RECONOCIMIENTO AUTOMÁTICO DEL HABLA UTILIZANDO WAVELETS

Al inicio del presente ciclo se planteó llegar hasta la construcción de un prototipo para probar algunos resultados de la investigación realizada hasta el momento.

Para el presente trabajo se han tomado muestras de personas comprendidas entre los 18 y 65 años de edad, debido a que los niños por ejemplo presentan un timbre de voz muy fino y por ahora aun no hemos estudiado esos casos, dichas muestras comprenden las vocales y en algunos casos los números del 1 al 10.

A continuación explicaremos las fases que comprende el desarrollo del trabajo.

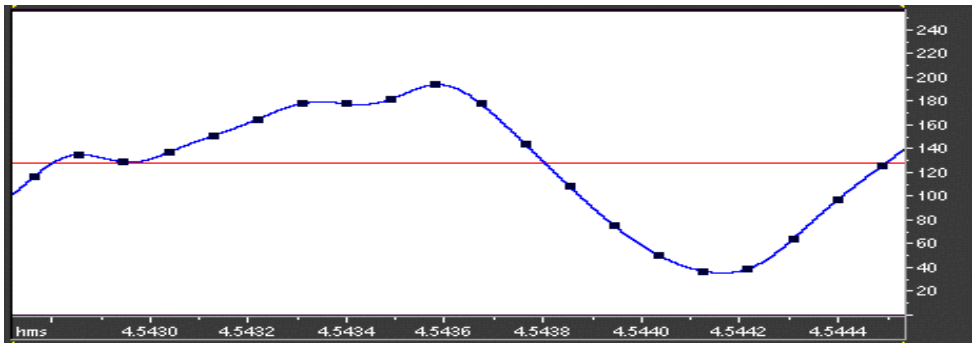
1. PROCESAMIENTO DE LA SEÑAL:

1.1. Captura de la señal:

La señal de voz básicamente está constituida por ondas de presión producidas por el aparato fonador humano. La manera obvia de capturar este tipo de señal se realiza mediante un micrófono, el cual se encargará de convertir la onda de presión sonora en una señal eléctrica.

A partir de la señal analógica obtenida se hace necesario convertir la señal a formato digital para poder procesarla en la computadora.

Por ejemplo tenemos una señal la cual queremos digitalizarla:



Como podemos observar, tomamos un valor en voltaje equivalente a la onda real el que será cuantizado, es decir transformado a digito así su correspondiente valor binario es el que será almacenado en la memoria.

Todo este proceso se resume a::

- **Muestreo:**

  Para realizar esto debemos saber que la señal vocal tiene componentes frecuenciales que pueden llegar a los 10 khz., sin embargo la mayor parte de los sonidos vocales tiene energía espectral significativa hasta los 5 khz.

  El muestreo de una señal consiste en el paso de la señal de la forma analógica al ámbito discreto, es decir viene a ser el proceso de captura de puntos (muestras) que sean necesarios para poder representar la señal en una unidades de tiempo (segundo), para esto debemos de tener muy en cuenta el siguiente teorema de muestreo:

  $Fs = 2 * Fa$, $Ts = 1/Fs$, $x(t) \longrightarrow x(nTs)$

  Fs = Frecuencia de sampleo Fa = Más alta frecuencia de la señal
  Ts=Periodo de sampleo n = entero mayor igual que cero

  donde las muestras tomadas corresponderán a los correspondientes valores x(nT) en la señal continua.

- **Cuantificación:**

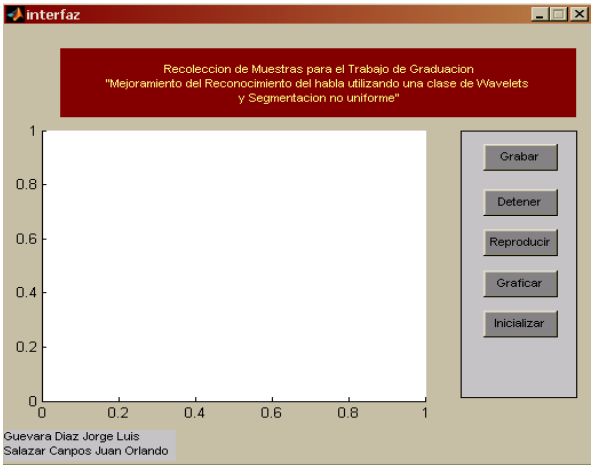
  Otra consideración que se debe tener en cuenta es la cuantificación de la señal, la cual involucra la conversión de la amplitud de los valores muestreados a forma digital usando un número determinado de bits. El número de bits usados afectará la calidad de la voz muestreada y determinará la cantidad de información a almacenar.

