EXTRACCION DE CARACTERÍSTICAS

Habiendo revisado la teoría básica del procesamiento digital de señales, estaremos en condición de plantear algoritmos para poder procesar las señales de habla, el objetivo principal de este tema será hallar los patrones de características, para esto revisaremos un modelo de extracción de características llamado MFCC,

1. Algoritmos de extracción de características

Existen algunos algoritmos extractores de características en los que mencionaremos los siguientes:

- Algoritmo de características acústico fonéticas
- Algoritmo basado en trasnsformada de Fourier (características espectrales)
- Algoritmo basado en coeficientes cepstrales
- Codificación Predictiva Lineal (LPC)
- LPC - cepstrum
- Coeficientes Cepstrales en escala Mel (MFCC)
- Predicción Perceptual Lineal (PLP)
- MFCC con longitud del tracto vocal normalizada
- Algoritmos basados en wavelets

1.1 Algoritmos de características acústico fonéticas

Los algoritmos de extracción de características de este tipo sugieren extraer información por frame de la señal y parametrizar cada frame de acuerdo a un conjunto acciones del hablante como fricación nasal, voz, etc, o propiedades físicas relacionadas al modelo fuente filtro como ubicación de las formantes, ancho de las formantes, energias de las frecuencias, etc

1.2 Algoritmo basado en transformada de Fourier (características espectrales)

Se suelen usar los coeficientes obtenidos de la trasformada discreta de Fourier de manera directa, la desventaja de estos algoritmos es que no hacen uso del modelo fuente filtro, que indica que la fuente (exitación, pitch), está con convolución (multiplicación en el dominio de la frecuencia) con los componentes del tracto vocal (filtro), , no se aprovecha la información del pitch (fuente)

1.3 Algoritmo basado en coeficientes cepstrales

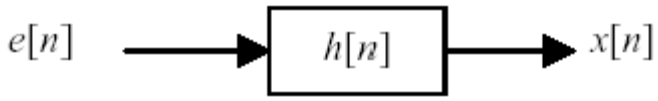
El modelo fuente filtro establece que la exitación está en convolución con la filtro, es decir con el tracto vocal. ¿Cómo extraer la información presente en la fuente y en el filtro de manera independientes?, es decir separar las dos señales.

Para lograr esto se realiza un proceso o una trasformación cuyo objetivo es desconvolucionar las señales, es decir un proceso inverso al operador convolución

A continuación se detalla un modelo llamado Cepstrum o Cepstro, cuyo nombre fue formado haciendo un reverso de las palabras spectrum.

1.3.1 Procesamiento Cepstral

Como se conoce la excitación e[n] representa a la frecuencia fundamental que se produce en las cuerdas vocales y el filtro h[n] representa las resonancias del tracto vocal dadas por los labios, faringe, dientes, paladar, etc., que cambian sobre el tiempo.



la idea es convertir la convolución

$$x[n] = e[n] * h[n]$$

en una suma

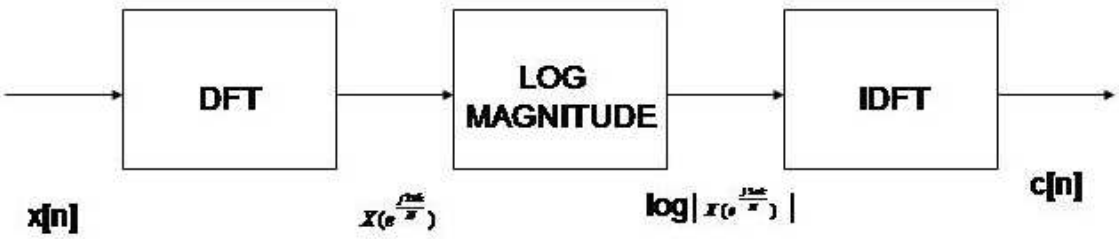
$$\hat{x}[n] = \hat{e}[n] + \hat{h}[n]$$

Se define Cepstrum como una trasformación que nos va a permitir separar la fuente de excitación del filtro (glotis), se asumirá un valor N para el cual el cepstrum del filtro (glotis) $\hat{h}[n] \approx 0$ para $n \geq N$ y el el cepstrum de la excitación $\hat{e}[n] \approx 0$ para $n < N$ asumiendo, esto se podrá recuperar aproximadamente $\hat{h}[n]$ y $\hat{e}[n]$ de $\hat{x}[n]$.

