para obtener el título de

INGENIERO EN TELECOMUNICACIONES

p r e s e n t a

RICARDO IBARRA SALINAS

Director de Tesis: M. I. Abel Herrera C.

México, D. F. 1998

A mis padres:

Dr. Ricardo Armando Ibarra Ríos
Enf. Ruth María Salinas Ochoa

Quienes a lo largo de mi vida, siempre han estado a mi lado
con su amor, cariño, apoyo, experiencia y tantas cosas
que nunca terminaría de escribir.

A mi hermano:

Gabriel Ibarra Salinas

Con quien he compartido y recibido su alegría

Agradecimientos

Al M.I. Abel Herrera Camacho, por su apoyo y amistad.

A mis abuelitas: María del Refugio Ríos y Felicitas Ochoa.

A mi tío Gabriel Arturo Ibarra Ríos, por su invaluable apoyo.

A mi tía Esperanza Ríos Farías, por su entusiasmo y cariño.

A María Elena Martínez Cortés, por su cariño y apoyo en tantos momentos.

A mis amigos: Arturo Gardida, Juan Luis Reyes, Rocío Mendoza, Arturo Toledo, Fernando "Chapo", Josué Ortiz, Mariela, Leticia, Hugo, Miguel, Juan Carlos, Leonardo Canseco con quienes he compartido mucho de mi vida.

A mis familiares: Miguel Angel Salinas y Blanca Zavala, Juvencio Salinas, por su entusiasmo, apoyo, y consejo.

A mis maestros, en especial a: Ing. Roberto Macías, M.C. Salvador Landeros, Ing. Gabriel Jaramillo, Ing. Jesús Reyes, Ing. Carlos Castillo, por compartirme sus conocimientos, dedicación, experiencia, y con su ejemplo me impulsaron en muchas ocasiones a seguir adelante.

A la familia Telerín: Sergio, Mauricio, Alex, Javier, Checho, Juan, con quienes compartí la alegría de aprender.

A las familias: Reyes Ramírez y Martínez Cortés.

A la Facultad de Ingeniería y a la UNAM, por todos los conocimientos y las oportunidades que me ha brindado.

ÍNDICE

Prefacio

Uno de los sueños de los seres humanos ha sido transformar su entorno para convertirlo en un mundo que le permita desarrollar sus habilidades, crear un mundo distinto y dedicarse a virtudes que enaltezcan su espíritu.

En esa búsqueda hemos soñado con máquinas y autómatas, que desarrollen las labores cotidianas. Poco a poco los cuentos de ciencia ficción se vuelven realidad, existen grandes avances tecnológicos y es inacabable el número de cambios que el hombre puede realizar.

El reconocimiento automático de voz, es uno de tantos sueños, y con un poco de imaginación podemos encontrar un sin número de aplicaciones, un paso a una nueva forma de comunicación. Ahora las máquinas nos pueden escuchar y entender, dándóles así un toque más humano.

En mi tesis, he encontrado muchas sorpresas, algo tan cotidiano como escuchar una conversación, se convierte en un reto. Entre filtros, transformadas, algoritmos, programas, y teoría, se devela no sólo ese sueño de ser escuchado por un autómata, sino el milagro de ser humano y poseer habilidades difíciles de implementar.

Todavía falta mucho por hacer en esta área en investigación y la presente es tan sólo una parte de ese sueño. Constituye un sistema basado en muchos trabajos, gran parte de la información no está contenida en libros de texto, dejando paso a la innovación de ideas. No existen algoritmos de reconocimiento automático que sean totalmente exactos. Para el desarrollo de sistemas de este tipo se requiere de todo lo aprendido, y mucha paciencia para experimentar, y meses de programación. Pero después de todo la recompensa es grande, contribuir al desarrollo de este sueño.

El tema que se expone consta del planteamiento de dos sistemas de reconocimiento uno de ellos usando coeficientes de predicción lineal y otro usando la transformada KLT, ambos con características especiales que se abordan en cada sección.

Las ecuaciones, figuras y tablas que se mencionan en el texto están numeradas con el capítulo al cual corresponden y un número secuencial que comienza en cada uno. Al referirlas en el texto se indica únicamente el número secuencial, significando que esta ecuación está en el mismo capítulo.

La referencia bibliográfica en el texto se encuentra indicada por corchetes [ ]

Todos los programas fueron elaborados en una computadora personal usando Turbo C o Turbo Pascal, para algunas gráficas se uso Matlab 4.0. El sistema se ejecuta en una PC, convencional, sin mayor requerimiento que una tarjeta de sonido, aún cuando falta por hacer que dicho sistema tenga un perfil comercial, constituye una prueba de los algoritmos involucrados en el reconocimiento automático de voz. Como otra característica principal es que los excelentes resultados así como la velocidad de reconocimiento lograda, han dado la libertad de intercambiar el idioma y observar inclusive mejoras en el español.

I. Introducción

I.1 Objetivo

Hacer un estudio, diseñar e implementar dos sistemas de reconocimiento automático de voz, de palabras aisladas, en una computadora personal. Probar sus capacidades y generar un sistema que reconozca palabras en el idioma inglés y español, dichas por un usuario.

I.2 Problemas en un sistema de reconocimiento de voz

Actualmente el área de reconocimiento automático de voz es un área en investigación, por lo cual no existen metodologías bien definidas, que garanticen resultados. Debido a lo anterior se requiere de varios conocimientos en el área de procesamiento digital de señales, técnicas de programación y de experimentación para la generación de un sistema de reconocimiento.

Un sistema reconocimiento de palabras aisladas es considerado como el nivel más básico de implementar, a continuación presento algunos de los problemas que se enfrentan en esta área.[9] [18]

I.2.1 Locutores distintos

Existen variaciones de uno a otro locutor en la pronunciación, de tal forma que la señal de voz contiene información dependiente del locutor. Cada persona tiene diferentes formas de expresión, acentuaciones particulares, tono y timbre distintos en la pronunciación.

I.2.2 Ambigüedad

Las variables acústicas no son mapeadas uno a uno en variables fonéticas, variando además de un idioma a otro. (Ejemplo: Cuando alguien dice un nombre propio, a veces pedimos nos sea deletreado). Tal es el caso de los alófonos, que son variantes de los fonemas se escuchan pero no se escriben.

I.2.3 Variaciones del locutor

Aún cuando una misma persona pronuncie una palabra, existen variaciones, causadas por:

1) Descuido: Los parlantes cortan palabras, frecuentemente las palabras de corta duración son pronunciadas tan rápido que son más bien golpes en la garganta, más que fonemas.

2) Variaciones fonéticas. Las frecuencias formantes, así como frecuencia fundamental, y duración de las transiciones en los formantes, cambian con la edad.

3) Coarticulación: Las características fonéticas de los sonidos de voz son afectados por el contexto.

4) Estados Anímicos del parlante: Las situaciones de tranquilidad o estrés producen variaciones importantes en la pronunciación de palabras.

5) Respiraciones, chasquidos, en la pronunciación.

6) Baja claridad en el parlante.

I.2.4 Duración en el tiempo de palabras del mismo parlante o distinto

La velocidad con la cual se pronuncia la misma palabra, raramente es igual, depende del parlante. Incluso una misma persona difícilmente podría pronunciar la misma palabra, dos veces con la misma duración.

I.2.5 Diferencias en sexos

Las frecuencias formantes varían con el sexo. Es conocido por experimentos que son más eficientes los sistemas de reconocimiento de voz para mujeres que para hombres. Se ha observado que las voces pertenecientes a mujeres tienen mejor definición de las frecuencias fundamentales y mayor claridad armónica.

I.2.6 Ruido e Interferencias

Los humanos tienen gran habilidad para reconocer la voz, aún en ambientes extremadamente ruidosos. Sin embargo al implementar un sistema físico nos enfrentamos a serios problemas

que afectan considerablemente a los sistemas de reconocimiento hasta el momento. Algunos de los tipos de ruido que afectan a un sistema de reconocimiento, son los siguientes:

1) **Ruido Blanco:** Producido por circuitos electrónicos
2) **Ruido Impulsivo**: Producido por cambios de estado del sistema, (prender o apagar el micrófono, lectura de disco duro, inducción de ruido del bus de datos)
3) **Ruido Ambiental**: Producido por el entorno, (resonancias, ecos, viento, parlantes ajenos, etc.)
4) **Ruido del equipo**: En esta categoría se agrupan (retroalimentación, ventilador de la computadora, corriente eléctrica).
5) **Ruido de cuantización**: Generado por el proceso cuantización, en el cual se aproxima el valor de una muestra al nivel más cercano de cuantización.

De los anteriores los más dañinos al sistema son: Impulsivo, Ambiental, ya que estos en algunos casos invaden el ancho de banda utilizado en el reconocimiento, y su presencia es impredecible. Los no mencionados como dañinos, influyen en el reconocimiento, pero son manejables, si se utiliza una base generada en el sistema, en el cual se reconoce, digamos que todas las repeticiones contienen algo en común.

I.2.7 Incapacidad de realizar un modelo eficiente y completo del sistema auditivo humano

Hasta el momento no existe un sistema auditivo electrónico, como el humano. Existen modelos para cada parte del oído y se ha trabajado mucho sobre las funciones de transferencia del oído. Sin embargo muchos de los procesos que se llevan a cabo en los humanos, están en investigación. Existen modelos muy elaborados que tratan sobre las cavidades del oído, avances en fisiología humana en la percepción y teorías sobre acústica, sin embargo aún quedan muchos misterios por resolver, caso a tratar el reconocimiento y el tipo de procesamiento que lleva la voz llegar al cerebro humano.

I.2.8 Tiempo de reconocimiento

Podemos implementar sistemas más robustos, usando modelos del oído, reducción de ruido, métodos numéricos, para tener un sistema con alto nivel de reconocimiento. Sin embargo entre más complejo sea el sistema en tiempo de reconocimiento aumenta considerablemente, debido al número de cálculos involucrados.

I.2.9 Implementación

La mayor parte de los algoritmos que se emplean en procesamiento de señales digitales, requieren, gran cantidad de código de programación. El manejo de señales digitales se ve seriamente limitado por la capacidad de memoria de y cálculo de las computadoras existentes. Los programas que se han venido desarrollando en estaciones de trabajo, y el uso de una computadora personal requiere; de técnicas de programación que sean eficientes y orientadas a la mayor compresión de datos buscando el menor uso de recursos de computo.

I.3 Tipos de sistemas reconocimiento automáticos

Se dividen en las siguientes categorías mostradas en la (Tabla 1), incrementando su complejidad por niveles. [2]

NIVEL	TIPO DE SISTEMA	CARACTERÍSITICAS
1	PALABRAS AISLADAS	Reconoce palabras separadas por pausas.
2	DETECTOR DE PALABRAS	Reconoce una palabra específica, en una conversación.
3	PALABRAS CONECTADAS	Reconoce palabras sin pausas entre ellas.
4	ENTENDIMIENTO DE LENGUAJE	Usa un banco de información del lenguaje, idealmente, debe entender el significado de la frase.

Tabla I-1 Características de los sistemas de reconocimiento

El sistema que trataremos es de primer nivel, y requiere de una base de referencias para reconocer el patrón de entrada

II. Antecedentes

II.1 Probabilidad y estadística

En nuestro sistema consideraremos las señales de voz como procesos aleatorios, es por ello que es importante recordar alguna propiedades de los procesos aleatorios y a su vez definir las notaciones que se seguirán en el texto. Usando los métodos de estimación de parámetros que se describen en estadística se tienen las siguientes definiciones [16].

II.1.1 Estimadores

a) Media de un de un procesos aleatorios ergódico.

La media en las señales de voz se considera cero, debido a que no trabajamos con offsets de voltaje directo.

Ec. II-1

$$\tilde{\mu}_X = \bar{X} = \frac{1}{n}\sum_{i=1}^{n} X_i$$

b) Varianza de un variable aleatoria que representa a una señal, como es sabido corresponde a la potencia promedio de una señal.

Ec. II-2

$$\tilde{\sigma}_X = \frac{1}{n}\sum_{i=1}^{n}(X_i - \bar{X})^2$$

c) Covarianza entre dos variables aleatorias

Ec. II-3

$$\tilde{\sigma}_{XY} = \frac{1}{n}\sum_{i=1}^{n}(X_i - \bar{X})(Y_i - \bar{Y})$$

II.1.2 Vectores de variables aleatorias

Un vector de variables aleatorias, esta constituido por un conjunto de variables aleatorias, y nos sirve para analizar el comportamiento que existe entre un número 'm' de variables aleatorias. Supóngase un vector de m, variables aleatorias.

Ec. II-4

$$X = \begin{bmatrix} X_1(n) \\ X_2(n) \\ X_3(n) \\ \cdots\cdots \\ X_m(n) \end{bmatrix}$$

b) La matriz de autocovarianza

Esta definida por

Ec. II-5

$$\sum X = E\{XX^T\} - \mu_X \mu_X{}^T$$

Expandiendo la expresión anterior vemos que la diagonal principal se compone de la varianzas de cada variable aleatoria, fuera de ella se encuentran las covarianzas.

Ec. II-6

$$\sum X = \begin{bmatrix} \sigma_{X_1X_1} & \sigma_{X_1X_2} & \cdots\cdots & \sigma_{X_1X_m} \\ \sigma_{X_2X_1} & \sigma_{X_2X_2} & \cdots\cdots & \sigma_{X_2X_m} \\ \vdots & \vdots & & \vdots \\ \sigma_{X_mX_1} & \sigma_{X_mX_2} & \cdots\cdots & \sigma_{X_mX_m} \end{bmatrix}$$

II.2 Procesamiento Digital de Señales

II.2.1 Muestreo

El proceso de muestreo lo podemos ver como una forma de discretizar en el tiempo una señal continua en tiempo. Consiste en tomar muestras de una señal analógica, en intervalos de tiempo definidos, por lo general estos intervalos son iguales. Existe una característica muy importante que no podemos pasar por alto, la frecuencia de muestreo y su impacto sobre la señal con la que trabajamos.

Sabemos que el teorema de Nyquist, establece que la frecuencia de muestreo debe ser al menos del doble de la frecuencia máxima de la señal, para que evitemos efectos de aliasing (traslape) en el dominio de la frecuencia, y podamos recuperar la señal.[13]

El teorema anterior establece un concepto antitraslapes, pero de ninguna manera establece la calidad con la que la señal es muestreada. Algo que debe tenerse en cuenta es que la señal muestreada, no representan de manera única a una señal. Entonces la calidad con la cual podemos reconstruir la señal depende de la frecuencia de muestreo, ya que nos acercaremos más a la señal analógica. Esta frecuencia de muestreo a sido determinada para voz como mínimo, 8000Hz, considerando que el ancho de banda de voz (300-3400Hz) después de ser procesado permita una señal entendible. Las tarjetas de sonido comerciales para computadoras personales manejan frecuencias de muestreo (8 kHz, 11.025 kHz, 22.050 kHz, 44.100kHz).

II.2.2 Cuantización

Después de ser muestreada la señal debemos cuantizarla para poder representarla con un número finito de niveles. La cuantización usada en nuestro caso es lineal (todos los niveles equidistan entre sí). Es sabido de acuerdo a los estudios realizados en acústica, que el oído humano que distingue alrededor de 280 niveles de intensidad. [19] Por lo tanto al cuantizar podemos recurrir a usar tan sólo 256 niveles, si los usáramos todos. Es por ello que existen en el mercado tarjetas de sonido de 8 bits, para codificar la cuantización. Sin embargo previendo

fenómenos de distorsión, pérdida de señales de baja intensidad, motivos experimentales, mejor relación señal a ruido, y compatibilidad numérica, se utilizan de 16 bits, (65536 niveles).

II.2.3 Análisis de señales en tiempo corto

II.2.3.1 Ventaneo de señales

Las señales de voz tienen variaciones importantes en el tiempo, por lo cual es necesario utilizar ventanas para analizar el comportamiento de la señal en un intervalo de tiempo, y no en su totalidad [9]. De manera que el ventaneo de una señal, consiste en tomar sólo algunas muestras de la señal en un intervalo de tiempo. Matemáticamente puede expresarse como:

Ec. II-7

$$Q_n = \sum_{m=-\infty}^{\infty} T[x(m)]w(n-m)$$

Donde la ecuación anterior es la base para el análisis en tiempo corto, Q(n) es la señal ventaneada, la T[ ] es cualquier transformación que efectuemos sobre la señal, y w(n) es una función de peso que sirve para evitar los efectos de suponer una porción de señal.