  La señal de voz exhibe un rango dinámico de unos 50 a 60 dB. por lo que resultaría suficiente una cuantificación de 8 a 9 bits para una buena calidad de voz.

El proceso de captura de la señal lo hemos efectuado con:

- ✓ Un micrófono M 750 H (V) Dynamic Stereo HeadPhone Microphone Combo, con control de volumen, un rango de frecuencias de 20 Hz a 20 KHz, una impedancia de 32 Ohms, una sensibilidad de –58 db.
- ✓ Una tarjeta de sonido SoundMax integrada a una placa Intel 850EMV2.
- ✓ La frecuencia fue de 8000 Hz.
- ✓ El tamaño para cada amplitud fue de 8 bits, con lo que podemos obtener 256 amplitudes diferentes.

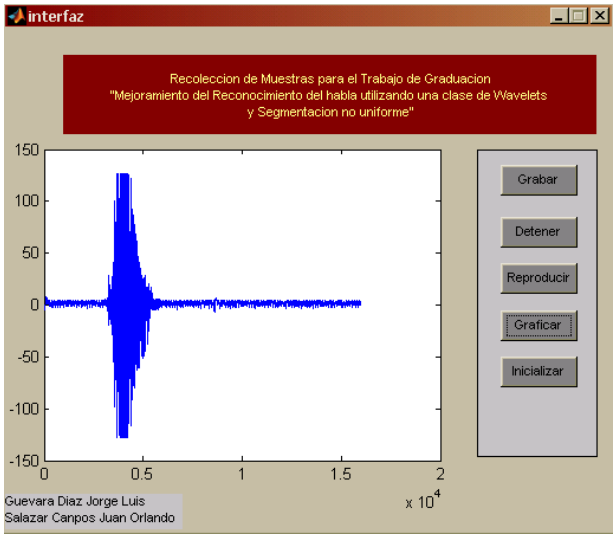
Para este fin diseñamos e implementamos la siguiente interfaz, la cual nos devolverá un archivo (vector) con los datos (números enteros) de la señal digitalizada:



.

Por ejemplo algunas muestras de vocales graficadas son:

Vocal: "a".



1.2. Selección y Eliminación de Segmentos Inservibles:

Una vez que hemos obtenido un vector con los datos de la señal digitalizada tenemos que eliminar las secciones que no contienen información valida para nuestros fines, tales como los valores del inicio y del final captados ya sea por la demora en pronunciar una vocal o por la demora en detener la grabación.

Para esto obtenemos un umbral al cual le sumamos cierto valor, con el cual vamos a comparar las muestras, si existen valores que estén por debajo de éste se eliminarán.

El umbral lo obtenemos de la siguiente manera:

$$\text{valor} = \frac{1}{radio} \sum_{i}^{radio+i} |x|$$

radio = Número de muestras que se están evaluando.

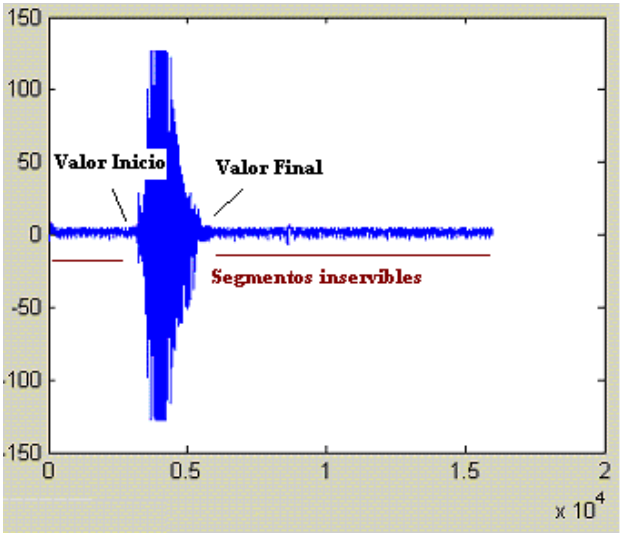
umbral = (valorI + valorF)/2 + n

Donde:

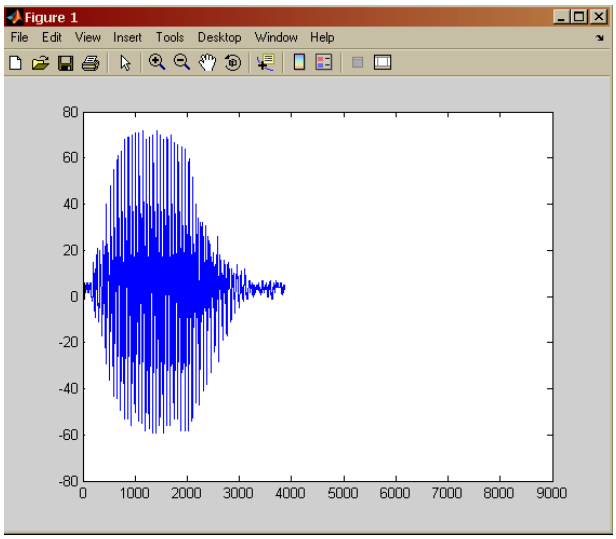
ValorI = valor de las x primeras muestras.
ValorF = valor de las x últimas muestras.
n = un valor fijo.

En el trabajo obtuvimos el umbral de aproximadamente 14, del promedio de los valores absolutos de las 10 primeras muestras y de las 10 últimas, pues con seguridad éstos valores representan momentos de silencio los cuales no nos sirven, el valor que le sumamos (n) fue de 5, de esta forma recorremos el vector con un radio de 5 elementos, se empieza a desde el principio si su promedio es menor que el umbral se descartan caso contrario se fija el ***inicio*** de la señal que deseamos, para encontrar el final de la misma manera empezamos a buscar los promedios de los valores absolutos de las muestras pero empezando por el final hasta encontrar el valor al que llamaremos ***final.***

Una vez que tenemos los valores de inicio y final recortamos el vector de muestras.



Por ejemplo la vocal i quedaría de la siguiente manera:



1.3. Normalización:

Luego de tener solamente la información necesaria, tenemos que normalizar para poder trabajar uniformemente, esta fase trata de llevar a un determinado rango y sus equivalentes todas los valores de las muestras.

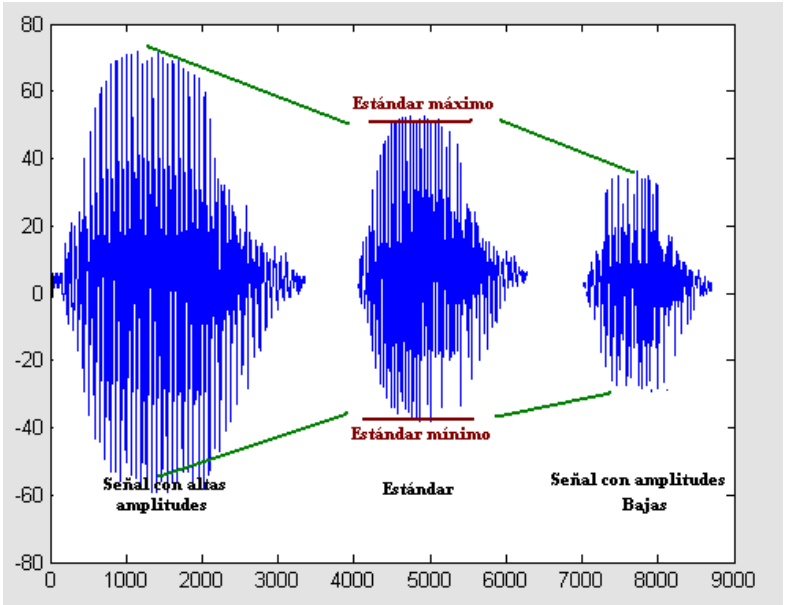
Así:

$$[a,b] \longrightarrow f \longrightarrow [c,d] \longrightarrow T \longrightarrow [m,n]$$

$T(y)=Ay+B$
$T(c)=m$
$T(d)=n$

$$A = (n-m)(d-c) \quad B=m-(n-m)/(d-c)$$
$$T(y) = (n-m)y+md-nc/(d-c)$$

Luego de haber normalizado tenemos los valores de las señales entre un determinado rango para todos.