El cepstrum $D[]$ de una señal digital $x[n]$ esta definido:

$$\hat{x}[n] = \frac{1}{2\pi} \int_{-\pi}^{\pi} \ln |X(e^{jw})| e^{jwn}$$

El análisis de $\hat{x}$ nos permitirá conocer información del tracto vocal que se encuentra en la parte baja del cepstrum y la información del filtro (glotis) contenida en la parte alta del cepstrum, luego se puede separar fácilmente $e[n]$ de $h[n]$ asumiendo el valor N antes mencionado y haciendo la operación inversa $D[]^{-1}$ del cepstrum.



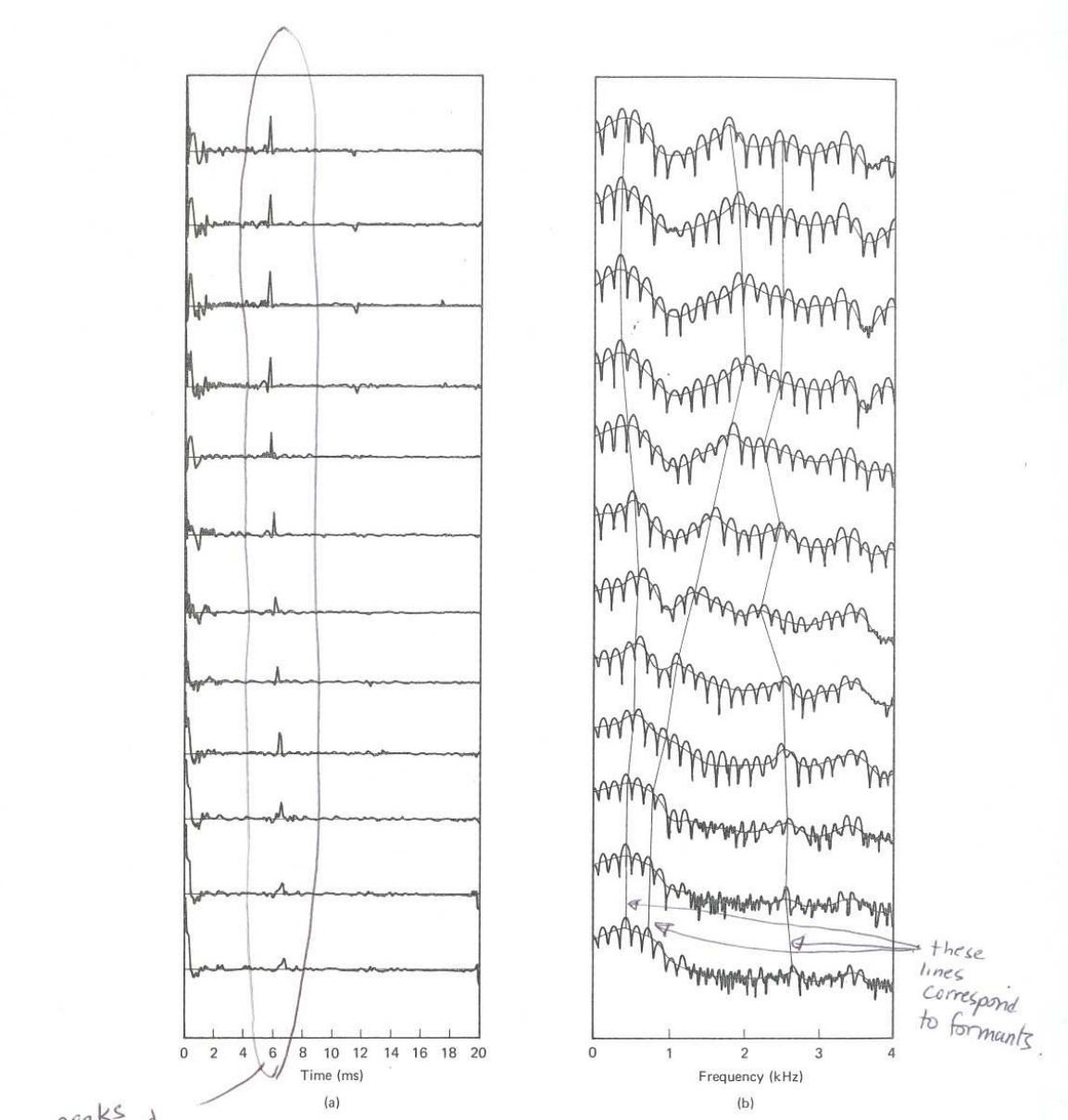


Figure 12.28 (a) Cepstra and (b) log spectra for sequential segments of voiced speech. Original with cepstrally smoothed superimposed.