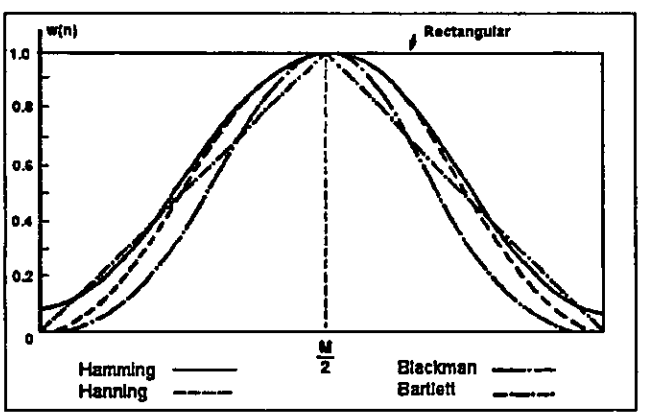

Fig. II-1 Principales ventanas de análisis en tiempo corto

Existen distintos tipos de ventanas, que se utilizan dependiendo de las características de la señal. Algunas de ellas se muestran en la (Fig 1), Hamming, Hanning, Blackman, Bartlett.[13]

Otro parámetro empleado es el traslape entre ventanas que sirve para evitar perdida de información entre ventanas, este parámetro según los autores [22], [9] es del 10% de la duración de una ventana.

El ventaneo tiene distintos fines, actúa como un filtro paso bajas, pero la aplicación más importante es evitar resultados como: la energía de fuga al obtener la transformada de Fourier o perder características de la señal por considerar sólo una parte. La determinación de la ventana que vamos a utilizar depende de las características de la señal.

La ventana que se eligió para los sistemas es la ventana de Hamming, debido a las características de voz, y a que es una ventana con mayor atenuación en los lóbulos laterales.

Ec. II-8

$$w[n] = \begin{cases} 0.54 - 0.46\cos(2\pi\, n / M) & 0 \le n \le M \\ 0 & otro \end{cases}$$

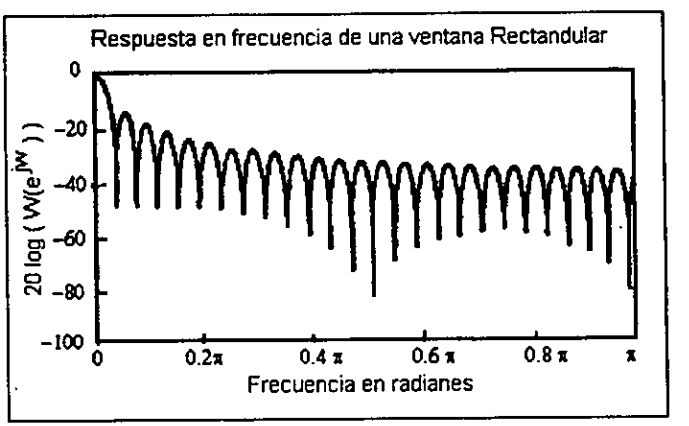

Fig. II-2 Respuesta en frecuencia de una ventana Rectangular

En la (Fig. 2), vemos la respuesta de una ventana rectangular, se observa como los lóbulos laterales son mayores que para una ventana Hamming como la (Fig. 3). Si realizamos una transformada de Fourier con ventana de Hamming, tenemos mayor energía de fuga al obtener su espectro, es decir la energía se pierde en los lóbulos laterales.

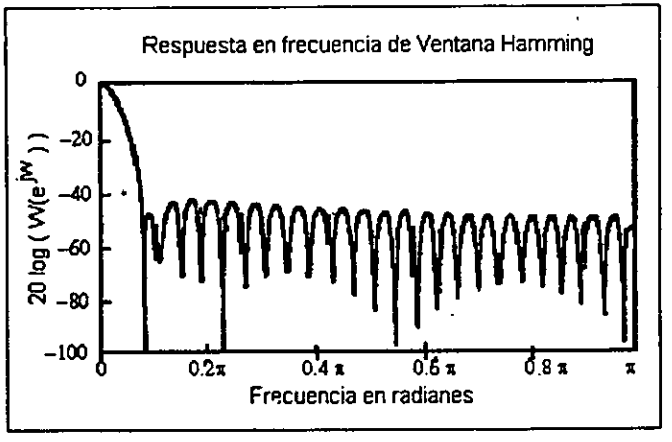

Fig. II-3 Respuesta en frecuencia de una ventana de Hamming

II.2.3.2 Energía en tiempo corto

La energía de una señal discreta en el tiempo se define por la siguiente expresión

Ec. II-9

$$E = \sum_{m=-\infty}^{\infty} x^2(m)$$

Si ahora ventaneamos la señal y obtendremos una expresión de la energía en una ventana.

Ec. II-10

$$E_n = \sum_{m=-\infty}^{\infty} [x(m)w(n-m)]^2$$

La ecuación anterior puede escribirse como una convolución de la señal con una ventana w(n), en este caso usamos Hamming, obteniendo la (Ec. 11), en donde h(n)=w(n)*w(n)

Ec. II-11

$$E_n = \sum_{m=-\infty}^{\infty} x(m)^2 h(n-m)$$

II.2.3.3 Magnitud Promedio

Una forma de determinar de manera más sencilla los cambios de energía es aproximar la energía en tiempo corto con a una función magnitud promedio definida por: [9]

Ec. II-12

$$M_n = \sum_{m=-\infty}^{\infty} |x(m)| w(m)$$

Las ventajas que tenemos al usar esta función con respecto a usar energía son:

- La energía aumenta cuadráticamente dando por resultado valores mucho más grandes, mientras que la magnitud promedio tiene su máximo en el máximo valor de la señal.
- Al no ser tan sensible a grandes valores, permite mejor análisis de pequeños valores.
- Simplifica los cálculos algebraicos

II.2.3.4 Tasa de cruces por cero en tiempo corto

La tasa de cruces por cero es una medida de la frecuencia contenida en una señal. Esta estimación es mejor para señales de banda angosta. Por ejemplo una señal senoidal de frecuencia Fo, muestreada a una tasa Fs, tiene Fs/Fo muestras por ciclo de señal. Cada ciclo tiene 2 cruces por cero así que a lo largo del tiempo la tasa promedio de cruces por cero es [4]

Ec. II-13

$$Z = \frac{2F_o}{F_S} \quad cruces/muestra$$

De tal manera que la tasa de cruces por cero, es una manera de estimar la frecuencia de la señal senoidal. La interpretación de la tasa de cruces por cero en voz, burdamente estima las propiedades espectrales. La definición para la función de cruces por cero es (Ec. 15).

Ec. II-14

$$Z_n = \frac{\sum_{m=-\infty}^{\infty} |\text{sgn}[x(m)] - \text{sgn}[x(m-1)]| w(n-m)}{2N}$$

donde ' sgn ', es la función signo, la expresión en valor absoluto toma valores de 2 cuando hay cruce por cero, 0 cuando no hay cruces, por ello dividimos entre (2 N), obteniendo un promedio de cruces por cero.

II.2.4 Transformada discreta de Fourier DFT.

Para el análisis en frecuencia de señales discretas en el tiempo {x(n)}, convertimos la señal en el dominio del tiempo a una representación equivalente en el dominio de la frecuencia. La transformada de Fourier de una señal discreta, es un espectro continuo. Este espectro continuo no es una forma de representación conveniente de la secuencia x(n). De tal forma que usaremos otra representación que se denomina DFT (Discrete Fourier Transform), que corresponde a un muestreo de la transformada continua de Fourier. [12]

Ec. II-15

$$X(w) = \sum_{n=-\infty}^{\infty} x(n) e^{-jwn}$$

En la (Ec. 16) X(w), representa un espectro continuo de la señal discreta x(n), si consideramos un muestreo del espectro en determinados puntos, $X(2\pi k / N) \quad k = 0,1,\ldots,N-1$ Tenemos:

Ec. II-16

$$X(k) = \sum_{n=0}^{N-1} x(n) e^{-j2\pi\, kn/N} \quad k = 0,1,2,\ldots,N-1$$

Que es la DFT, existen muchos estudios sobre ella, el método que se utilizó para obtener la DFT fue Radix-2, que es un algoritmo rápidos, generando así la denominada DFFT (Discrete Fast Fourier Transform) [5] [6].

II.2.4.1 Observaciones sobre la DFFT

Para este análisis de la DFFT, nos interesan varios aspectos observados, uno de ellos es que la DFT, es un muestreo del espectro, dicho muestreo involucra una resolución en frecuencia determinada de la siguiente manera.

Ec. II-17

$$\Delta f = \frac{Bw}{(N/2)}$$

La ecuación anterior indica que tomaremos ' N ' puntos del ancho de banda que tengamos, la división entre 2, se debe a que la DFFT es simétrica, el punto N/2 es la máxima frecuencia. Tal pareciera que aumentar el número de puntos ' N ', en la transformada sería una solución para tener mejor la resolución. Sin embargo en señales de voz, tenemos variaciones importantes en el tiempo que suceden en intervalos de tiempo cortos, como los fonemas /t/, /p/, o simplemente cambios de vocal a vocal, lo cual nos impide tener un número grande de muestras para analizar. De esta forma se vuelve una decisión en la cual a costa de mayor resolución perderíamos cambios o transiciones importantes de voz, mientras que poder analizar pequeños cambios implica tener una baja resolución en frecuencia.

II.2.5 Espectogramas

Los espectogramas que se utilizan en voz son representaciones del espectro de la señal, en intervalos de tiempo corto, generalmente en este caso usamos la DFFT de 256 puntos con ventanas de Hamming, con ello determinamos las frecuencias que existen en intervalos de tiempo. [18]

Ec. II-18

$$I = N\left[1/f_s\right] = 256\left[1/11025\right] = 23.21ms$$

De (Ec. 17) podemos obtener la resolución en frecuencia del espectograma.

Ec. II-19

$$\Delta f = \frac{5512.5}{(256/2)} = 43.06\ Hz$$

De las (Ec. 18,19) podemos darnos cuenta que la DFFT, es tan sólo una aproximación al espectro real de la señal, situación un tanto incomoda puesto que las ventanas analizan intervalos de tiempo que pueden ser largos para análisis, y la resolución en frecuencia es perceptible para ciertas frecuencias, sin embargo es un bosquejo del espectro.

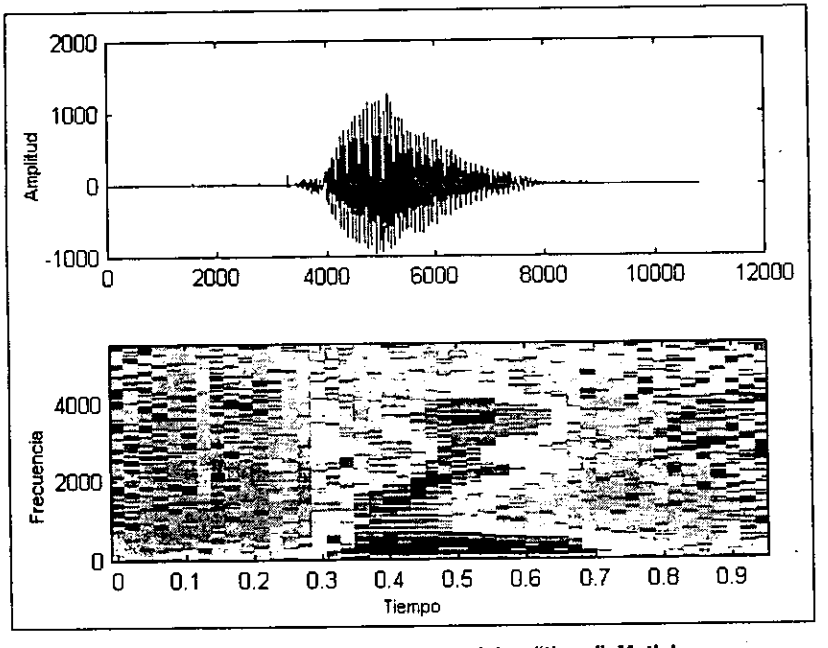

Fig. II-4 Señal y espectograma palabra "three", Matlab

Como manera de representación obtenemos una gráfica de 3D. En el ejemplo presentado en la (Fig. 4), la figura superior la señal en el tiempo y la inferior el espectograma. En las ordenadas tenemos la frecuencia, la máxima es la mitad de la frecuencia de muestreo (fs=11025 Hz). En las abscisas tenemos el eje del tiempo donde 0.96 corresponde al final de la palabra. Los colores representan la magnitud de la DFFT en Rojo los valores máximos de la DFFT y en Azul los mínimos.

Observamos ciertas características importantes de la voz:

1) En la zona de silencio, existen todas las frecuencias, características del ruido blanco, como se observa en tonos azules aleatorios.

2) En el sonido 't', existe un incremento súbito de corta duración de gran número de frecuencias.

3) Y en el sonido vocálico 'ee', podemos observar distintos formantes (frecuencias predominantes), para la generación de sonidos con fonación se utilizan varias frecuencias al mismo tiempo, correspondiendo a las cuerdas vocales del ser humano.

4) En la 'r', existen predominantemente frecuencias bajas, (vibraciones).

II.3 Características de la Voz Humana

II.3.1 Frecuencias fundamentales de la voz

La frecuencia fundamental de la voz humana varía según el sexo, se ha determinado su intervalo de valores para cada uno, como se muestra en la (Tabla 1).

Sexo	Frecuencia Fundamental
Masculino	50-250 Hz
Mujer	120-500 Hz

Tabla II-1 Frecuencias de voz por sexo

II.3.2 Frecuencias e Intensidad de conversación

En una conversación generalmente el rango de frecuencias empleado es de 100 Hz a 8000 Hz, mientras que la intensidad para una conversación natural a un metro de distancia tiene sus picos de 60 a 70 dB.

II.3.3 Características auditivas Humanas

II.3.3.1 Umbrales de audición humana

El rango de frecuencias que percibe un humano esta en el ancho de banda de (16Hz-18kHz). En el límite inferior de frecuencias el sonido se escucha como un tren de pulsos, mientras que en el superior se va perdiendo hacia el silencio. Es sabido que esta capacidad de audición se va perdiendo con la edad, reduciendo notablemente la percepción de la frecuencia superior de audición, en jóvenes llega hasta 20kHz la frecuencia superior. [19]

Otro aspecto importante es que se ha demostrado que el oído humano percibe 280 niveles de intensidad diferentes, y 1400 tonos distintos.

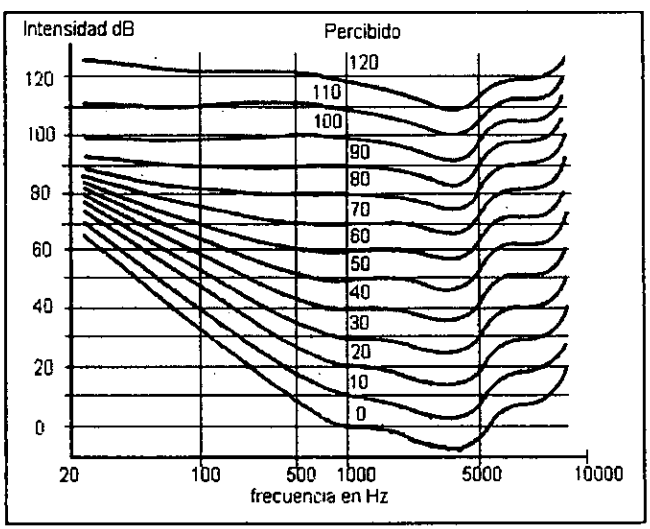

Fig. II-5 Respuesta del oído en intensidad y frecuencia

II.3.3.2 Percepción de Intensidad (fons)

La intensidad con la cual percibimos un sonido, depende de la frecuencia y la intensidad con la cual se genera el sonido. Según experimentos realizados por Fletcher-Munson (1933), se obtuvieron gráficas como la (Fig. 5). En ella podemos ver de manera implícita la respuesta que

presenta el oído. En el eje de ordenadas se traza la intensidad, en las abscisas se traza la frecuencia, las curvas que obtenemos son de percepción de los sonidos.

Por ejemplo un sonido de (60 dB a 100 Hz), se percibe igual que otro con (40 dB a 1000 Hz). Esto lo obtenemos de seguir la curva marcada con 40 dB en el eje central (1000 Hz). Para obtener las intensidades de los sonidos, las referimos a una constante de (10^{-16} W/cm^2) considerada como el mínimo umbral de audición humana.

De la (Fig. 5), también nace el concepto de 'fons', un 'fon' se define como la intensidad con la cual se percibiría un sonido si este tuviera una frecuencia de 1000 Hz. De manera que usando la figura si tuviéramos un sonido de (40 dB a 100 Hz) equivale a 10 fons.