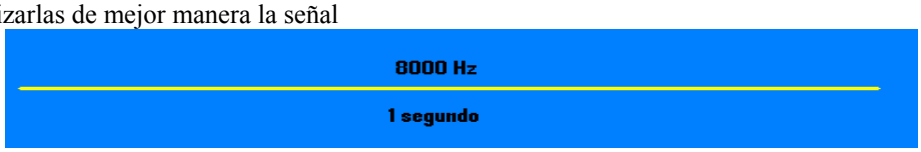


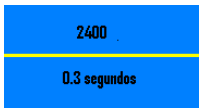
1.4. Segmentación Uniforme:

La segmentación consiste en dividir un vector en varias partes, pueden ser iguales (segmentación uniforme) o desiguales (segmentación no uniforme).

Mediante una segmentación nosotros podremos obtener una mejor caracterización de la señal, por lo que hemos decidido segmentar el vector que nos queda en partes equivalentes a 0.3 segundos, ya que es la unidad mínima que contiene información válida, teniendo cuidado que estos segmentos deben de contener un tamaño de una potencia de 2.

Si tenemos que en 1 segundo capturamos 8000 muestras en 0.3 segundos tendremos 2400 muestras, pero la potencia de 2 más próxima es $2^{11} = 2048$. En el caso de las vocales nos puede salir un solo segmento pero en palabras grandes tendremos más, pudiendo así caracterizarlas de mejor manera la señal

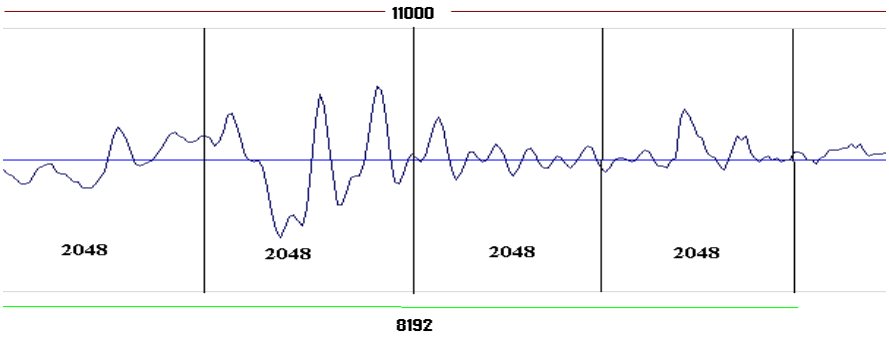




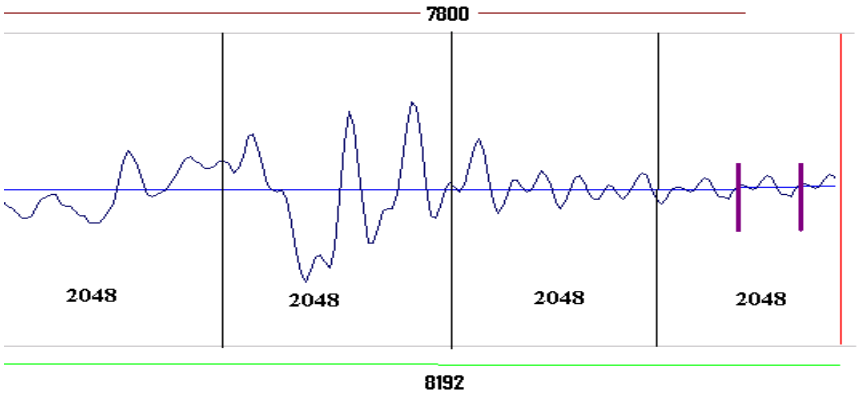
Obteniendo un vector con el siguiente tamaño:

Tamaño del vector es: 2^11 * Número de Segmentos..

Acá se presentan dos casos cuando el tamaño del vector es superior al que deseamos obtener, calculamos el tamaño que deseamos y descartamos los últimos valores.