Figure from Oppenheim + Schafer. "Discrete-Time Signal Processing"

Coeficientes Cepstrales de la señal de habla, la parte baja corresponde al tracto vocal, la parte alta corresponde a la información provenientes de las cuerdas vocales. Fuente: Oppenheim

1.4 Codificación Predictiva Lineal

Supongamos que una señal ***x[n]***, puede ser vista como la combinación lineal de las ***p*** muestras pasadas por ciertos coeficientes más un error

$$x[n] = \sum_{k=1}^{p} a_k x[n-k] + e[n]$$

y podemos predecir la señal de la siguiente manera:

$$\tilde{x}[n] = \sum_{k=1}^{p} a_k x[n-k]$$

entonces el error será:

$$e[n] = x[n] - \tilde{x}[n] = x[n] - \sum_{k=1}^{p} a_k x[n-k]$$

Si analizamos una vecindad de tamaño ***N***, para una muestra ***m***, tenemos

$$x_m[n] = x[m+n]$$

Se puede definir el error para este segmento como:

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

Debemos tratar de minimizar el error con respecto a a_i, para esto derivamos el error del segmento con respecto a esta variable y la igualamos a cero

Lo cual nos produce el siguiente resultado

$$\sum_n e_m[n] x_m[n-i] = 0 \qquad 1 \le i \le p$$

esta ecuación puede ser puede ser expresada como un conjunto de ***p*** ecuaciones lineales

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

Podemos definir los coeficientes de correlación de la siguiente manera

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

obteniendo una ecuación conocida como ecuaciones *Yule-Walker*

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

Existen algunos algoritmos para solucionar estas ecuaciones entre los cuales se tienen:

- Método de covarianza
- Método de autocorrelacion
- Formulación Látice

1.4.1. Método de autocorrelación

Se puede mostrar

$$\phi_m[i,j] = R_m[|\, i-j \,|]$$

lo cual corresponde a

$$\sum_{j=1}^{p} a_j R_m[|\, i-j\,|] = R_m[i]$$

interpretando este conjunto de ecuaciones como matriz tenemos

$$\begin{pmatrix} R_m[0] & R_m[1] & R_m[2] & \cdots & R_m[p-1] \\ R_m[1] & R_m[0] & R_m[1] & \cdots & R_m[p-2] \\ R_m[2] & R_m[1] & R_m[0] & \cdots & R_m[p-3] \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ R_m[p-1] & R_m[p-2] & R_m[p-3] & \cdots & R_m[0] \end{pmatrix} \begin{pmatrix} a_1 \\ a_2 \\ a_3 \\ \cdots \\ a_p \end{pmatrix} = \begin{pmatrix} R_m[1] \\ R_m[2] \\ R_m[3] \\ \cdots \\ R_m[p] \end{pmatrix}$$

que es una matriz simétrica con una diagonal donde todos los elementos son iguales, eta matriz se llama *Toeplitz* que es solucionado por un algoritmo llamado la recursión de Durbin's

Algoritmo:

1. Inicialización

$$E^0 = R[0]$$

2. ***For i=1,..,p***

$$k_i = \left(R[i] - \sum_{j=1}^{i-1} a_j^{i-1} R[i-j] \right) / E^{i-1}$$

$$a_i^i = k_i$$

$$a_j^i = a_j^{i-1} - k_i a_{i-j}^{i-1}, \qquad 1 \le j < i$$

$$E^i = (1-k_i^2) E^{i-1}$$

3. Solución Final

$$a_j = a_j^p \qquad 1 \le j \le p$$

1.4.2. Analisis de espectro con LPC

Se define un error de predicción normalizado de la siguiente manera:

$$\sum_n u_m^2[n] = 1$$

y una ganancia:

$$e_m[n] = G u_m[n]$$

en consecuencia podemos calcular la ganancia

$$E_m = \sum_n e_m^2[n] = G^2 \sum_n u_m^2[n] = G^2$$

analizando en el dominio de la frecuencia el comportamiento del análisis LPC tenemos

$$H(e^{j\omega}) = \frac{G}{1 - \sum_{k=1}^{p} a_k e^{-j\omega k}} = \frac{G}{A(e^{j\omega})}$$