Otra observación importante, es que de 2000-5200 Hz el oído es altamente sensible en ese intervalo de frecuencias necesitamos menos potencia para que el oído perciba el sonido. También podemos ver que alrededor de 1000 Hz el oído conserva una linealidad y su respuesta con la cual percibimos corresponde a la intensidad del sonido. [3]

II.3.3.3 Enmascaramiento

Es un fenómeno que ocurre cuando un sonido oculta la percepción de otro sonido diferente, se da en función de la intensidad y frecuencia de ambos sonidos. El Enmascaramiento en reconocimiento de voz puede influir de dos formas: nocivamente si una fuente ruidosa cancela nuestra señal de voz, o bien podemos usarlo a nuestro favor eliminando componentes de señal, simplificándola para usar solamente aquellas frecuencias que el oído humano escucha. Sin embargo la implementación de un sistema de este tipo es algo complicada, debido a que este proceso de Enmascaramiento, funciona conjuntamente con los fenómenos de percepción de armónicas y Pitch, además el fenómeno se comporta no linealmente y depende mucho de la frecuencia el nivel de atenuación de las otras señales. [3]

II.3.3.4 Bandas Críticas

Existe otro fenómeno observado en el oído humano, denominado bandas críticas. Se han realizado experimentos usando un generador de ruido blanco con ancho de banda variable, al percibir los sonidos provenientes de este generador se observó una variación en la percepción del sonido con respecto al ancho de banda del ruido. [3]

Numero de Banda	Frecuencia Central	Ancho de Banda Crítico	Frecuencia Inferior, fci.	Frecuencia Superior, fcs.
1	50	-	-	100
2	150	100	100	200
3	250	100	200	300
4	350	100	300	400
5	450	110	400	510
6	570	120	510	630
7	700	140	630	770
8	840	150	770	920
9	1000	160	920	1080
10	1170	190	1080	1270
11	1370	210	1270	1480
12	1600	240	1480	1720
13	1850	280	1720	2000
14	2150	320	2000	2320
15	2500	380	2320	2700
16	2900	450	2700	3150
17	3400	550	3150	3700
18	4000	700	3700	4400
19	4800	900	4400	5300

Tabla II-2 Bandas Criticas para la región de voz en canal telefónico

El experimento consistió en lo siguiente: Se generó ruido con un ancho de banda (fci,fcs) inicialmente pequeño, con frecuencia central Fo. Gradualmente se aumentaba el ancho de banda, en cambios inferiores al denominado ancho de banda crítico para dicha Fo, no se escucha cambio alguno, percibiéndose como un sólo tono de frecuencia Fo, el ruido. Pero súbitamente se escucha un cambio al pasar un límite de ancho de banda denominado Banda Critica, de tal forma que se obtuvo la (Tabla 2).

Si graficamos el ancho de banda contra frecuencia central obtenemos la (figura 2).

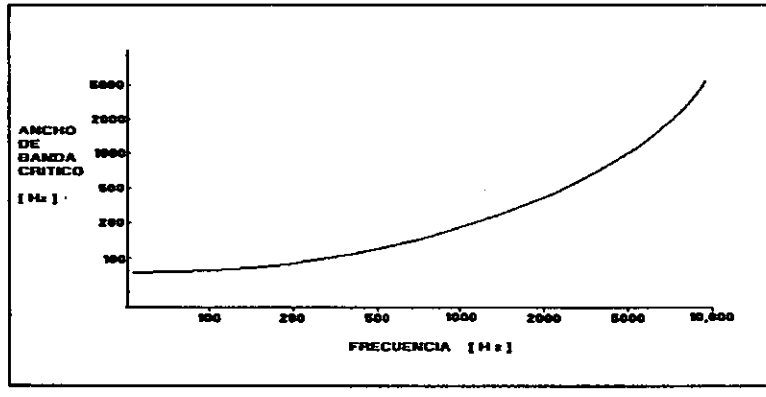

Fig. II-6 Ancho de banda crítico para frecuencias centrales

II.3.3.5 Detección de frecuencia fundamental virtual (Pitch)

Es conocido que aún cuando la frecuencia fundamental no esté presente en un sonido complejo, el oído humano es capaz de identificarla. Tal es el caso de la voz humana, la frecuencia fundamental para una voz masculina de 120 Hz, es claramente percibida a través de un canal telefónico, (300-3400 Hz, ancho de banda de voz considerado para un canal telefónico) aún cuando claramente está fuera del ancho de banda telefónico. [9]

II.3.3.6 Frecuencias armónicas

Las componentes armónicas no se escuchan como sonidos separados, sino que pareciera como si oyéramos un sólo tono. El alto contenido armónico le da al sonido, "timbre" o "calidad de tono", gracias a esto es que podemos identificar las vocales.

III. Ajuste Dinámico de Tiempo (DTW)

Las variaciones de la velocidad del locutor al hablar, causan fluctuaciones no lineales en la duración de las palabras. Un ajuste lineal, es insuficiente para compensar las diferencias de duración de las palabras. Es por ello que se ha utilizado una técnica que recurre a la programación dinámica, denominada DTW (Dynamic Time Warping o Ajuste Dinámico de Tiempo) Las diferencias de dos patrones de voz, son eliminadas ajustando el eje del tiempo de una de ellas a otra o ajustando ambos ejes para tener la máxima coincidencia entre ellas.

III.1 Principio de ajuste

Las señales de voz pueden ser representadas como una secuencia de vectores, [20]

$$A = a_1, a_2, \ldots, a_i, \ldots, a_I$$
$$B = b_1, b_2, \ldots, b_j, \ldots, b_J$$

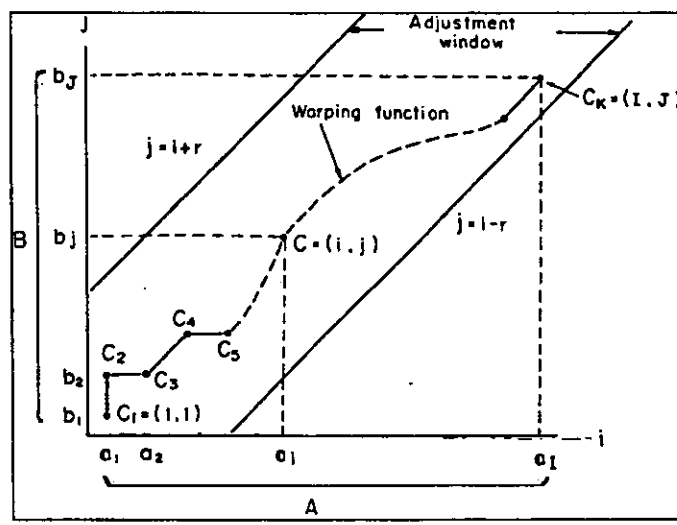

Fig. III-1 Región de ajuste, función de ajuste entre dos señales

Si queremos ajustar las fluctuaciones en el eje del tiempo entre estas dos secuencias, consideremos la (Fig. 1). En cada uno de los ejes ponemos las señales que queremos ajustar, de tal forma que generamos una matriz c(i,j), la idea ahora es encontrar un camino entre el punto inicial y final tomando en cuenta que tengamos una función de costo mínimo entre dichos puntos. A esta función de camino entre el punto inicial y final la llamamos función de ajuste c(k)=(i(k), j(k)).

Originalmente este problema se podría tratar con el Algoritmo de Dijkstra o Bellman-Ford de teoría de redes (1956) con la distancia de un punto a otro, sin embargo tenemos serios problemas de uso de memoria y tendríamos que generar una matriz de nodos demasiado grande de dimensión variable, lo cual ha ocasionado que se busquen restricciones a los puntos posibles para la función de costos y otros algoritmos para simplificación de trayectorias. Sin embargo la idea original de programación dinámica nace de estos algoritmos de intercambios de tablas en redes de computadoras.

III.2 Restricciones a la función de ajuste

1) Debe ser **monotónica**:

Ec. III-1

$$i(k-1) \leq i(k) \quad y \quad j(k-1) \leq j(k)$$

Debido a que estamos manejando un eje de tiempo, el no cumplir esta restricción implicaría que la señal se regrese en tiempo.

2) Condición de **continuidad**:.

Ec. III-2

$$i(k) - i(k-1) \leq 1 \quad y \quad j(k) - j(k-1) \leq 1$$

Debe existir la función de ajuste para todos los puntos pues de no ser así estaríamos dejando de considerar ventanas

3) Como resultado de las dos condiciones enunciadas, tenemos que la función de Warping tiene la siguiente relación entre puntos, denominado antecesores de c(k).

Ec. III-3

$$c(k-1)=\begin{cases}(i(k), j(k)-1)\\(i(k)-1, j(k)-1)\\(i(k)-1, j(k))\end{cases}$$

4) Condiciones de frontera: Se considera para el ajuste que comenzamos en un punto inicial (i(1), j(1))=(1,1), y llegamos a uno final (i(K), j(K))=(I, J).

En la (Fig. 2), podemos ver como se hace una función de ajuste dadas dos secuencias una de ellas con 6 muestras la otra con 5. Usando las restricciones anteriores podemos ver la forma que tiene la función de ajuste, que es dicho de otra forma la función de costos mínimos entre un punto y otro.

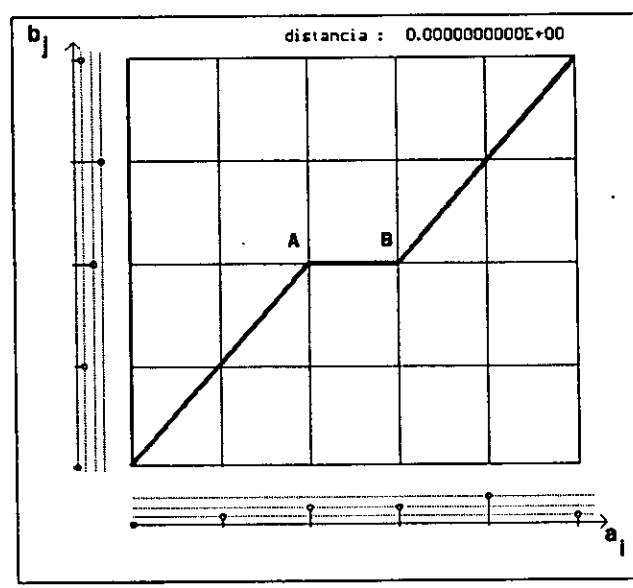

Fig. III-2 Ejemplo de función de ajuste entre dos secuencias

Se puede observar que en el segmento AB, la función es una línea horizontal, ello indica que al comparar ambas señales, la información contenida en el punto a(3) es redundante o bien si la suprimimos ambas secuencias tendrán mayor parecido, la función de costos mínimos indica en que puntos es viable una supresión de muestras si quisiéramos que ambas señales se parecieran más. Sin embargo en nuestro caso no nos interesaremos por hacerlas semejantes sino por medir las diferencias.

5) Región de ajuste: Es una restricción que limita en un entorno la función de ajuste, esto debemos hacerlo para evitar comparaciones que permitan demasiadas variaciones en los ejes de tiempo, y a su vez no limiten en exceso el ajuste de diferencias . El uso de regiones hace que tengamos menor número de cálculos. [9]

Estas regiones de ajuste han sido estudiadas [22] y sus parámetros se han obtenido experimentalmente. La que mejores resultados ha arrojado, es la propuesta por Sakoe y Chiba, consiste en una ventana como la que se muestra en la (figura 1). Podemos mejorar los resultados al aplicar una modificación (Alhmed) [4], que consiste en trazar una recta que una el punto inicial con final, después trazar dos rectas paralelas a dicha recta separadas por una distancia 'r'. De tal forma que

Ec. III-4

$$|i(k) - j(k)| \leq \frac{I}{J} + r$$

6) Restricción de comparación debida a la longitud de señales. La longitud de las señales, (I/J) debe estar entre (0.5, 2). En los límites de esta condición la función de ajuste se vuelve una línea recta, más allá del intervalo saltaríamos puntos, condición posible pero no deseable debido a un ajuste sin considerar partes de señal. Se ha encontrado que el intervalo para permitir un buen ajuste es (0.6, 1.33) debido a que grandes pendientes, causan comparaciones irreales entre patrones cortos y largos. [22]

7) Restricción sobre la pendiente de la función de ajuste es una restricción en el número de veces que la función puede ser horizontal o vertical antes de ser diagonal, para evitar un mal Warping con ajustes muy largos algunos segmentos (Fig. 3)

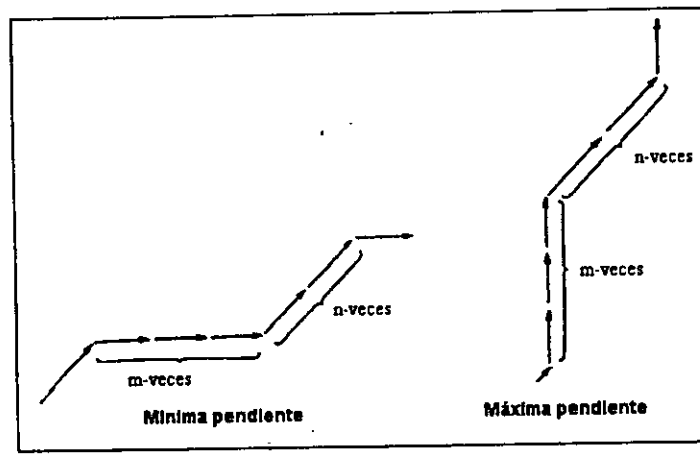

Fig. III-3 Pendientes máxima y mínima en la función de ajuste

Sakoe y Chiba [20] hicieron muchos estudios sobre esta pendiente, determinando así que la condición óptima es cuando P=(n/m)=1. Con lo cual, tenemos una simplificación de caminos.

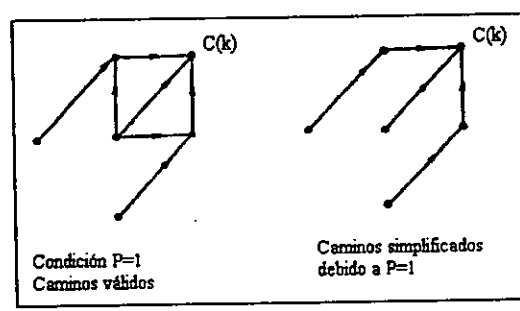

Fig. III-4 Antecesores válidos si P=1.

La solución al problema se ha convertido en considerar las trayectorias de la (Fig. 4) como una máscara que barre la región permitida para la función de ajuste dinámico.

III.3 Algoritmo DTW

A continuación se presenta el diagrama de flujo de programación dinámica para realizar un ajuste dinámico en el tiempo, Sakoe-Chiba, Existen diversos estudios [23], [20] que demuestran que el siguiente algoritmo y ecuación dinámica es la mejor en cuanto a resultados.

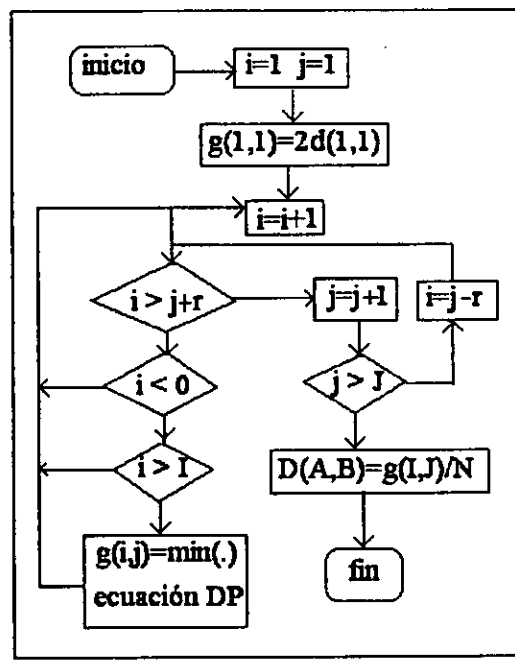

Fig. III-5 Diagrama de flujo DTW

la ecuación DP es la siguiente para P=1.

Ec. III-5

$$g(i,j) = min\begin{bmatrix} g(i-1,j-2)+2d(i,j-1)+d(i,j) \\ g(i-1,j-1)+2d(i,j) \\ g(i-2,j-1)+2d(i-1,j)+d(i,j) \end{bmatrix}$$

Cabe señalar que estos posibles caminos, realizan un ajuste en ambos ejes de tiempo, por ello se considera que esta ecuación DP, es simétrica.

III.4 Ejemplo de un ajuste real

En la (Fig. 6), podemos observar un ejemplo con señales reales, que corresponden a dos palabras /eight/, del mismo locutor.