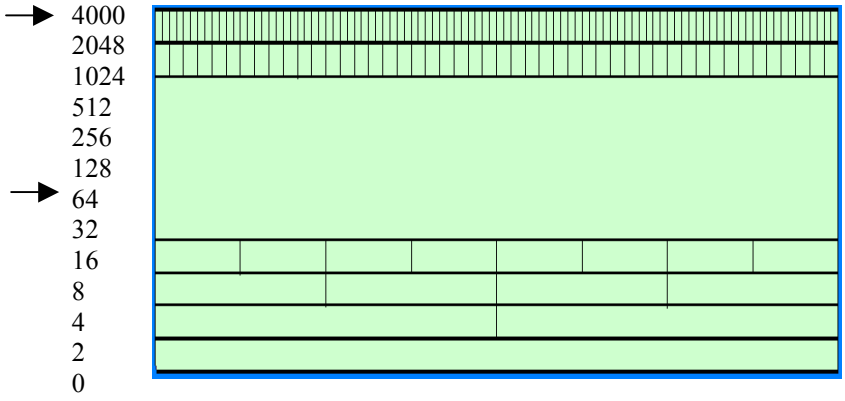


El otro caso es cuando la cantidad de valores del vector no superar el tamaño deseado, en este caso cogemos los últimos 5 valores y los duplicamos hasta completar el tamaño desead.

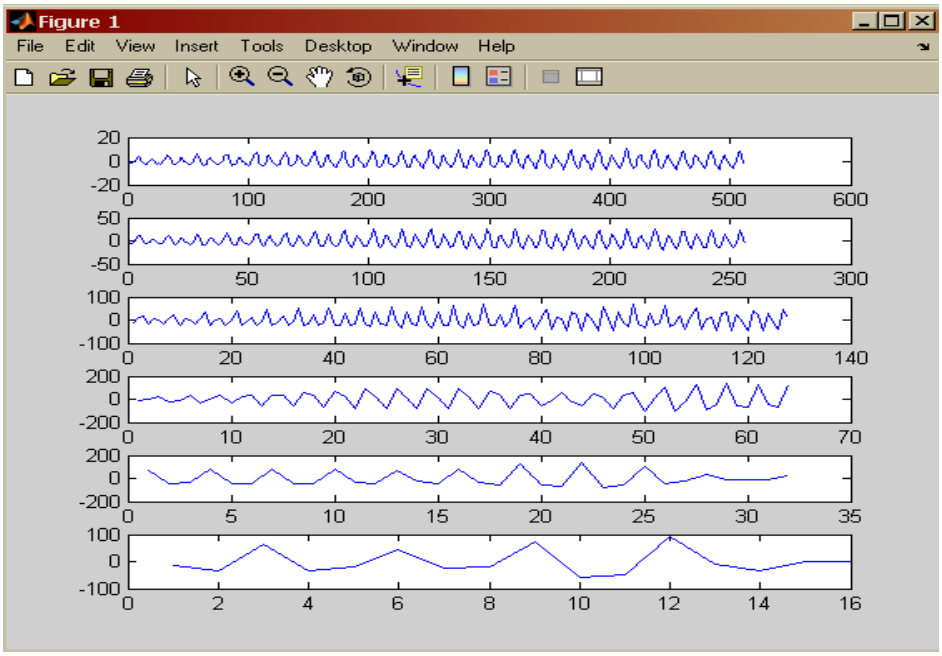


1.5. Aplicación de la Transformada Rápida de Wavelets Haar:

Una vez que tenemos los segmentos con los tamaños adecuado aplicamos la Transformada Rápida de Wavelets Haar a cada uno con un nivel de descomposición 6, ya que entre estas frecuencias se encuentra la mayor parte de información válida.



Por ejemplo para las vocales solamente tendremos un segmento, lo que no pasa con los números, observemos la imagen, pertenece a la vocal i descompuesta en 6 niveles:



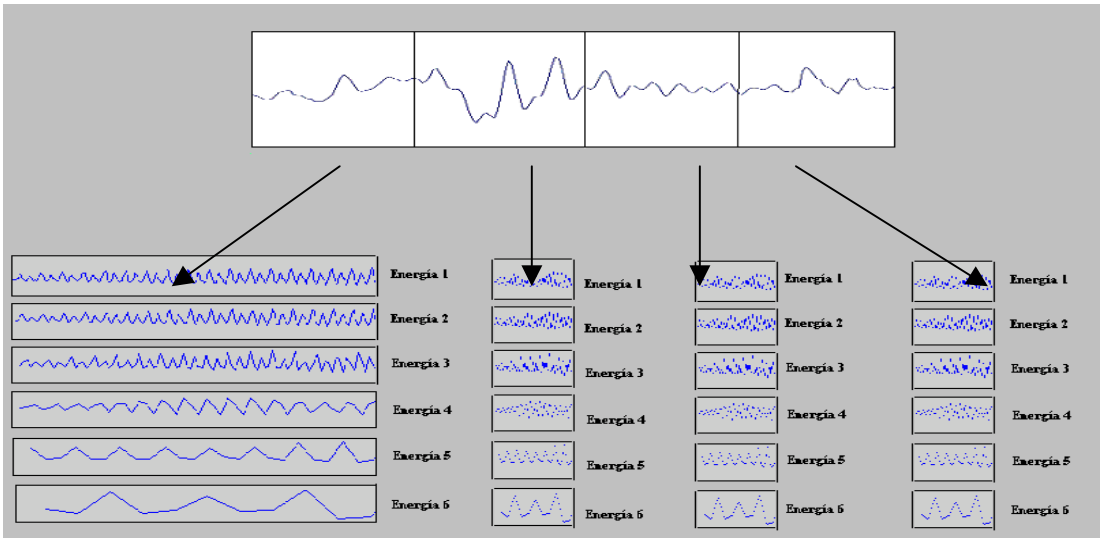
1.6. Formación del Patrón de Características:

En este punto debemos de tener mucho cuidado ya que una mala técnica para la extracción de características podría hacer que funcione de manera defectuosa la red neuronal que se implementará.

Hemos decidido que el patrón de características este formado por la energía de cada nivel, por lo que tendremos 6 energías por cada segmento. Luego para formar el patrón general de la señal concatenaremos estas energías (si hubiesen varias), de izquierda a derecha.

$$\text{Energía} = \frac{1}{k}\sum_{1}^{k}(x^2)$$

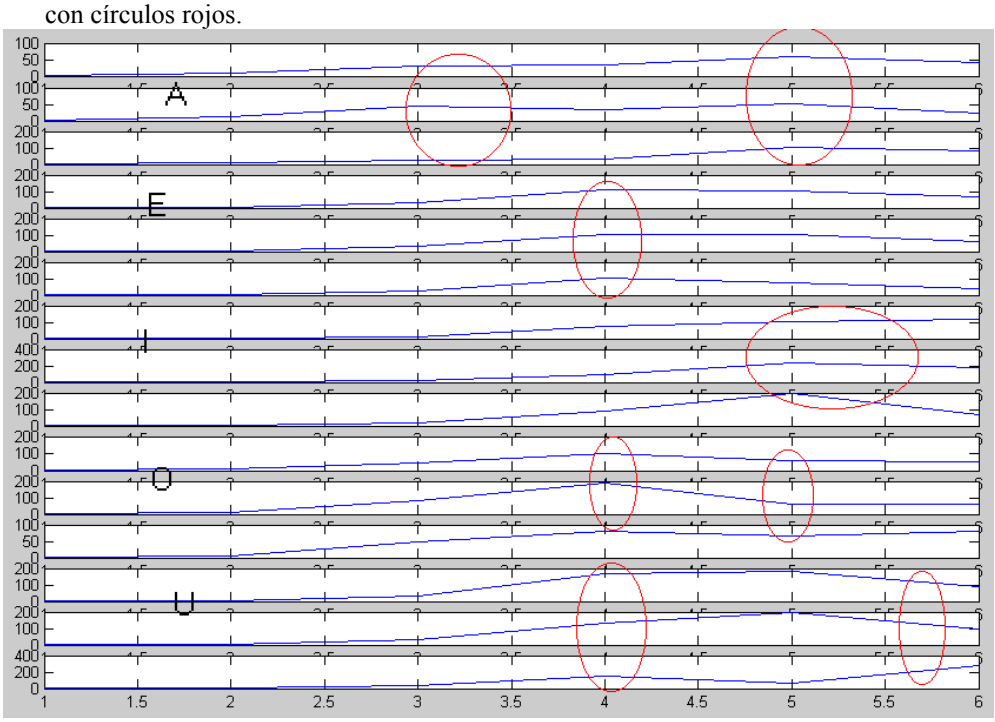
k = Cantidad de elementos del nivel de descomposición



Patrón de características:

Energía 1	Energía 2	Energía 3	...	...	Energía 6	...	...	...	...	Energía 1	Energía 2	Energía 3	...	...	Energía 6	...	...	...			