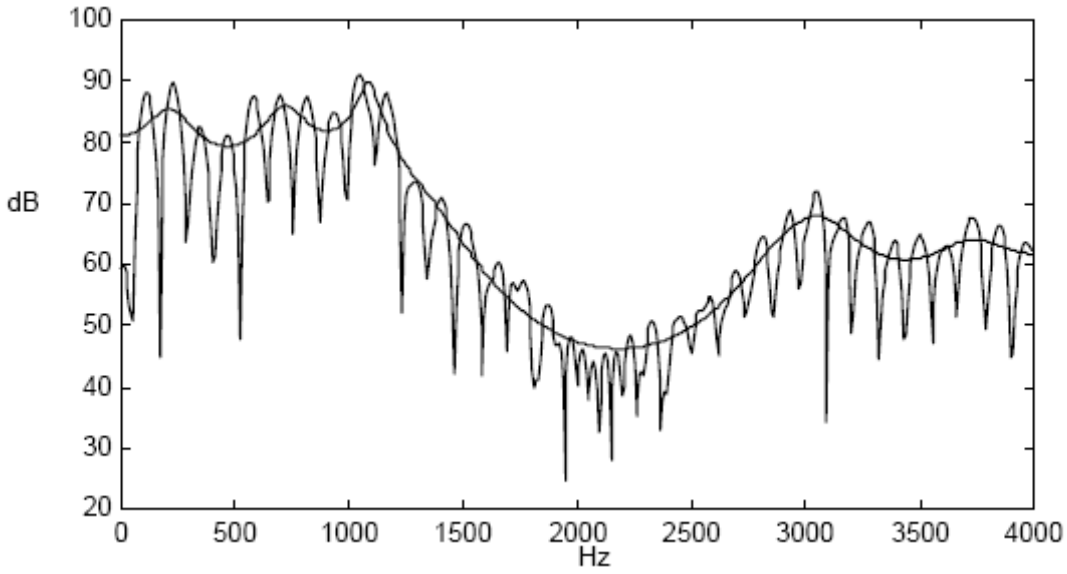


Figura: LPC del fonema /*ah*/ de la palabra *lives*, con ventana Hamming de 10ms y p=14