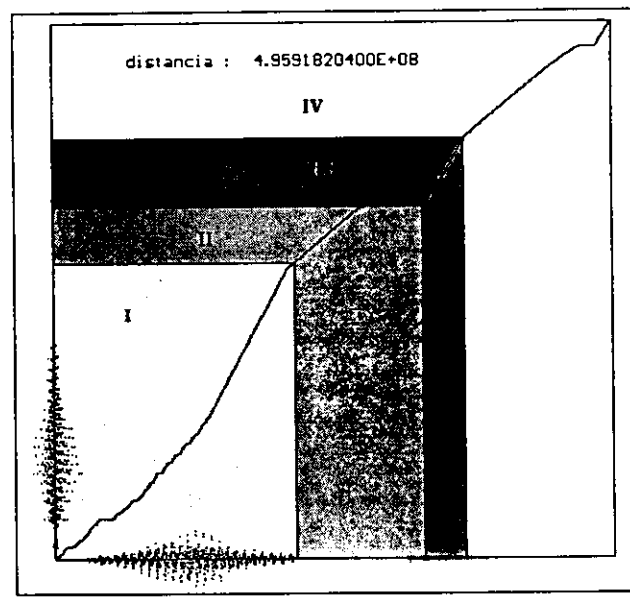

Fig. III-6 Ejemplo de DTW con señales de voz

La función Warping, se calculó usando las señales en el tiempo y distancia euclidiana [20], entre puntos. Para el análisis del significado de la función de ajuste trataremos cuatro regiones.

I	La función realiza pequeños ajustes en ambas señales
II	La función tiende a ser horizontal debido a la presencia de silencio en la abscisa mientras que en las ordenadas ya llegó voz, una función horizontal indica que todos los puntos de abscisa podemos condensarlos en uno solo de la ordenada .
III	Al comenzar voz de nuevo, nuevamente la función se vuelve diagonal
IV	Nuevamente al tener silencio largo la función se vuelve más horizontal indicando una posible compresión de la señal en las abscisas

Tabla III-1 Descripción de las regiones del DTW

Como conclusión importante de la (Tabla 1), la función de ajuste (Warping), indica que pedazos de señal podemos condensar o reducir para que se asemeje una a otra. Sin embargo en el caso del sistema de reconocimiento, no nos interesa saber la función de costos, ni si una región o otra puede parecerse, sino una medida general de similitud entre una y otra señal. Por ello en nuestro no vamos acumulando la función de Warping, lo cual llevaría gran cantidad de datos, sino que únicamente vamos tomando el camino mínimo y sumando distancias por puntos. Como medida total ocupamos la suma total de costos Ck(I , J).

IV. Detección de inicio - fin de palabras aisladas

El problema de detección de inicio y fin de una palabra en presencia de ruido es complicado debido a que el ambiente en el cual trabajamos no está libre de ruido. Para grabaciones en cuartos a prueba de ruido, podemos recurrir únicamente al uso de la energía en tiempo corto para la detección, sin embargo en ambientes con ruido es necesario tomar otras consideraciones.

En el sistema de reconocimiento, el poder determinar que cual es el inicio y fin de una palabra, nos da ciertas ventajas:

1) Procesar menor número de información, por ende menor tiempo de reconocimiento
2) Comparar como es el caso de DTW únicamente los patrones de información,
3) Evitar confusiones por ruido o señales de fondo
4) Para DTW es sumamente importante, debido a la condiciones iniciales y las restricciones para realizar ajustes.

Sin embargo encontramos diversos problemas en la detección.

1) Existe espurias de ruido que se pueden confundir con señal.
2) Las palabra pueden contener silencios dentro de si (ej. /s/, /i/,/ /, /ks/ - six), que pueden confundirse con un falso principio o fin.
3) Los fonemas fricativos (/f/ , /th/, /h/) tienen baja energía.
4) Existen sonidos muy cortos (ej. /t/, /p/, /k/)
5) Detección de fonemas nasales al final de la palabra.
6) Hay respiraciones del locutor, que pueden confundirse por su duración.
7) Los micrófonos tienen cierta resonancia, después de grabar sobre todo vocales.
8) Los niveles de ruido, pueden confundirse con señal de voz

Ante las dificultades para la detección de inicio y fin, se ha desarrollado un método que consiste en considerar las características de los sonidos, (Tabla 1)

Sonidos Vocálicos	Tienen alto contenido de energía Ocupan las frecuencias inferiores del espectro de voz humana.
Sonidos No-Vocálicos	Tienen bajo contenido de energía Ocupan las frecuencias superiores del espectro de voz humana.

Tabla IV-1 Clasificación de los sonidos

De tal forma que podemos implementar un detector que incluya ambas características, análisis de energía y frecuencia. Debido a que sólo requerimos una estimación de la frecuencia y energía podemos hacer uso de las estimaciones, cruces por cero y magnitud promedio.

IV.1 Algoritmo para detección inicio - fin

Rabiner y Sambur propusieron en [9] un método para su localización.

IV.1.1 Detección de inicio

1) Calcular las funciones; cruces por cero { Z(n) } y magnitud promedio de la señal { M(n) }.

2) Considerar que las primeras ventanas (cinco) son ruido, con lo cual tenemos :

$$Ms(n)=\{E(1), E(2), \ldots, E(5)\}$$
$$Zs(n)=\{Z(1), Z(2), \ldots, Z(5)\}$$

3) Calcular la media y la desviación estándar para caracterizar al ruido y obtener los siguientes umbrales

Umbral	Nombre del umbral	Valor
ITU	Umbral superior de energía	0.5 max { M(n) }
ITL	Umbral inferior de energía	$m_{Ms} + 2\sigma_{Ms}$
IZCT	Umbral de cruces por cero	$m_{Zs} + 2\sigma_{Zs}$

Tabla IV-2 Umbrales usados para la detección inicio - fin

4) Recorremos la función **M(n)** incrementando **'n'**, desde **(n=6)** hasta que **M(n)>ITU**, este punto es el inicio de palabra detectado por medio de energía (**Ie**), al buscar el punto ITU

estamos garantizando presencia de señal, al recorrer hacia el punto de inicio buscamos el punto de incremento importante de energía apartir de estar seguros que existe señal.

5) Decrementamos **'n'**, desde **(n=Ie)** hasta que sucedan alguna de las siguientes condiciones en la función de cruces por cero, pues lo que ahora buscamos es la posibilidad de que a un sonido vocálico de preceda un sonido no vocálico.

- Si **{ [(n < Ie-25) ó (n<6)] y [Zs(n) < IZCT] }** significa que no encontramos alguna porción de señal con aumento importante de frecuencia en 25 ventanas anteriores por lo tanto el punto de inicio es **Ie.**

- Si encontramos que **[Zs(n) > IZCT]** menos de tres veces seguidas significa que sólo fue una espiga de ruido, el punto inicio sigue siendo Ie.

- Si encontramos que **[Zs(n) > IZCT]** más de tres veces seguidas hemos encontrado un sonido no - vocálico, entonces buscamos el punto **'n'** para el cual **[Zs(n) < IZCT]** es decir hasta que después de haberse mantenido arriba del umbral, baje de él indicando el comienzo del sonido no - vocálico, por ello movemos el inicio de la palabra a **Iz.**

IV.1.2 Detección de fin

Para la detección de fin de palabra, según Rabiner hacemos lo mismo pero en sentido inverso a partir del punto 4) de la sección anterior como si detectáramos un inicio con la señal invertida en el tiempo.

IV.1.3 Modificaciones al método de Rabiner-Sambur

En los experimentos que realicé observé que existe cierta resonancia del micrófono al terminar de pronunciar una palabra de forma que las estadísticas del ruido no son iguales al principio de la palabra que el final, por ello haciendo una modificación al método de Rabiner, se deben volver a calcular las estadísticas usadas para el fin de palabra.

El método anterior tiene varias desventajas, los parámetros de número de ventanas de análisis de ruido y número de ventanas permitidas para buscar sonidos no-vocálicos, fueron determinados por Rabiner para su aplicación particular, con sus ventanas de análisis. En la experimentación observé que para varias palabras el método falla debido a estas consideraciones sobre todo en palabras con respiraciones, alto ruido, y fonemas muy cortos.

Una modificación que hice al método es ocupar en lugar de la Magnitud promedio, la función Log (Mn), la cual da mayor suavidad y peso a las áreas de baja aportación de energía, con lo cual casi prescindimos del uso de cruces por cero.

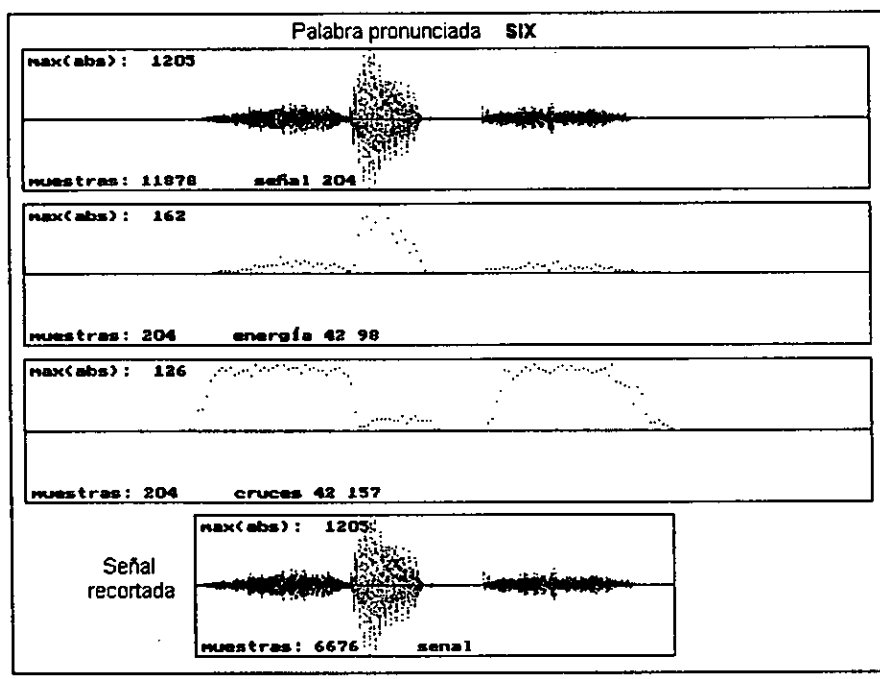

Fig. IV-1 Programa para la detección de inicio - fin

V. Sistema de Reconocimiento LPC

V.1 Descripción del sistema de reconocimiento LPC

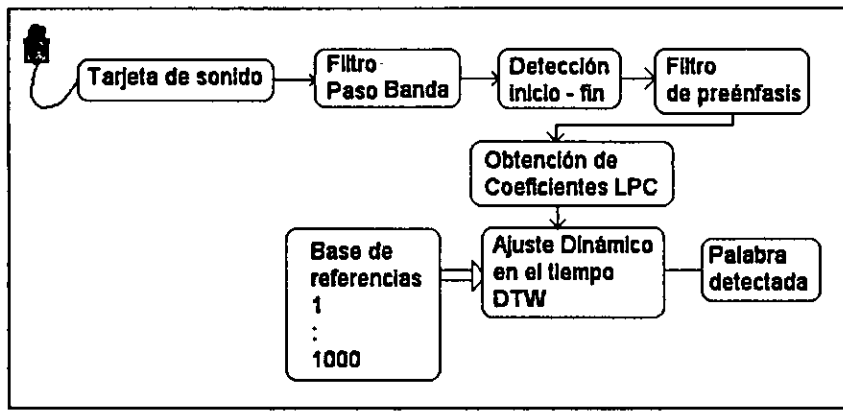

Fig. V-1 Diagrama de Bloques del sistema LPC

La (Fig. 1), representa el diagrama de bloques del sistema de reconocimiento desarrollado. La palabra a reconocer es digitalizada por la tarjeta de sonido, se filtra (300-4000 Hz), evitando así cálculos innecesarios y mejorar el reconocimiento, se recorta la señal para tener un inicio y fin de palabra, se obtienen los coeficientes LPC del predictor, estos valores junto con los coeficientes de correlación se comparan por medio de DTW con mil referencias de las cuales se encuentran almacenados sus LPC y coeficientes de autocorrelación, finalmente sale del sistema la palabra de la base con la cual se reconoció la entrada, que corresponde a la palabra con mínima distancia.

V.2 Filtro de Preénfasis

El modelo LPC tiene buen desempeño el bajas frecuencias y un bajo desempeño en altas frecuencias debido a que es un predictor lineal. Para tratar este problema, aplicamos a la señal un filtro paso altas adaptivo de primer orden, llamado filtro de Preénfasis. Su frecuencia de corte dB esta entre (100 - 1000 Hz) sin tener relevancia su ubicación. Con ello intentamos

hacer el espectro más plano. De manera que el filtro de Preénfasis logra que el espectro medido tenga un rango dinámico similar en toda la banda de frecuencias. [11]

La función de transferencia de este filtro es:

Ec. V-1

$$H(z) = 1 - a\,z^{-1}$$

expresado como una ecuación en diferencias tenemos

Ec. V-2

$$y(n) = x(n) - a\,y(n-1)$$

Por lo general se escoge a=[0.9-1.0], tomándose usualmente a=0.95. Es factible dar al filtro mayor adaptabilidad usando a=R(0)/R(1), un cociente de coeficientes de correlación.

V.3 Coeficientes de predicción lineal, LPC (Linear Predictive Coding)

V.3.1 Modelo LPC

La idea principal de la codificación, con los coeficientes de predicción lineal, consisten en tener un modelo matemático que genere voz como lo hacen los humanos.

El concepto de predicción lineal nace de la idea de aproximar una muestra de voz, usando una combinación lineal de las 'p' muestras anteriores, donde 'p' es el orden del sistema.

Ec. V-3

$$s(n) \approx a_1 s(n-1) + a_2 s(n-2) + \ldots + a_p s(n-p)$$

La ecuación anterior se interpreta como un filtro lineal de todos polos, los coeficientes 'a', se van calculando para cada ventana de análisis de la señal., los coeficientes del predictor se determinan para que el error de predicción sea mínimo. [11]

Si quisiéramos implementar este modelo de predicción a síntesis de voz tendríamos que considerar dos entradas al sistema, haciendo una analogía con los seres humanos. Para la

producción de sonidos vocálicos se usa como entrada un tren de pulsos periódicos y por otra parte para la producción de no - vocálicos usamos una fuente de ruido. A este modelo lo llamamos Modelo Fuente Filtro, el cual se muestra en (Fig. 2)

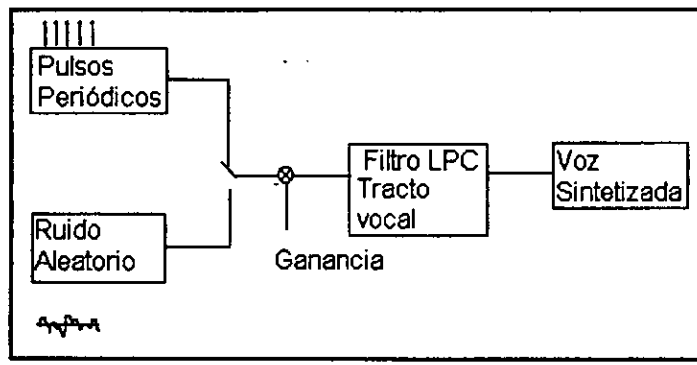

Fig. V-2 Modelo Fuente Filtro

La función de transferencia del sistema es la siguiente,

Ec. V-4

$$H(z)=\frac{S(z)}{GU(z)}=\frac{1}{1-\sum_{i=1}^{p}a_i z^{-i}}=\frac{1}{A(z)}$$

Donde:

G	Ganancia del filtro / entrada
S	Señal deseada
U	Entrada
a	Coeficientes del predictor
p	Orden del predictor

V.3.2 Ecuaciones de análisis LPC

Ahora bien hasta el momento no se hemos calculado los coeficientes 'a' . Para ello, supongamos que la entrada al sistema es un tren de pulsos u(n), y antitransformando la función de transferencia, podemos expresar la señal queremos representar en términos del sistema.

Ec. V-5

$$s(n) = \sum_{k=1}^{p} a_k s(n-k) + G\,u(n)$$

Si consideramos una estimación se la señal, en base a las 'p', muestras anteriores partiendo del principio de predicción lineal, (combinación lineal) tendremos que:

Ec. V-6

$$\tilde{s}(n) = \sum_{k=1}^{p} a_k s(n-k)$$

El error de predicción e(n) lo definimos como una diferencia de la señal deseada y la estimada.

Ec. V-7

$$e(n) = s(n) - \tilde{s}(n) = s(n) - \sum_{k=1}^{p} a_k s(n-k)$$

así pues la función de transferencia del error, se describe por la (Ec. 8)

Ec. V-8

$$\mathrm{A}(z) = \frac{E(z)}{S(z)} = 1 - \sum_{k=1}^{p} a_k z^{-k} = GU(z)$$

V.3.3 Obtención de los coeficientes LPC

Queremos obtener los coeficientes 'a', de la señal para que las propiedades espectrales del filtro, se ajusten a la ventana de análisis de la señal. Como las propiedades de la señal de voz cambian en el tiempo, los coeficientes del filtro se tienen que calcular ventana por ventana, es decir cada ventana queda representada por un conjunto de coeficientes LPC.