Una vez que se formado el patrón de características (vector), este queda listo para ser ingresado a la red neuronal, para ello nosotros hemos agrupado en el caso de las vocales, varias muestras para cada vocal, la siguiente figura nos muestra de mejor manera esta agrupación y se puede notar las características propias de cada vocal que han sido resaltadas con círculos rojos.



2. FASE DE RECONOCMIENTO:

Para esta fase se tomaron 5 muestras por persona y por vocal, es decir 5 muestras de la vocal a por una persona, 5 de la e y así hasta completar las vocales, un proceso similar se hizo para algunos números.

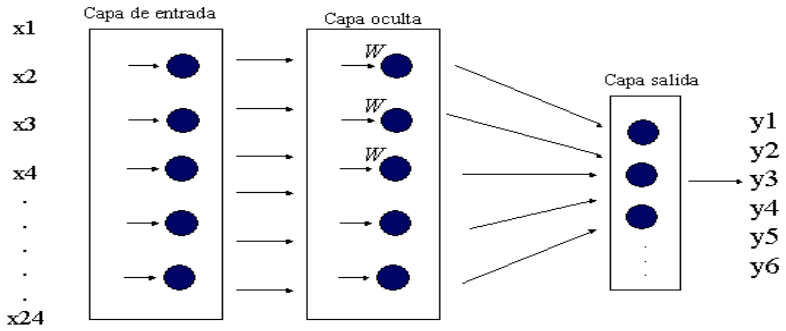
2.1. Fase de entrenamiento.

Para el presente trabajo hemos optado por una Red Neuronal BackPropagation por sus buenos resultados en las diferentes aplicaciones que se le ha dado.

Nuestro modelo consta de una capa de entrada que depende del patrón de entrada y de la cantidad de segmentos que se tengan, una capa oculta con 15 neuronas y una capa de salida con 6 neuronas. Usa la función de activación del tipo sigmoidal

$$f(x) = \left(1 + e^{-x}\right) - 1$$

Las muestras tomadas tienen que pasar por todas las fases antes descritas. No hemos eliminado el ruido por que la red debe reconocer una vocal en ambientes ruidosos.



Algoritmo de backpropagation:

1. Estructura de la red
2. Escoger funciones de activación
3. Dar vectores de entrenamiento
4. Inicializar los pesos
5. Calcular la salida de la red "O"
6. Calcular los deltas para la capa salida y oculta
7. Ajustar pesos capa salida y oculta
8. Propagar el error hacia atrás y ajustar los pesos de las diferentes capas
9. Repetir los pasos 3-8 con el siguiente vector de entrenamiento hasta que el error sea bien pequeño

2.2. Fase de Emparejamiento (Aplicación):

Esta es la fase final de aplicación una vez que se haya entrenado a ala red, estará en condiciones de reconocer las vocales no importa quien lo diga.

El proceso empieza en el momento en que la persona pronuncia la vocal luego se procesa la señal y como resultado obtendremos un mensaje indicándonos si la señal pertenece o no a una vocal .

CONCLUSIONES:

- El Hardware empleado en la captura de la señal influye en los resultados de éste proceso.
- Los valores capturamos dependen del muestreo y la cuantificaron.
- En el ambiente aunque no se pronuncie habla alguna siempre hay señales que son captadas.
- Algunos métodos de Normalización pueden variar los datos de la señal muestreada.
- La segmentación nos facilita la formación del patrón de características de una señal.
- Los wavelets nos brindan una mejor información de una señal analógica que otro métodos convencionales no pueden.´
- Los wavelets nos ayudan en la extracción de características para formar un patrón único.
- Una red Neuronal BackPropagation puede trabajar adecuadamente pero el costo de aprendizaje es alto.

SUGERENCIAS:

A pesar de ser un tema de investigación en el cual incursionamos en temas relativamente nuevos podemos sugerir lo siguiente:

- Mejor los métodos de Eliminación de Segmentos Inservibles y de Transformación de la señal a un Rango Determinado.
- Implementar una segmentación no uniforme que es uno de los fines del trabajo.
- Probar si se puede obtener un mejor patrón de características usando los 9 niveles de descomposición y no 6 como se lo está haciendo
- En la selección de patrones investigar si existe otro método mejor que el cálculo de las energías por cada nivel.
- Implementar una red neuronal especial para el reconocimiento del habla.
- Probar con otro tipo de wavelets, ya que los wavelets de Haar son muy básicos.