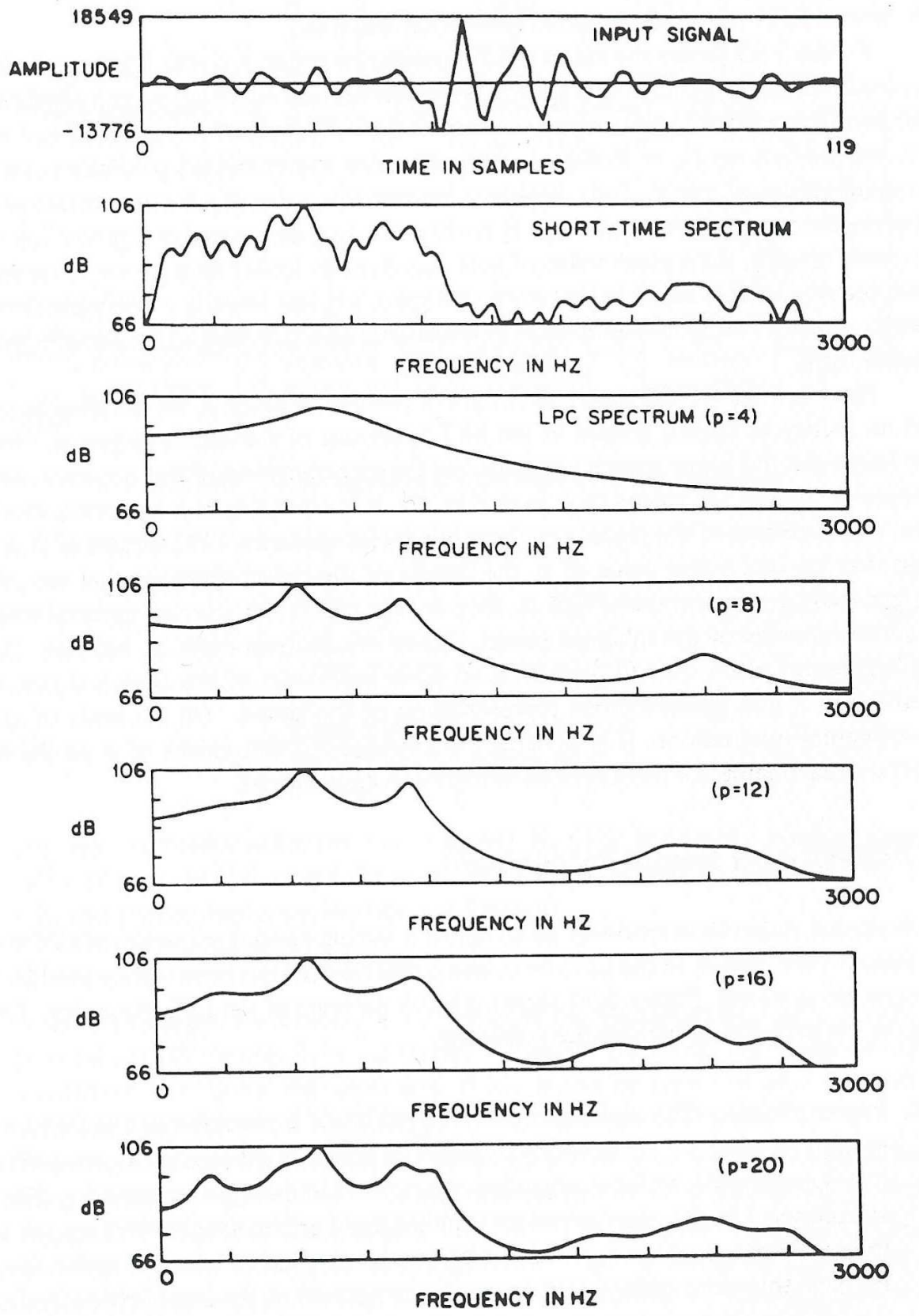


Figura: conforme p aumenta el modelo LPC progresivamente aproxima al espectro original, se debe utilizar aproximadamente un número de p's con la siguiente formula Fs/1khz +2-4 , por ejemplo para 10khz un p aproximadamente de 12 o 14

1.4.3 LPC y Reconocimiento automático del habla

Los coeficientes a's por si mismos, tienen un rango muy dinámico, varían demasiado con pequeños cambios en la señal, sin embargo existen varias trasformaciones como los coeficientes de reflexión, log area ratios, LSP parámetros; la trasformación que mejor trabaja es el LPC cepstrum

1.5 LPC Cepstrum Tema de investigación

1.6 Coeficientes Cepstrales en Escala Mel (MFCC)

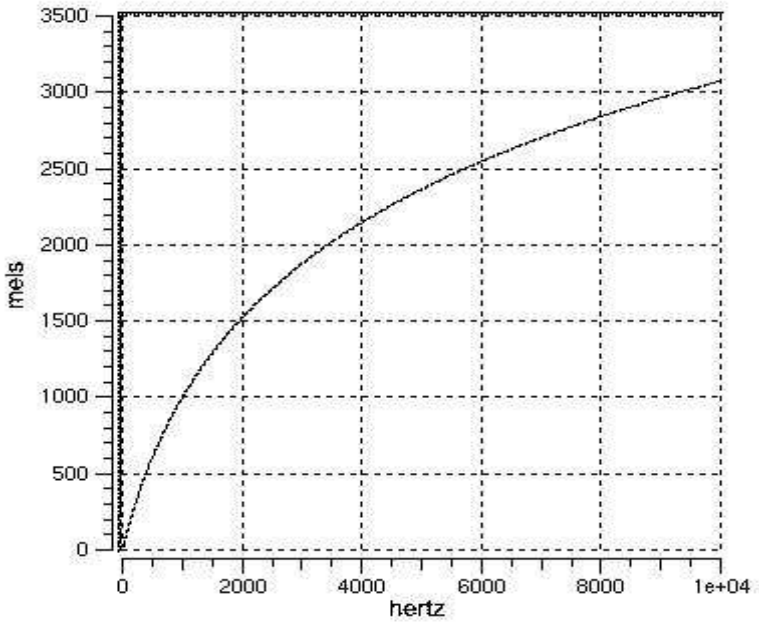
Un método mas eficiente para sacar características y que es el mas usado actualmente en reconocedores comerciales son los Coeficientes Cepstrales en Escala Mel, este método es un método robusto que hace uso de la Trasformada de Fourier para obtener las frecuencias de la señal. El objetivo es desarrollar un conjunto de valores de características basados en criterios perceptuales, diversos experimentos muestran que la percepción de los tonos en los humanos no está dada una escala lineal, esto hace que se trate de aproximar el comportamiento del sistema auditivo humano. Los Coeficientes Cepstrales en Frecuencia Mel (MFCC) son una representación definida como el cepstrum de una señal ventaneada en el tiempo que ha sido derivada de la aplicación de una Transformada Rápida de Fourier, pero en una escala en frecuencias no lineal, las cuales aproximan el comportamiento del sistema auditivo humano