Para obtener los coeficientes LPC, debemos buscar que minimicen el error cuadrático medio de predicción en la ventana que queremos representar. Para ello definiremos la señal de voz en tiempo corto y los segmentos de error en el tiempo n.

Ec. V-9

$$s_n(m) = s(n+m)$$

Ec. V-10

$$e_n(m) = e(n+m)$$

el error cuadrático medio de la señal en el tiempo 'n' se describe por (Ec. 11)

Ec. V-11

$$E_n = \sum_m e_n^2(m)$$

de acuerdo a las ecuaciones (Ec. 7, 9,10) podemos escribir la ecuación anterior como.

Ec. V-12

$$E_n = \sum_m \left[s_n(m) - \sum_{k=1}^{p} a_k s_n(m-k) \right]^2$$

Minimizando la función de error anterior, obtenemos (Ec. 14)

Ec. V-13

$$\frac{\partial e}{\partial a_i} = 0 \quad Para \quad i = 1,2,\ldots, p$$

Ec. V-14

$$\sum_m s_n(m-i)\, s_n(m) = \sum_{k=1}^{p} a_k \sum_m s_n(m-i)\, s_n(m-k)$$

En esta expresión (Ec. 14) la segunda sumatoria del lado derecho es la covarianza en tiempo corto de $s_n(m)$. Definida por la (Ec. 15).

Ec. V-15

$$\phi_n(i,k) = \sum_m s_n(m-i) s_n(m-k)$$

Podemos expresar usando la definición anterior nuestro resultado (Ec. 14) :

Ec. V-16

$$\boxed{\phi_n(i,0) = \sum_{k=1}^{p} a_k \, \phi_n(i,k)}$$

La cual describe un conjunto de 'p' ecuaciones, con 'p' incógnitas.

V.3.3.1 Método de autocorrelación (ventaneo)

Como podemos observar el error de predicción, depende mucho 'm', debido al aumento del error de predicción al principio de la ventana por suponer cero, y al final de la ventana por calcular estimaciones, recurrimos al uso de ventanas como Hamming que minimicen el peso de la señal en los límites de la ventana. La señal ventaneada esta dada por

Ec. V-17

$$s_n(m) = \begin{cases} s(m+n)w(m) & 0 \le m \le N-1 \\ 0 & otro \end{cases}$$

Tomando en cuenta la señal ventaneada el error cuadrático medio usando (Ec. 11)

Ec. V-18

$$E_n = \sum_{m=0}^{N-1+p} e_n^2(m)$$

La (Ec. 15) al ventanear se convierte en

Ec. V-19

$$\phi_n(i,k) = \sum_{m=0}^{N-1+p} s_n(m-i)s_n(m-k) \quad \begin{cases} 1 \le i \le p \\ 0 \le k \le p \end{cases}$$

o bien

Ec. V-20

$$\phi_n(i,k) = \sum_{m=0}^{N-1-(i-k)} s_n(m)s_n(m+i-k) \quad \begin{cases} 1 \le i \le p \\ 0 \le k \le p \end{cases}$$

Observamos que sólo depende de (i - k), entonces la covarianza se reduce a una función de autocorrelación

Ec. V-21

$$\phi_n(i,k) = r_n(i-k) = \sum_{m=0}^{N-1-(i-k)} s_n(m)s_n(m+i-k) \quad \begin{cases} 1 \le i \le p \\ 0 \le k \le p \end{cases}$$

Debido a la simetría de la correlación podemos expresarla por

Ec. V-22

$$\sum_{k=1}^{p} r_n(|i-k|)a_k = r_n(i) \quad 1 \le i \le p$$

Que matricíalmente genera un sistema de ecuaciones a resolver.

Ec. V-23

$$\begin{bmatrix} r_n(0) & r_n(1) & r_n(2) & \cdots & r_n(p-1) \\ r_n(1) & r_n(0) & r_n(1) & \cdots & r_n(p-2) \\ r_n(2) & r_n(1) & r_n(0) & \cdots & r_n(p-3) \\ \vdots & \vdots & \vdots & & \vdots \\ r_n(p-1) & r_n(p-2) & r_n(p-3) & \cdots & r_n(0) \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ \vdots \\ a_p \end{bmatrix} = \begin{bmatrix} r_n(1) \\ r_n(2) \\ r_n(3) \\ \vdots \\ r_n(p) \end{bmatrix}$$

V.3.3.2 Algoritmo Levinson - Durbin para obtención de coeficientes LPC

Usando el método de análisis LPC de correlación, vemos que la matriz (Ec. 23), es una matriz simétrica, observando los términos por columnas, nos damos cuenta que tiene una característica especial, los índices de la correlación son progresivos hacia arriba y abajo de la diagonal principal, además la diagonal principal es r(0) . A esta matriz se le llama Toeplitz, una de sus propiedades importantes es que el sistema (Ec. 23) siempre tiene solución. [4]

Debido a las características enunciadas existe un método más efectivo para el cálculo de los coeficientes LPC, que otros métodos para la resolución de sistemas de ecuaciones. Este método desarrollado por Levinson - Durbin, consiste en lo siguiente: [11]

{ entrada al algoritmo : coeficientes de autocorrelación r(n)}

1) $E(0) = r(0)$

2) $K_i = -\dfrac{r(i) + a_1^{(i-1)} r(i-1) + \ldots + a_{i-1}^{(i-1)} r(1)}{E(i-1)} \quad i = 1,2,\ldots,p$

3) $a_i^i = k_i$

4) $a_j = a_j^{(i-1)} + k_i a_{i-j}^{(i-1)} \quad j = 1,\ldots,i-1$

5) $E(i) = (1 - k_i^2) E(i-1)$

Se calculan iterativamente los pasos del 2) al 5), hasta que se tengan todos los valores de a_k, del siguiente conjunto $a_k = \{a_1^p, a_2^p, \ldots, a_p^p\}$ *para* $j = 1,\ldots,p$ que corresponden a los coeficientes LPC.

Una observación por hacer es que el algoritmo anterior no funciona para el caso que E(0)=0, es decir que la señal sea completamente 0, por ello es importante la detección de inicio y fin de palabra.

V.4 Distancia de Itakura - Saito

Los coeficientes LPC como hemos visto pertenecen a un filtro generador de voz, y se aplica en sistemas de síntesis, es muy común que encontremos en el mercado sintetizadores que usen coeficientes LPC orden 8-12.

Pero en nuestra aplicación de reconocimiento, haremos a un lado las fuentes y nos concretaremos usar los LPC, representando a la señal de voz, conservando información sobre ella. Lo anterior nos lleva a pensar que en lugar de utilizar toda la señal de voz simplemente podemos usar los LPC, que correspondan a cada una de las ventanas de análisis que tomemos

de la señal completa de voz y compararlos de alguna manera para decidir si una palabra es igual a otra.

Una de las distancias más utilizada y que mejores resultados ha producido, es la llamada distancia de Itakura - Saito, [21]

Ec. V-24

$$D(a_R,a_T) = \log\left[\frac{a_R V_T a_R}{a_T V_T a_T}\right] \neq D(a_T,a_R)$$

Variables	Dimensión	Significado
a_R	1 x (p+1)	Vector de coeficientes LPC de referencia
a_T	1 x (p+1)	Vector de coeficientes LPC de prueba
V_T	(p+1) x (p+1)	Matriz de autocorrelación Toeplitz de prueba

Tabla V-3

Como podemos analizar es una distancia no simétrica, no es lo mismo una señal como referencia que como prueba, se han hecho intentos por promediar ambas distancias pero no hay mejoras notables en el reconocimiento y sí aumentos en el cálculo.

Esta distancia surge al buscar el error cuadrático medio de predicción que tiene un sistema al producir una señal, y el error que se tiene al ser producida por otro sistema. La relación logarítmica simplemente es para resaltar las diferencias. (Maximum Likelihood Ratio)

V.5 Compresión de datos del sistema

La idea que lleva reducir el número de datos de la señal de voz es poder representar a la señal de voz con sus características esenciales para el reconocimiento y a su vez trabajar con menor número de recursos, buscando mayor velocidad de reconocimiento y precisión.

Bloque	Longitud de archivo (bytes)	Compresión	Observaciones
Señal digital	22,000	A/D	**Muestreo:** 11025 Hz **Cuantización:** 65536 niveles **Codificación:** 16 bits/muestra
Detección inicio - fin	13,200	6:10	Reducción en un 60% al quitar silencios al inicio y fin de la palabra grabada.
Filtrado	26,400	2:1	Aumenta debido al cambio de manejo numérico de enteros a flotantes (entero 16 bits, flotante 32 bits)
LPC orden (p=8)	4,320	1:6	**Longitud de ventana:** 128 **Traslape:** 18 muestras **Número de ventanas:** 6600/(128-18) = 60. **Longitud de la cadena por ventana:** **LPC** =(9)*32 = 288 bits **Coef. Correlación** = (9)*32 = 288 bits
Compresión Total		1:5	

Tabla V-4 Compresión (aproximada) de información del sistema

En el sistema descrito tenemos una compresión de datos como la mostrada en la (Tabla 4), que se obtuvo usando la longitud promedio de una grabación de voz por palabra (aprox. 1 s), calculada de 1000 repeticiones de los dígitos en inglés.

VI. Sistema de Reconocimiento KLT

VI.1 Descripción del sistema KLT

El sistema desarrollado se describe en forma de diagrama de bloques de la siguiente manera:

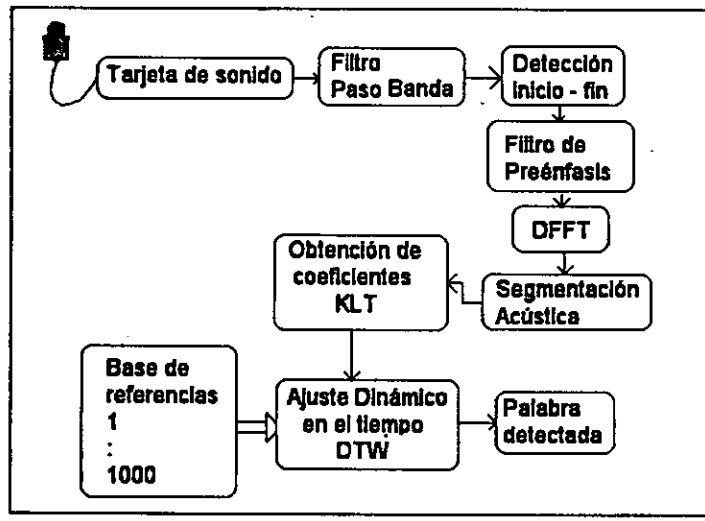

Fig. VI-1 Diagrama de Bloques del sistema KLT

La palabra a reconocer es capturada por la tarjeta de sonido, se filtra (300-4000 Hz), se recorta para tener un inicio y fin de palabra, se obtiene la DFFT, el siguiente paso es tener los puntos de segmentación acústica generando intervalos donde prevalezcan las mismas características en sentido estadístico, se obtiene una matriz de eigen vectores y valores propios [7] para cada segmento acústico Se comparan por medio de DTW con las referencias, dando como salida la señal de referencia con la cual fue reconocida la palabra a reconocer.

VI.2 Transformada KLT

KLT es una transformación óptima en sentido estadístico, decorrelaciona la secuencia en el dominio de la transformada. [14], lo cual la hace ser mejor de la FFT y DCT. Es un proceso

independiente un coeficiente del otro, las funciones son eigen vectores de la matriz de covarianza y los elementos diagonales son las varianzas de la transformada. [15]

Sea una vector de variables aleatorias

Ec. VI-1

$$X = \{x(0), x(1), \ldots, x(N-1)\}$$

La expansión en serie KLT, se obtiene de la siguiente manera (i = 0...N-1)

Ec. VI-2

$$\aleph_i = < x_i, \phi_i > / < \phi_i, \phi_i >$$

La transformada inversa KLT para recuperar la señal es.

Ec. VI-3

$$X = \sum_{i=0}^{N-1} \aleph_i \phi_i$$

Si sólo tomamos en cuenta algunos coeficientes. (D < N)

Ec. VI-4

$$\tilde{X} = \sum_{i=0}^{D-1} \aleph_i \phi_i$$

Ahora queremos determinar ϕ para que el error cuadrático medio sea mínimo, entre la predicción y la señal original, pero tomando en cuenta sólo una parte de los coeficientes de la expansión KLT. Tenemos que el error cuadrático medio es:

Ec. VI-5

$$\varepsilon = E\left[(x - \tilde{x})^2\right]$$

Ec. VI-6

$$\varepsilon = E\left[\left\langle \sum_{i=D}^{N-1} \chi_i \phi_i, \sum_{i=D}^{N-1} \chi_i \phi_i \right\rangle\right]$$

Queremos que ϕ sea ortogonal nuestra transformación puesto que debe decorrelacionar la señal, además la transformación debe conservar la energía, razón por la cual también debe ser ortonormal

Ec. VI-7

$$< \phi_i, \phi_k >= \partial_{ik}$$

Considerando lo anterior (Ec. 6) se transforma en:

Ec. VI-8

$$\varepsilon = E\left[\sum_{i=D}^{N-1} |\chi_i|^2\right]$$

Ec. VI-9

$$\varepsilon = E\left[\sum_{i=D}^{N-1} |< x_i, \phi_i >|^2\right]$$

Ec. VI-10

$$\varepsilon = E\left[\sum_{i=D}^{N-1} |< \phi_i{}^T x x^T \phi_i >|^2\right]$$

Ec. VI-11

$$\varepsilon = \sum_{i=D}^{N-1} \phi_i{}^T E[x x^T] \phi_i$$

Donde podemos observar que el valor esperado corresponde a la matriz de covarianza, y que el error cuadrático medio de estimación depende del número de coeficientes (N-1-D) no considerados por el truncamiento de coeficientes KLT. Llamaremos A, a dicha matriz de covarianzas.

Ec. VI-12

$$[A] = E[x x^T]$$

Usando multiplicadores de Lagrange, para minimizar el error, Hacemos que:

Ec. VI-13

$$\left(\frac{\partial}{\partial \phi_i}\right)\{\varepsilon - \mu_i < \phi_i, \phi_i >\} = 0$$

Ec. VI-14

$$([A] - \mu_i [I_N])\phi_i = 0$$

Si $\phi_i = [\phi_0, \phi_1, \ldots \phi_{N-1}]$ Podemos expresar la ecuación anterior como:

Ec. VI-15

$$[\phi]^T [A][\phi] = diag[\mu_0, \mu_1, \ldots \mu_{N-1}]$$

Entonces una expresión importante es que el error cuadrático medio, es la suma de los valores característicos que no considerados, por ello es importante tomar los más grandes. Además el espacio que debemos generar debe ser un espacio ortonormal, Finalmente expresamos el error cuadrático medio por (Ec. 16) donde las μ, representan los valores característicos. [17]

Ec. VI-16

$$\varepsilon = \sum_{i=D}^{N-1} \mu_i$$

VI.2.1 Propiedades de la transformada KLT

$l.i.m.\,\tilde{x}(n) = x(n)$

$\sum_{i=0}^{N-1} \phi_p(n)\phi_m^T(n) = 1$ sólo si m=p, 0 en otro caso

$E\{A_n A_m^T\} =$ ln si M=n, 0 en otro caso

$E\{x^2(n)\} = Rxx(0) = \sum_{n=1}^{\infty} \lambda_n$

MSE normalizado es igual a $\dfrac{\sum_{n=D+1}^{\infty} \lambda_n}{\sum_{n=1}^{\infty} \lambda_n}$

VI.2.2 Ventajas y Desventajas de la transformación KL

Ventajas

1) Decorrelaciona (Para la distribución Gausiana, los coeficientes KLT son independientes)
2) Empaqueta la mayor energía (varianza) en el menor número de coeficientes de transformación
3) Minimiza en mse.

Desventajas

1) No hay un algoritmo rápido
2) Es muy difícil de programar y no se saben de antemano los eigen valores y eigen vectores.
3) La matriz de covarianza se va recalculando.
4) KLT no es una transformada fija sino que tiene que generarse para cada señal.
5) Si la usamos en el dominio del tiempo introduciendo señal en ella tendremos matrices de dimensión aleatoria.

VI.3 Implementación en el sistema de la transformada KLT

En varios textos [14] [15] se encuentra que la transformación KLT es una transformada de referencia puesto que su dificultad de implementación y la inexistencia de algoritmos rápidos la hacen impráctica. El hecho de generar matrices de campos aleatorios, hace que su implementación no sea redituable en transmisión de datos, o compresión. Sin embargo en el sistema de reconocimiento que se implementa se hace una modificación a dicha transformada.