REFERENCIAS

1. Andrew K. Chan y Jaideva C Goswanmi, “Fundamental of Wavelets. Theory, Algorithms and Aplications”, Texas A&M Univertisy.
2. Bellman, R., Kalaba, R. "Dynamic Programming and Modern Control Theory" Academic Press Inc., 1965.
3. Bernal Bermúdez Jesús, Bobadilla Sancho Jesús, Gómez Vilda Pedro., “Reconocimiento de Voz y fonética acústica”. Printed in México.
4. Fabián Acquaticci., Sergio Gwirc, Diego Brengi, “Aplicación de Redes Neuronales para el Control de Calidad de Productos Lácteos Uht”, Instituto Nacional de Tecnología Industrial, Centro de Investigación en Tecnología Electrónica e Informática, Argentina, Bueno Aires.
5. FORNEY G. D. The Viterbi Algorithm, “Proceedings of the IEEE, Vol. 61, Mar. 1973, pp 268.
6. L. G. Weiss, "Wavelets and wideband correlation processing”, IEEE Signal Process. Magazine, January 1994.
7. Linde Y., A. Buzo y R. M. Gray, “An Algorithm for Vector Quantizer Design”, IEEE Transactions on Communications, Vol. COM-28, pp 84.
8. M. J. Shensa, "The discrete wavelet transform: Wedding the a trous and Mallat algorithms", IEEE Trans. Signal Process, October 1992.
9. Myers y L. R. Rabiner, “Connected Digit Recognition Using a Level-Building DTW Algorithm”, C. S., IEEE Transactions on Acoustics, Speech and Signal Processing, Vol. ASSP-29, n° 3, pp 351.
10. Myers, C. S., Rabiner, L.R. "A Level Building Dynamic Time Warping Algorithm for Connected Word Recognition" IEEE Trans. on Acoustic, Speech and Signal Processing, vol. ASSP-29, num. 2, pp.284-297, April 1981.
11. Ney H., “Architecture and Search Strategies for Large-Vocabulary Continuous-Speech Recognition”, Proc. of Nato Advanced Study Institute on Speech Recognition and Understanding, Bubion, Spain, 1993, pp.59-84.
12. Ney, H. "Stochastic Grammars and Pattern Recognition", Proc. of Nato ASI, 1990, pp. 319-344.

13. Niemann H., M. Lang & G. Sagerer, "Recent advances In speech understanding and dialog systems", 5pringer Verlag, 1988.
14. O. Rioul and P. Duhamel, "Fast algorithms for discrete and continuous wavelet transforms", IEEE Trans. Inform. Theory, March 1992.
15. Pablo Faundez, Alvaro Fuentes, "Procesamiento Digital de Señales Acústicas utilizando Wavelets". Instituto de Matemáticas UACH.
16. Priegue, R. y García Martínez, R., "Reconocimiento de la voz mediante una red neuronal de kohonen", Centro de Ingeniería del Sw e Ingeniería del conocimiento, Argentina, Bueno Aires.
17. Poza M. J., J. F. Mateos y J. A. Siles, "Design of an Isolated Word ASR for the Spanish Telephone Network", Proceedings de International Conference on Signal Processing, Beiging, 1990.
18. Poza M. J., J. F. Mateos y J. A. Siles, "Audiotext with Speech Recognition and Text to Speech Conversion for the Spanish Telephone Network", Proceedings de Worldwide Voice Systems'90, London.
19. Rabiner L. R. y B. H. Juang, "An introduction to Hidden Markov Models", IEEE ASSP MAGAZINE, January 1986.
20. Rabiner L. R., "A Tutorial on, LIMM and Selected Applications M Speech Recognition", Proceedings of the IEEE, Vol. 77, n° 2, pp 257.
21. Richard P. Loppmann, "Neural Nets for Computing", ICASSP 1989. 7. Trends In speech recognition, W. A. LEA, Prentice Hall, 1980.
22. Sakoe, H., "Two-Level DP-Matching- A Dynamic Programming - Based Pattern Matching Algorithm for Connected Word Recognition", IEEE Trans. on Acoustic, Speech and Signal Processing, vol. ASSP-27, num.6, pp.588-595, December 1979.