Davis y Mermelstein en 1980 mostraron que los MFCC son beneficiosos para el Reconocimiento Automático del Habla.

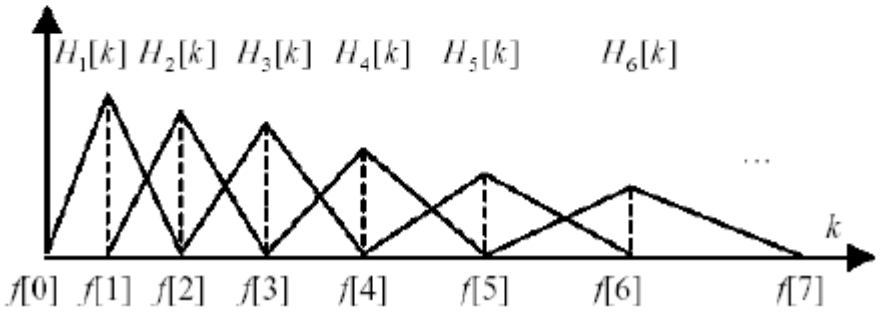
Dada una Transformada Discreta de Fourier de una señal de entrada:

$$X[k] = \sum_{n=0}^{N-1} x[n] e^{\frac{-j2\pi nk}{N}}, \quad k = 0, 1, ..., N-1$$

Se define un banco de filtros M, con (m = 1, 2, ...,M) donde el filtro m es un filtro triangular dado por:



$$H_m[k] = \begin{cases} 0 & \text{si } k < f(m-1) \\ \frac{k-f(m-1)}{f(m)-f(m-1)} & f(m-1) \le k \le f(m) \\ \frac{k-f(m-1)}{f(m+1)-f(m)} & f(m) \le k \le f(m+1) \\ 0 & k > f(m+1) \end{cases}$$



Estos filtros calculan el promedio del espectro alrededor de cada frecuencia central. Definimos f_l como la frecuencia mas alta y f_h como la frecuencia mas baja del banco de filtros en Hz, *Fs* es la frecuencia de Muestreo en Hz, *M* el numero de filtros y *N* el tamaño de la Transformada Rápida de Fourier. Los puntos límite *f(m)* son uniformemente espaciados en la escala Mel:

$$f(m) = \frac{N}{F_s}\beta^{-1}(\beta(f_1) + m\frac{\beta(f_h) - \beta(f_1)}{M+1})$$

donde la escala Mel β esta dada por:

$$\beta(f) = 1125\ln(1 + \frac{f}{700})$$

y su inversa β^{-1} esta dada por:

$$\beta^{-1}[b] = 700(\exp(\frac{b}{1125}) - 1)$$

Entonces finalmente se calcula el logaritmo de la energía de cada filtro:

$$S(m) = \ln(\sum_{k=0}^{N-1} |X_a(k)|^2 H_m(k)), \quad 0 < m < M$$

El Cepstrum en Frecuencia Mel es la Transformada Discreta del Coseno de las salidas de los *M* filtros:

$$c(m) = \sum_{m=0}^{M-1} S(m) \cos(\pi n(\frac{m - \frac{1}{2}}{M}))$$

donde *M* varía para diferentes implementaciones de 24 a 40, para el Reconocimiento Automático del Habla generalmente son usados los primeros 13 coeficientes. Este algoritmo es ampliamente usado para obtener el vector de características en sistemas de Reconocimiento Automático del Habla.