En lugar de introducir en el sistema una señal de dimensión variable, se toma el espectro en bandas críticas de la señal para introducirlo en la transformada, de esta manera tenemos siempre una dimensión fija de la matriz KLT.

Algunas de las consideraciones importantes para implementar esta transformada en el sistema de reconocimiento son:

1) Se considera un vector de variables aleatorias, donde cada variable aleatoria representa una banda crítica. De esta forma tenemos un vector de 15 renglones, donde cada elemento es un vector de dimensión variable de acuerdo al numero de ventanas que se considere, van a ser representadas en la codificación.

Ec. VI-17

$$X = \begin{bmatrix} X_{BC_1}(m) \\ X_{BC_2}(m) \\ X_{BC_3}(m) \\ \vdots \\ X_{BC_{15}}(m) \end{bmatrix}$$

2) Cada variable aleatoria, toma sus valores, de las m, ventanas que se consideren. En si cada variable aleatoria podría considerarse como un proceso aleatorio, donde m es el tiempo.

Con las consideraciones anteriores ahora obtenemos la matriz de covarianzas para el vector anterior.

Ec. VI-18

$$\sum X = E\{XX^T\} - \mu_X \mu_X{}^T$$

El primer termino del lado derecho de la ecuación anterior es la matriz de autocorrelación. Es interesante desarrollar la ecuación anterior, ya que por la naturaleza del vector de variables aleatorias el calculo de esta se complica al calcularla directamente.

Ec. VI-19

$$\sum X = \sigma_{ij} = \frac{1}{m} \sum_{k=1}^{m} (X_i(k) - \tilde{X}_i)(X_j(k) - \tilde{X}_j)$$

donde:

Ec. VI-20

$$\tilde{X}_i = \frac{1}{m}\sum_{k=1}^{m} X_i(k)$$

Observando con cuidado la ecuación anterior, vemos que los subíndices de X, indican la variable aleatoria que se involucra. La k, representa la ventana correspondiente.

Ec. VI-21

$$\sum X = \begin{bmatrix} \sigma_{X_1X_1} & \sigma_{X_1X_2} & \cdots & \sigma_{X_1X_n} \\ \sigma_{X_2X_1} & \sigma_{X_2X_2} & \cdots & \sigma_{X_2X_n} \\ \vdots & \vdots & & \vdots \\ \sigma_{X_nX_1} & \sigma_{X_nX_2} & \cdots & \sigma_{X_nX_n} \end{bmatrix}$$

De la definición de la covarianza.

Ec. VI-22

$$\sum X = \begin{bmatrix} E\{X_1X_1\}-\mu_{X1}\mu_{X1} & E\{X_1X_2\}-\mu_{X1}\mu_{X2} & .... & E\{X_1X_3\}-\mu_{X1}\mu_{Xn} \\ E\{X_2X_1\}-\mu_{X2}\mu_{X1} & E\{X_2X_2\}-\mu_{X2}\mu_{X2} & .... & E\{X_2X_3\}-\mu_{X2}\mu_{Xn} \\ \vdots & \vdots & .... & \vdots \\ E\{X_nX_n\}-\mu_{Xn}\mu_{Xn} & E\{X_nX_n\}-\mu_{Xn}\mu_{Xn} & .... & E\{X_nX_n\}-\mu_{Xn}\mu_{Xn} \end{bmatrix}$$

Podemos observar la posibilidad de separar en dos matrices el resultado anterior

Ec. VI-23

$$\sum X = \begin{bmatrix} E\{X_1X_1\} & E\{X_1X_2\} & .. & E\{X_1X_3\} \\ E\{X_2X_1\} & E\{X_2X_2\} & .. & E\{X_2X_3\} \\ \vdots & \vdots & .. & \vdots \\ E\{X_nX_n\} & E\{X_nX_n\} & .. & E\{X_nX_n\} \end{bmatrix} - \begin{bmatrix} \mu_{X1}\mu_{X1} & \mu_{X1}\mu_{X2} & .. & \mu_{X1}\mu_{Xn} \\ \mu_{X2}\mu_{X1} & \mu_{X2}\mu_{X2} & .. & \mu_{X2}\mu_{Xn} \\ \vdots & \vdots & .. & \vdots \\ \mu_{Xn}\mu_{Xn} & \mu_{Xn}\mu_{Xn} & .. & \mu_{Xn}\mu_{Xn} \end{bmatrix}$$

Podemos ver que el primer termino es la multiplicación de el vector X, por el vector X transpuesto, y el valor esperado la suma de las matrices que se obtiene de la multiplicación de cada vector de bandas críticas. Para la segunda matriz, debemos obtener la media de cada variable aleatoria, ordenarlas como vectores y multiplicar el vector por su transpuesto.

Ec. VI-24

$$\sum X = \frac{1}{m}\sum_{k=1}^{m} \overline{X}(k)\overline{X}(k)^T - \mu_X \mu_X{}^T$$

VI.3.1 Algoritmo para calcular la matriz de covarianzas.

1) Calcular las medias de cada banda critica, y acomodarlas en forma de vector.

2) Multiplica el vector de medias por su transpuestos, para generar una matriz.

3) Multiplicar cada vector de bandas criticas por su transpuesto, e ir sumando estas matrices.

4) Dividir la matriz del punto 3, entre el número de ventanas.

5) Hacer la resta de las matrices del punto 4 menos la matriz del punto 2.

VI.3.2 Interpretación de la transformada KLT.

Ahora obtenemos los valores y vectores propios de la matriz de covarianzas. Existe un significado muy especial para estos valores y vectores. Los valores característicos representan la cantidad de energía que asignamos a cada componente del espacio ortonormal que formamos con las vectores característicos.

De tal manera que una deducción importante es que la suma de los valores característicos es igual a la suma de las varianzas de la diagonal principal de la matriz de covarianzas. Esto debido a que el nuevo espacio que hemos generado es conservador de la energía.

VI.3.3 Algunas notas adicionales de KLT

Existe un caso particular cuando p=.95 que es el coeficiente de correlación de la matriz adyacente de la matriz de covarianzas, entonces es un proceso de Markov de primer orden. [15]. Otra característica de esta matriz de covarianzas es su simetría, lo cual obliga a que los valores característicos sean reales, para el caso de una matriz positiva, obliga a que los valores propios también sean positivos.

VI.4 Segmentación acústica

Al analizar las características de las señales de voz vemos que existen segmentos en los cuales existen características distintivas, algunas de ellas tienen que ver con la energía, periodicidad, frecuencias fundamentales. La idea de segmentar nos ayuda en poder representar a la señal con menor número de información, usando algunos valores significativos para dicho segmento.

VI.4.1 Segmentación usando (MLR)

Surge de la idea de agrupar ventanas de análisis que tienen características estadísticas similares. En concreto obtenemos una función de densidad para un número de ventanas y comparamos cuantas ventanas vecinas tienen una semejanza en dicha función.

La función elegida para comparar señales se voz es una Gausiana debido al teorema de límite central. Los parámetros importantes en una Gausiana son varianza y media, considerando que la media sea constante, podemos entonces recurrir al cociente de máxima verosimilitud, en la estimación de varianzas, haciendo a un lado la media.

La idea básica de MLR, parte del modelo que se sigue para la detección de señales como se muestra en (Fig. 2), la cual describe como comparamos con un umbral que hipótesis es la que más probabilidad tiene de éxito, y con ello sabemos si la ventana tiene mayor semejanza a la anterior o a la siguiente.

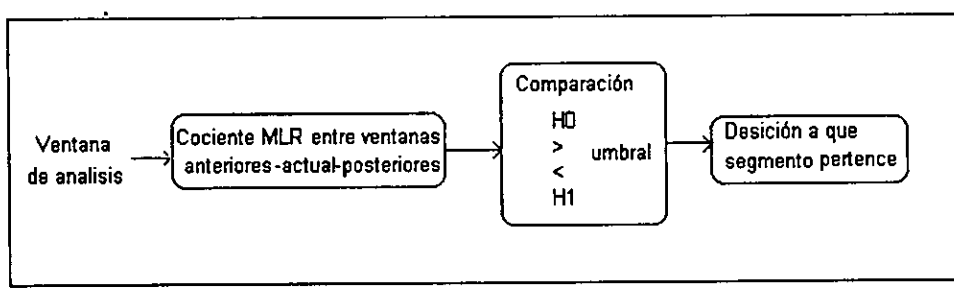

Fig. VI-2 Determinación de si una ventana pertenece o no a un segmento

La teoría expuesta en libros como [16], [17] explican ampliamente como demuestran estas pruebas de hipótesis para la detección de señales, en presencias de ruido blanco o de color como sería en sentido estricto nuestro desarrollo lo cual complicaría la expresión, y los cálculos, sin embargo la función más simple que se obtiene para la determinación de segmentos es un cociente de varianzas como se muestra en la ecuación. (Ec. 25)

Ec. VI-25

$$L(k) = \log\left[\frac{\sigma_{x+1}}{\sigma_x}\right]$$

Existen comentarios que van fuera del alcance de esta tesis, que indican cierta relación entre la transformada KLT, y la detección de ruido con frecuencias particulares, lo cual me motivo a saber un poco más sobre una posible segmentación más redituable.

Algunas características de esta segmentación son:

- Debemos fijar umbrales que determinen los segmentos más notorios.
- Este proceso de segmentación requiere de recorrer varias ocasiones la señal de principio a fin, para buscar y descartar variaciones falsas en la función de máxima verosimilitud.
- La función de máxima verosimilitud por si sola no es capaz de segmentar adecuadamente la señal de voz, y requiere de consideraciones especiales para el caso de ruido y voz, teniendo que detectar antes de segmentar fracciones de silencio en la señal.
- Requiere de la información adicional de niveles de energía, y componentes de alta frecuencia para llevar a cabo su cometido.
- Es la segmentación que más tiempo requiere de las tres mencionadas en esta tesis.
- Requiere de gran cantidad de valores de ventanas muy pequeñas para calcular las estadísticas en ellas, lo que ocasiona problemas de inestabilidad, debidas a un filtrado por ventaneo, muy corto.
- Presenta problemas en transiciones de vocal a vocal debido a que compara varianzas.

VI.4.2 Segmentación por mse-KLT

La segmentación MLR, genera segmentos de voz en base a la consideración de las varianzas. Sin embargo recordando la teoría de KLT, sabemos que el error cuadrático medio de predicción consiste en aquellos valores característicos que no tomamos en cuenta. De tal forma que podemos usar esto para segmentar la señal de voz. Es decir la idea que tuve fue que no tenía caso segmentar únicamente por diferencias de amplitud, sino tomar en consideración la capacidad de la representación que tendría en la codificación KLT, que verdaderamente lo que guardara como coeficientes representase a la señal, lo cual podia saberlo si esta señal podía ser reconstruida.

Elaboré un método consiste en calcular la matriz de covarianzas de cada ventana, obtener sus eigen valores, y verificar que no exceda cierto umbral el error cuadrático medio de predicción. Una vez que se rebasa dicho umbral se toma este punto como fin de segmento. Con lo cual tenemos no sólo la segmentación sino también los coeficientes KLT.

Al comparar los resultados de esta segmentación con MLR, y visual, se observan ciertas semejanzas ventajas y desventajas, la mayor ventaja es que a esta segmentación no le interesa que tan bien se vea sino que siempre conserva un error de reconstrucción, y podemos seleccionar cuantos valores característicos requerimos y en base a ello serán los cortes. Lo cual nos hace tener mejor control sobre el error y la cantidad de coeficientes KLT. Una situación peculiar es que después de un gran tramo de segmento sobre todo ruido, el método arroja un corte, esto se debe a que el ruido es altamente aleatorio y requiere de mayor cantidad de KLT y aumenta el error más rápido debido a la gran cantidad de información que requerimos para reconstruirlo. Por ello es recomendable un buen inicio y fin de palabra para el sistema.

Esta segmentación presenta ciertas características, que consisten lo siguiente:

- Se involucran más variables estadísticas media, covarianza, y varianza.
- La segmentación representa mejor a la señal puesto que los coeficientes KLT por segmento, tienen un error de predicción mínimo, mientras que MLR segmenta sólo en base a

varianzas, haciendo a un lado los parámetros que involucra KLT. De tal manera que cada segmento tiene coeficientes KLT que tienen un error determinado en la reconstrucción y representación de la señal.

- Otra ventaja es que recorremos la señal una sola ocasión
- Como desventaja se requiere de iterar la obtención de eigen valores varias ocasiones lo cual requiere de mayor tiempo de cómputo.
- No requiere información adicional únicamente el espectro que vamos a codificar.
- Obtenemos al mismo tiempo la transformada y la segmentación.

VI.4.3 Segmentación usando función de correlación (Coherencia)

El método anterior tiene la desventaja de recursividad sobre el algoritmo de obtención de valores propios, por ello recurrí a la investigación de una forma de saber el error de antemano en la codificación.

El coeficiente de correlación de dos vectores indica la independencia o dependencia lineal entre ellos. Entre mayor dependencia tengamos entre los vectores que tengamos en la matriz de covarianzas menor es el número de eigen valores que necesitamos para representar a la señal. Por ello podemos adelantarnos al conocimiento de que tan grande será nuestro error cuadrático medio de predicción, midiéndolo de manera indirecta en la dependencia o independencia de los vectores que dan origen a la matriz de covarianzas.

Revisando más en el tema encontré que la expresión obtenida por mí, corresponde a una función que se le a denominado función de coherencia, que se utiliza en detección de señales, con el denominado método de periodograma de Welch, para indicar que frecuencias son alteradas por ruido. Actualmente el uso de esta función esta implementado en paquetes de detección de señales (Matlab5.0 toolkit of signal processing)

La diferencia de su periodograma y mi función muy parecida radica en que en la función coherencia se usan varias ventanas, y en mi función es una correlación entre espectros de

ventana a ventana. Cabe señalar que los espectros manejados los use en una escala logarítmica lo cual nos da ventajas sobre el ruido.

Para Welch utiliza espectros

Ec. VI-26

$$C_{xy} = abs(P_{xy}^{\ 2}) / (P_{xx} P_{yy})$$

Donde usamos la densidad de potencia espectral de dos señales.

Pero aplicado a ventanas, podemos usar el espectro de bandas críticas, considerando cada ventana como señal independiente. Y vemos cierta relación entre el coeficiente de correlación y la función anterior, Welch llegó a esta expresión en base la consideración de la función de transferencia de un sistema y sus densidades de potencia como lo hacemos en probabilidad. Sin embargo yo obtuve una expresión similar por medio de buscar esa independencia o dependencia lineal en la matriz KLT, por el coeficiente de correlación, de las bandas críticas involucradas.

Ec. VI-27

$$C_{m,m+1}(k) = (Bc[m].Bc[m+1]) / (|Bc[m]||Bc[m+1]|)$$

Como podemos ver esta expresión no sólo nos recuerda a la correlación sino que involucra los términos $\sigma_{Bc1,Bc2}$, que se involucran en la matriz de covarianzas. (Ec. 24). Además también nos recuerda que la correlación es un indicador de que tan linealmente es dependiente un vector de otro. Además de que constituye una representación de la cantidad de error que tendríamos al proyectar un vector sobre otro, nosotros hemos conformado una matriz KLT, espacio ortogonal para representar a la señal en Bandas Criticas, entonces la correlación de estos vectores indica que tan bien podemos representar la señal en ese espacio ortonormal de KLT.

VI.5 Distancia de Brown

En el análisis de KLT, podemos ver que obtener los coeficientes KLT, requieren de mayor número de cálculos pues debemos codificar la señal sobre el espacio ortogonal, sin embargo se ha implementado una distancia, equivalente a la de Itakura-Saito para LPC, usando una matriz de vectores y valores propios que de alguna manera hemos visto también representan características de la señal. La relación empleada es la siguiente:

Ec. VI-28

$$d = \frac{\lambda^T (I - A^T A)\lambda}{n}$$

Donde:

I	Matriz identidad
A	A_{ij} = < v1, v2 > ; v1, v2 : son los vectores característicos, y < > producto punto.
Lambda	son los valores característicos ordenados de mayor a menor
n	nivel de análisis (número de valores característicos en el análisis)

Tabla VI-1 Datos usados en el calculo de distancia de Brown

Como podemos ver la fórmula anterior es una distancia no simétrica, ya que involucra los valores característicos de una de las señales no de ambas, por ello no es lo mismo hacer ajuste dinámico con una señal de comparación en las abscisas que intercambiando las señales.

VI.6 Compresión de datos del sistema

Bloque	Longitud de archivo (bytes)	Compresión	Observaciones
Señal digital	22,000	A/D	**Muestreo:** 11025 Hz **Cuantización:** 65536 niveles **Codificación:** 16 bits/muestra
Detección inicio - fin	13,200	6:10	Reducción en un 60% al quitar silencios al inicio y fin de la palabra grabada.
Filtrado	26,400	2:1	Aumenta debido al cambio de manejo numérico de enteros a flotantes (entero 16 bits, flotante 32 bits)
Bandas Criticas	1,680	1:15	**Ventanas:** 256 **Traslape:** 25 **Número de ventanas:** 28 **Espectro BC:** 15*4=60 bytes/ventana
KLT (4 segmentos, 3/7 eigen valores)	768 / 1792	1:2 / 10:9	**Unidad KLT por segmento:** (15 eigen valores + 15 eigen vectores) **Tamaño de matriz eigen vectores:** 15*3*4=180; 15*7*4=420 bytes **Tamaño de vector eigen valores:** 3*4=12; 7*4=28 bytes
Compresión	**Total**	**1:28 / 1:12**	

Tabla VI-2 Compresión (aproximada) de información del sistema

En el sistema descrito tenemos una compresión de datos como la mostrada en la (Tabla 2), que se obtuvo usando la longitud promedio de una grabación de voz por palabra (aprox. 1 s), calculada de 1000 repeticiones de los dígitos en inglés.

Como podemos ver la compresión para KLT, es mucho mayor que para LPC, siempre y cuando usemos pocos segmentos, y depende mucho de que tantos eigen valores consideremos, por ello es indispensable una buena codificación KLT..

VII. Bases de voz empleadas

Para los experimentos realizados, se usaron:

A) Bases de dígitos en Inglés, desarrollada por Texas Instruments, se utilizaron para medir la capacidad de reconocimiento del sistema, son gran variedad de parlantes, distintos acentos, y niveles de ruido variable, codificada con 16 bits por muestra, que consta de dos módulos:

1) Una base conocimiento con: 10 números (0-9), pronunciados por diez parlantes distintos, con 10 repeticiones por parlante (total 1000 palabras)

2) Una base de prueba de: 10 números (0-9), diez parlantes distintos, con 16 repeticiones por parlante (total 1600 palabras),

B) Una base en español, que consta de 1400 palabras; 10 números (0-9), 14 repeticiones, 10 parlantes, codificada con 16 bits por muestra. Las bases en español fueron realizadas por un trabajo anterior, y se implementaron en el sistema de reconocimiento para comprobar la validez del método al cambiar de idioma.

C) Una base generada para locutores durante la exposición, con 10 repeticiones por palabra, codificada con 16 bits por muestra. La base realizada para la exposición fue necesaria para poder involucrar parlantes que demostraran durante la presentación el funcionamiento del sistema.

VIII. Resultados y Análisis

VIII.1 Resultados de Segmentación y análisis.

A continuación se muestran ejemplos de segmentación usando dos algoritmos: MLR y el otro coeficiente de autocorrelación de espectro. Las escalas de las figuras y alineación son importantes para el análisis.

1) **Palabra "ONE", W11KAB4**

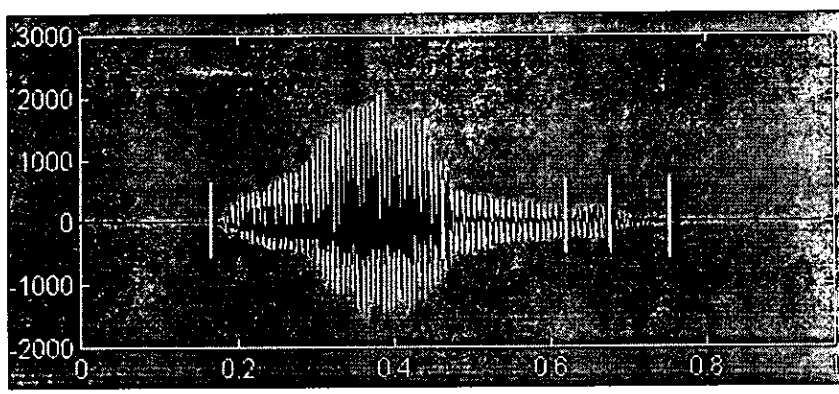

Fig. VIII-3 Señal en el tiempo Cortes con MLR

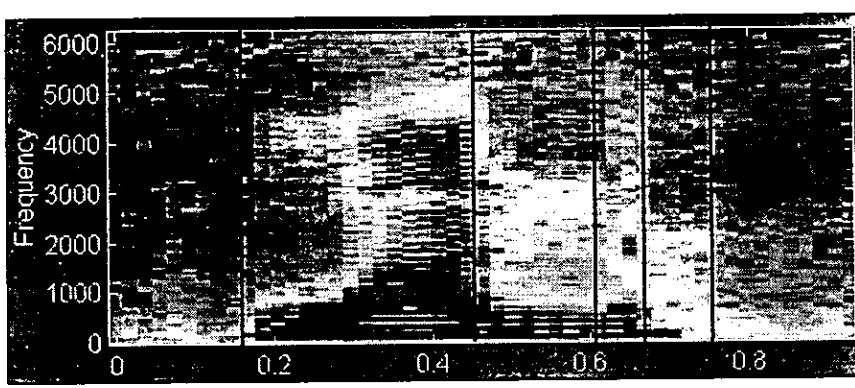

Fig. VIII-4 Espectograma con cortes MLR

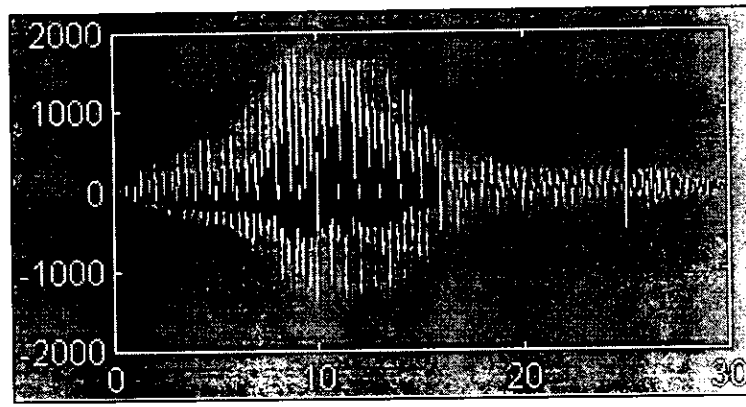

Fig. VIII-5 Segmentación correlación de espectro inicio fin

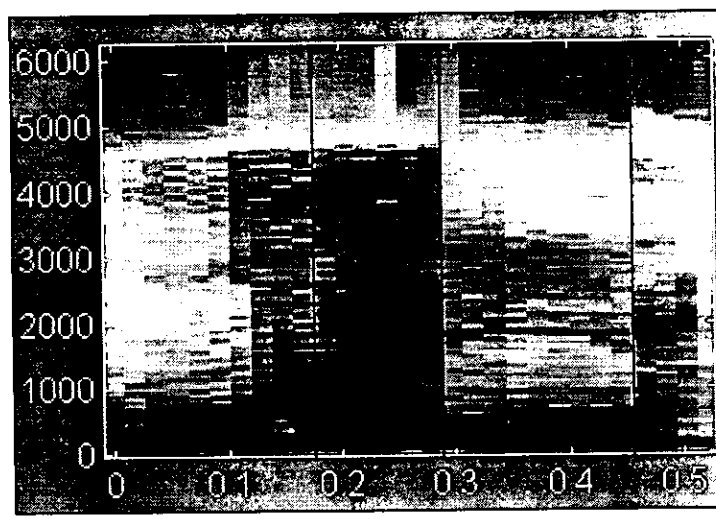

Fig. VIII-6 Espectograma de señal con inicio fin, cortes por correlación

Como podemos ver MLR deja un residuo de señal al final y prefiere el segmento resonante 'n' para segmentar y variaciones de amplitud de dicha resonancia, mientras que correlación si detecta el diptongo "ua" separándolo, esto se explica por lo siguiente; la autocorrelación indica periodicidad de la señal, una correlación cruzada refleja mejor cambios de periodicidad. Por ello es que Coherencia si detecta diptongos y MLR no pues el último sólo considera varianzas.

En el método de correlación de espectros se tiene que recurrir a la detección de inicio - fin anteriormente descrita, debido a lo ya visto en la sección VI.4.2 , para el caso de segmentos de ruido largo.

2) **Palabra "SEVEN" , W17KAB4**

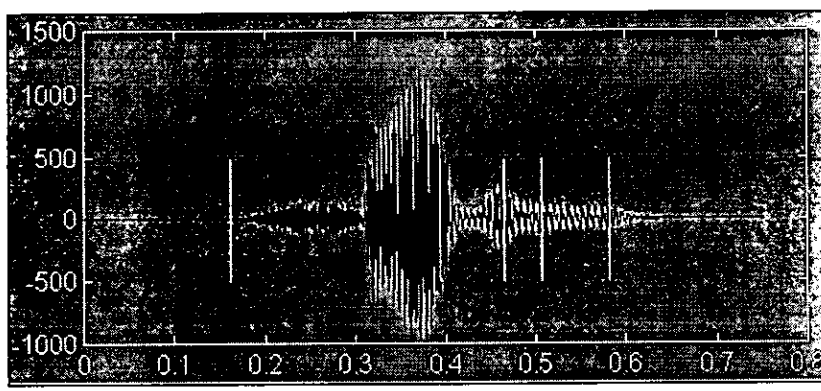

Fig. VIII-7 Señal en el tiempo segmentación con MLR

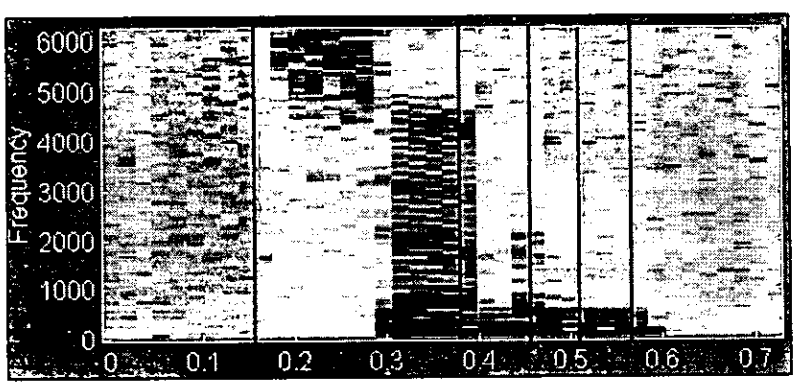

Fig. VIII-8 Espectograma segmentado con MLR

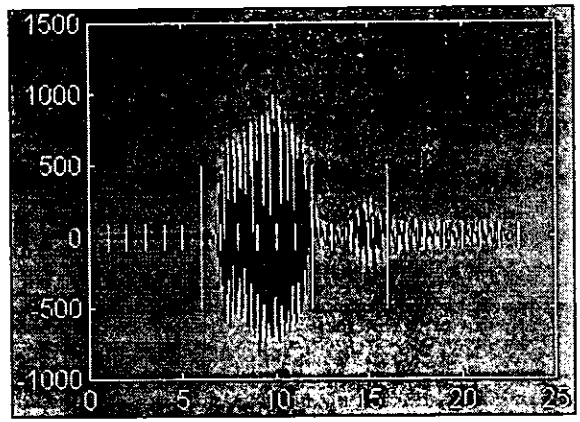

Fig. VIII-9 Señal el el tiempo segmentación por correlación

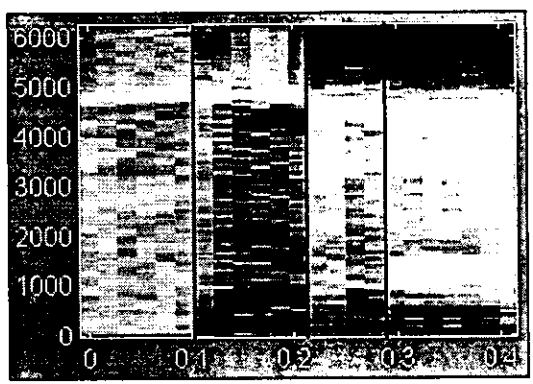

Fig. VIII-10 Espectro recortado segmentado por correlación

Como podemos observar para MLR, la transición de 's-e', no es detectada esto debido a que la 's' baja su energía al final, y la 'e' comienza suavemente, mientras que el segundo método si detecta este segmento, fortalecido por la correlación (cambio de periodicidad). Sin embargo observamos una desventaja en el método de correlación que se debe al tamaño de las ventanas, nuevamente regresamos al problema de escoger entre resolución en frecuencia y detectar cambios.

Es interesante el hecho de pedirle al programa un mayor número de segmentos al método de correlación para esta palabra "seven". Como se muestra a continuación en (Fig. 11).

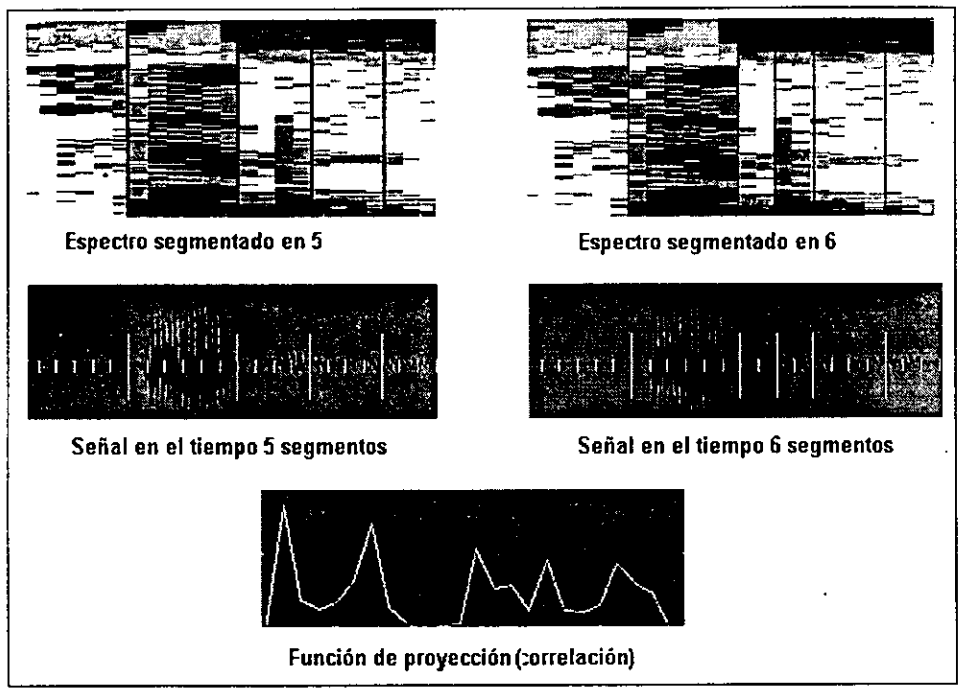

Fig. VIII-11 Segmentación con 5-6 segmentos

Podemos observar como se ve la función de correlación en la cual los picos indican variaciones importantes esta función y corresponden a donde debe segmentarse es una función de variación de la correlación.

3) **Palabra "ZERO"** usando el máximo número de segmentos con el método de coherencia

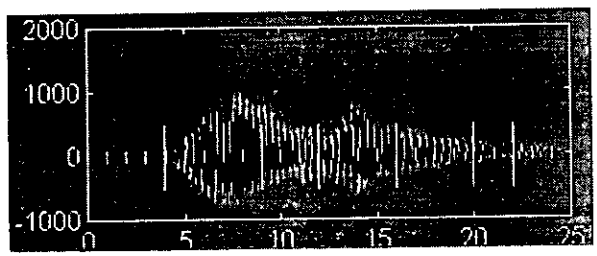

Fig. VIII-12 Señal en el tiempo segmentada libremente

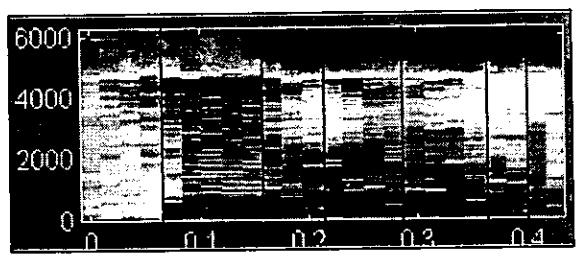

Fig. VIII-13 Espectro con segmentación

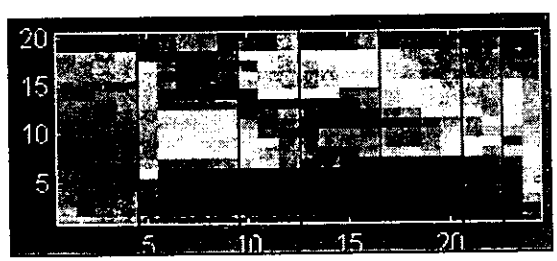

Fig. VIII-14 Bandas Críticas con la segmentación

La palabra "Zero" es una de las palabras con mayor número de cambios y vemos en la (Fig 14) su excelente desempeño del método de correlación. Como podemos observar la (Fig 13) con respecto a la (Fig 14) es como un espectro algo borroso, puesto que en la segunda ya hemos agrupado los puntos de la FFT, en bandas críticas.