1.6.1 Deltas y dobles deltas

Los cambios temporales en el espectro, tienen una importancia significativa en la percepción humana, una manera de capturar estos cambios es utilizando los *coeficientes delta* cuya finalidad es medir el cambio de los coeficientes en el tiempo.

Las características a usar en un reconocedor usando MFCC con Fs de 16 Khz, es la siguientes:

Por frame analizado (ventaneamiento) se tendrá un vector X_k

$$\mathbf{x}_k = \begin{pmatrix} \mathbf{c}_k \\ \triangle \mathbf{c}_k \\ \triangle\triangle \mathbf{c}_k \end{pmatrix}$$

donde:

- $\boldsymbol{c_k}$ 13 coeficientes MFCC (frame de 25 ms, separacion entre frames de 10 ms)
- 13 coeficientes delta MFCC (cada 40 ms)

$$\triangle \mathbf{c}_k = \mathbf{c}_{k+2} - \mathbf{c}_{k-2}$$

- 13 coeficientes delta delta

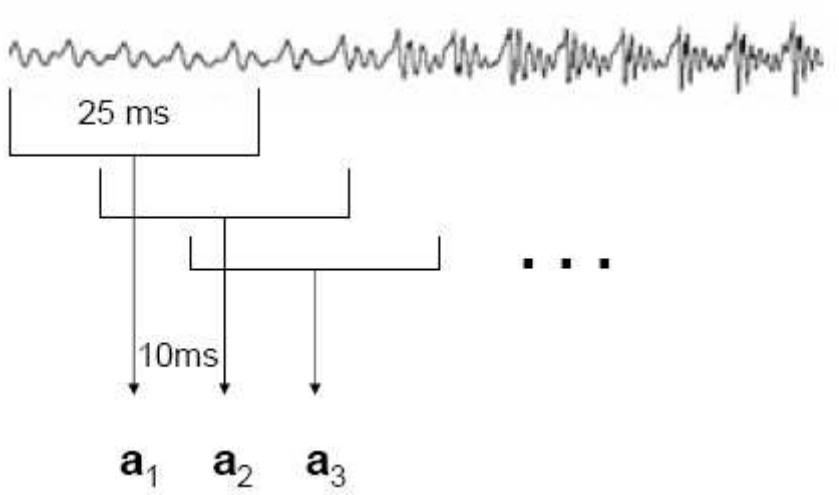
$$\triangle\triangle \mathbf{c}_k = \triangle \mathbf{c}_{k+1} - \triangle \mathbf{c}_{k-1}$$

otra manera de calcular estas derivadas es la siguiente:
para los 13 deltas

$$y'[t] = (y[t-2] - 8y[t-1] + 8y[t+1] - y[t+2])/12$$

para los 13 delta delta

$$y''[t] = y[t-1] - 2y[t] + y[t+1]$$



1.7 Prediccion Lineal Perceptual (PLP)

Mezcla las mejores carácterísticas de los MFCC y de los LPC, generalmente utiliza, recursión de Durbin para calcular los coeficientes LPC, luego trasforma estos coeficientes al LPC-cepstrum, pero los coeficientes de autocorrelacion no son computados en el dominio del tiempo.

1.8 Carácterísticas con longitud del tracto vocal normalizadas

Tratan de eliminar utilizando diversos algoritmos la variabilidad existente entre diversos hablantes, para solo trabajar con la variación propia de cada fonema.

Referencias Bibliográficas

[1] Oppenheim, A. V., Schafer, R. W., and Buck, J. R. 1999 *Discrete-Time Signal Processing (2nd Ed.)*. Prentice-Hall, Inc.

[2] H. Hermansky, (1990) “Perceptual Linear Predictive Analysis of Speech”, J. Acoust. Soc. Am., 87(4) pp. 1738-1752

[3] Huang, X., Acero, A., and Hon, H. 2001 *Spoken Language Processing: a Guide to Theory, Algorithm, and System Development*. 1st. Prentice Hall PTR.

[4] by Jurafsky and Martin SPEECH and LANGUAGE PROCESSING An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition Second Edition