VIII.2 Resultados Sistema LPC

Las tablas que a continuación se presentan son el resultado de varios experimentos de los cuales se escogieron aquellos el los cuales se lograron mejores resultados.

Resultados 1: Usando un filtro de preénfasis a la entrada, orden del predictor LPC p=8, coeficientes de correlación normalizados, inicio-fin según Rabiner-Sambur, ventanas de 128 en todo el sistema translape 18 puntos, distancia de Itakura-Saito y DTW con una región Alhamed de r=8, m=[0.6, 1/0.6].

	0	1	2	3	4	5	6	7	8	9
0	160	0	0	0	0	0	0	0	0	0
1	0	159	0	0	0	0	0	0	10	0
2	0	0	160	0	0	0	0	0	0	0
3	1	0	0	159	0	0	0	0	0	0
4	1	0	0	0	159	0	0	0	0	0
5	0	0	0	0	0	160	0	0	0	0
6	0	0	0	0	0	1	159	0	0	0
7	0	0	0	0	0	0	0	160	0	0
8	0	0	0	0	0	0	0	0	159	0
9	0	0	0	0	0	0	0	0	0	160

Tabla VIII-1 Sistema LPC con r=8

Tiempo de reconocimiento: **90 seg/palabra.**

Porcentaje de reconocimiento: **99.68 %**

El sistema LPC presenta las ventajas dichas pero observamos que altos resultados son a costa de tiempo y cantidad de memoria en la computadora, aún cuando parece tener ventajas el tiempo es alto en relación con KLT, de mayor dificultad de implementación.

VIII.3 Resultados sistema KLT

Resultados 2: Usando segmentación función coherencia, con el tercer valor caracterísitco (l_3<0.001) entonces el 99% de la energía se encuentra en los primeros 3 valores propios. Con lo cual hemos fijado un umbral. Se uso: filtro de preénfasis, filtro ancho de banda telefónico, espectro en dB, 15 bandas críticas, Warping r=5.

	0	1	2	3	4	5	6	7	8	9
0	100	0	0	0	0	0	0	0	0	0
1	0	100	0	0	0	0	0	0	0	0
2	0	0	100	0	0	0	0	0	0	0
3	0	0	0	99	0	0	0	0	1	0
4	0	1	0	0	99	0	0	0	0	0
5	0	0	0	0	0	100	0	0	0	0
6	0	0	0	1	0	0	99	0	0	0
7	0	0	0	0	0	0	1	99	0	0
8	0	0	0	0	0	0	0	0	99	1
9	0	2	0	0	0	0	0	0	0	98

Porcentaje de reconocimiento: 99.43%
Tiempo de reconocimiento: 27.7 s/palabra

Resultados 3: 8 segmentos fijos, región de warping=5

	0	1	2	3	4	5	6	7	8	9
0	95	0	3	0	0	1	0	0	0	1
1	0	90	1	1	0	1	0	0	0	5
2	0	0	97	0	0	0	0	2	0	1
3	0	0	1	98	0	0	0	2	0	0
4	0	0	0	1	98	0	0	1	0	0
5	0	2	0	0	1	96	2	1	1	0
6	0	0	0	0	0	2	95	0	3	0
7	0	0	0	0	0	0	0	99	0	1
8	0	0	0	1	0	0	0	0	99	0
9	1	4	0	0	0	0	0	2	0	93

Porcentaje de reconocimiento: 96.0%
Tiempo de reconocimiento. 54 seg./palabra

Resultados 4: 5 segmentos fijos, region de warping=4

Con la base de conocimiento

	0	1	2	3	4	5	6	7	8	9
0	94	0	3	0	0	0	1	2	0	0
1	0	86	0	0	0	1	0	4	0	8
2	0	0	93	2	0	0	0	3	2	0
3	1	0	2	90	0	0	1	4	0	2
4	0	0	0	0	97	1	0	1	0	1
5	0	2	0	0	2	88	3	4	0	1
6	0	0	0	1	0	0	92	3	4	0
7	0	0	0	0	0	2	3	95	0	0
8	0	0	0	9	0	2	4	0	85	0
9	2	6	0	0	0	1	0	1	0	90

Porcentaje de reconocimiento: 91.0%
Tiempo de reconocimiento: 46.22 seg./palabra

Resultados 5: Usando una segmentación libre pero con la función coherencia, con un valor de error que reflejara en alguna palabra una segmentación creíble.

Base de conocimiento

	0	1	2	3	4	5	6	7	8	9
0	90	0	3	0	0	1	0	1	0	5
1	1	83	1	1	5	0	0	1	0	8
2	0	0	94	4	0	0	0	2	0	0
3	0	0	1	95	0	0	0	3	1	1
4	0	0	0	0	97	2	0	0	0	1
5	0	2	0	1	1	89	1	4	0	2
6	0	0	0	0	0	2	91	5	2	0
7	0	0	0	0	0	0	0	100	0	0
8	0	0	1	0	0	0	1	1	97	0
9	1	5	1	0	0	0	0	2	0	91

Porcentaje: 92.7%
Tiempo de reconocimiento 38.69 seg./palabra

Resultados 6: (Prueba) Preénfasis, filtro 4.5 K, segmentación libre detectando todos los picos de error en la función coherencia,

	0	1	2	3	4	5	6	7	8	9
0	147	0	6	1	2	0	0	3	1	0
1	0	152	0	0	1	1	0	2	0	4
2	0	0	154	5	0	0	0	1	0	0
3	0	0	1	149	0	1	0	5	3	1
4	0	0	0	0	159	1	0	0	0	1
5	0	2	0	0	1	149	1	9	0	0
6	0	0	0	0	0	6	149	3	2	0
7	0	1	0	0	0	3	1	155	0	0
8	0	0	1	1	0	0	1	1	158	0
9	2	10	1	0	0	7	1	2	0	137

porcentaje reconocimiento 94.31%
tiempo: 67.30 s/palabra.

Resultados 7: (Prueba)

Parámetros: Pendientes de Warping(.5,2), filtro 4.5K, pre-énfasis .95, umbral de segmentador=0 .0001

	0	1	2	3	4	5	6	7	8	9
0	154	0	2	0	0	0	0	4	0	0
1	0	145	0	0	5	1	0	4	0	5
2	1	0	149	3	0	0	0	6	0	1
3	0	0	0	157	0	0	0	2	0	0
4	0	0	0	0	159	0	0	1	0	1
5	0	2	0	0	0	154	2	2	0	2
6	0	0	0	0	0	1	159	0	0	0
7	0	0	0	0	1	2	0	157	0	0
8	0	0	1	1	0	1	4	2	150	1
9	1	0	0	0	0	1	0	16	0	142

Porcentaje de reconocimiento 95.43%
Tiempo: 67.57 s/palabra

Resultados 8 (prueba):

Segmentador **MLR 3** valores característicos.

	0	1	2	3	4	5	6	7	8	9
0	126	2	4	2	2	2	1	12	0	9
1	0	103	1	4	3	6	2	23	1	17
2	3	0	119	11	3	6	1	8	7	2
3	0	1	4	128	0	4	5	7	8	3
4	3	2	0	0	134	14	2	3	0	2
5	0	3	0	1	12	128	7	6	3	0
6	0	0	1	2	1	22	116	8	10	0
7	1	1	2	9	7	13	0	123	1	3
8	1	0	1	6	1	5	29	6	111	0
9	3	17	2	3	1	7	1	13	1	111

Porcentaje de Reconocimiento 75.01%
Tiempo warping 36.67 s/palabra
Tiempo segmentador 10 seg./palabra

Resultados: 9 (Prueba)

Segmentador **MLR, 7** valores característicos

	0	1	2	3	4	5	6	7	8	9
0	150	0	2	1	1	0	0	2	0	4
1	0	140	1	2	1	2	0	5	0	9
2	0	0	147	7	0	1	0	3	2	0
3	0	0	2	152	0	1	3	1	0	1
4	1	1	0	0	148	7	0	2	0	1
5	0	2	0	0	5	146	3	3	1	0
6	0	0	0	1	0	9	141	4	5	0
7	0	1	0	4	3	9	0	142	0	1
8	0	0	1	3	0	2	10	2	142	0
9	2	10	1	0	0	7	1	6	0	133

Porcentaje de reconocimiento 90.5%
Tiempo warping 58s/palabra
Tiempo segmentador 10seg/palabra

Resultados 10: (Prueba)

Tabla de resultados usando **MLR, libre** con 7 valores característicos

	0	1	2	3	4	5	6	7	8	9
0	151	0	4	0	1	0	0	3	1	0
1	0	150	0	0	2	1	0	1	1	5
2	0	0	154	1	0	0	0	2	1	2
3	0	0	4	151	0	0	0	2	2	1
4	1	1	0	0	156	1	0	0	0	1
5	0	0	0	0	0	153	0	1	0	6
6	0	0	3	5	0	0	144	4	4	0
7	0	0	1	0	1	0	0	156	1	1
8	0	0	6	1	0	0	1	0	151	1
9	0	5	1	1	0	1	0	0	0	151

Reconocimiento 94.81%
Tiempo: 165 s/palabra

Experimento 11: Resultados usando coherencia **Base en Español.**

	0	1	2	3	4	5	6	7	8	9
0	139	0	0	1	0	0	0	0	0	0
1	1	138	0	0	1	0	0	0	0	0
2	1	3	136	0	0	0	0	0	0	0
3	1	0	0	137	0	0	2	0	0	0
4	0	0	0	0	140	0	0	0	0	0
5	0	0	0	0	0	140	0	0	0	0
6	0	0	0	3	0	0	137	0	0	0
7	1	0	0	1	0	0	0	138	0	0
8	0	0	1	0	2	0	0	0	137	0
9	0	0	0	1	0	0	0	0	0	139

Reconocimiento 98.64%.

Aún cuando el nivel de ruido es considerable aumenta el reconocimiento debido a que las palabras en español no se confunden tanto en los dígitos.

Conclusiones

En general los sistemas de reconocimiento cada vez van logrando robustecerse, sin embargo aún no esta totalmente terminado el trabajo en esta área, los dos sistemas desarrollados presentan buenos resultados, he buscado en el desarrollo de los programas la mínima cantidad de código, el uso mínimo de recursos y la mejor depuración de rutinas para que su ejecución sea lo más rápida posible. Comparado con otros trabajos desarrollados, ambos sistemas han reducido el tiempo de reconocimiento considerablemente, así como los requerimientos de sistema. Llegando a ejecutarse en una PC convencional.

Aún cuando actualmente los sistemas de reconocimiento automático se orientan al uso de palabras interconectadas, debido a la implementación de sistemas con características lingüísticas. El uso de los sistemas presentados para palabras aisladas, tienen aún futuro, pues usando métodos de segmentación de palabras más precisos, es factible usar el reconocimiento de palabras aisladas.

Ambos sistemas presentan características de análisis en frecuencia y tiempo, la implementación del sistema LPC es más sencilla, mientras que KLT debido al uso de matrices hace necesario el uso de algoritmos más complicados. No existe una variación importante en la velocidad de procesamiento de uno y otro sistema esto debido a que en LPC la puntos por señal que entran al ajuste dinámico es mayor, pero en KLT el cálculo de la distancia aún cuando son menor número de puntos es más lenta aunado a la obtención de coeficientes que involucran el cálculo de los valores y vectores propios de la matriz de covarianzas.

Los resultados obtenidos son alentadores. Los sistemas implementados pueden ejecutarse en una PC convencional, lo cual ya es un paso más. Sin embargo cabe señalar que aún falta mucho por hacer en trabajos futuros. Debido a las observaciones realizadas en cada uno de los sistemas me-parece que es viable continuar con el perfeccionamiento de estos dos sistemas, sugiriendo la implementación de los siguientes puntos:

- Creación de un sistema de cancelación de ruido para hacer más robusto el sistema, tal como LMS algoritmo de Widrow, las tarjetas de sonido comerciales actualmente tienen dos canales para cumplir con la transmisión full-duplex, de tal forma que es viable la implementación de un cancelador de ruido usando un canal para recepción de la señal de voz, y el otro para la grabación de ruido cercano a la fuente, tal como se ha implementado en cabinas de aviones o trenes.

- Los sistemas electrónicos son más eficientes que la ejecución de programas. Además en cuestiones de filtros, obtención de LPC, correlaciones y otras operaciones, son más fácil de implementar con circuitos integrados que usando programas. Por lo cual se sugiere el uso de hardware para la realización de algunas operaciones en el sistema de reconocimiento.

- En el sistema KLT se utilizó la DFFT, que presenta ciertas imprecisiones ya comentadas, por ello se recomienda la implementación de un banco de filtros para la obtención de bandas críticas, esto también es más sencillo de implementar por medio de Hardware.

- Para la aplicación comercial, es necesario crear interfaces que hagan útil estos algoritmos, tales como: Procesadores de textos con reconocimiento, Marcación telefónica, Ejecución de instrucciones, y mayor número de aplicaciones.

- Un mayor estudio en el modelo del oído humano, considerando los umbrales de audición para una conversación y posible implementación de algunas reglas para la simplificación de componentes inaudibles.

- Implementación de reglas fonéticas que ayuden al reconocimiento.

- Un estudio más profundo del método de segmentación usando KLT y funciones de coherencia, ya que pueden ser como he demostrado un posible camino para la condensación de información.

Bibliografía

Libros de Texto

1)	**Automatic Speech Recognition SPHINIX System,** Prentice Hall,.1991	TK7882. S65 Fac. Ciencias
2)	Waibel & Lee, **Readings in Speech Recognition,** Morgan Kaufmann, 1990	TK7882.S65 R42 Posgrado F.I.
3)	B. Scharf, **Foundations of Modern Auditory Theory,** Academic Press Vol. I, II, 1970.	QP461 T6 Fac. Ciencias
4)	Chris Rowden, **Speech Procesing,**	TK7882.865
5)	Elliot-Rao, **Fast Transforms Algorithms, Analyses, Applications,** Academic Press, 1982	QA403.5 E54 Fac. Ciencias
6)	Embree, Paul & Kimble, Bruce, **C Language Algorithms For Digital Signal Processing,** Prentice Hall, 1990.	Fac. Ciencias
7)	Goldberg, **Matrix Theory with Applications,** Mc. Graw Hill	QA188. G637
8)	Jayant & Noll, **Digital Coding of Waveforms,** Prentice Hall, 1984.	TK5102.5 J28
9)	L. Rabiner **Fundamentals of Speech Recognition**, Prentice-Hall, 1993	Posgrado F. I.
10)	Oppenheim-Shafer, **Digital Signal Processing,** Prentice Hall, 1975	
11)	P.Papamichalis,**Practical Approaches to Speech Coding,** Prentice-Hall,1987.	Posgrado F.I.
12)	Proakis, **Advanced Digital Signal Processing**, Prentice Hall	Ciencias

13)	Proakis, **Digital Signal Processing**, McMillan	Posgrado F.I.
14)	Rao-Yip, **Discrete Cosine Transform**, MacMillan	TK5102.5 R36 Fac. Ciencias
15)	S.Rinath Rajasekaran, **Introduction to Statistical Signal Processing with applications**	TK5102.5 S74 Fac. Ciencias
16)	Scharf, Louis L., **Statistical Signal Processing; Detection, Estimation, And Time Series Analysis**, Addison Wesley, 1991	Fac. Ciencias
17)	Shanmugan Breipohl K, **Random Signals: Detection Estimation and Data Analysis**	TK5102.5 S43 Fac. Ciencias
18)	T. Robinson, **Speech Analysis, Computer Speech and Language Processing**, Cambridge University 1996.	Posgrado F.I.
19)	Van Bergeijk, **Las Ondas y el Oido**	QP461. V33 Fac. Ciencias

Artículos:

20) Hiroaki Sakoe & Seibi Chiba, **Dynamic Programming Algorithm Optimization for Spoken Word Recognition**, IEEE Transactions on Acoustics, Speech and Signal Processing, Febrero 1978.

21) Itakura, **Minimum Prediction Residual Principle**, IEEE Transactions on Acoustics, Speech and Signal Processing, Febrero 1975

22) L. Rabiner et al, **Considerations in Dynamic Time Warping Algorithms for Discrete Word Recognition**, IEEE Transactions on Acoustics, Speech and Signal Processing, Diciembre 1978.

23) L. Rabiner et al, **Time Warping Algorithms for Word Recognition**, IEEE Transactions on Acoustics, Speech and Signal Processing, Diciembre 1978