ALGORITMO COPER PARA LA DETECCIÓN DE ACTIVIDAD DE VOZ.

Bach. Fernando Peralta Reyes
f7peralta@hotmail.com

Bach. Anibal Cotrina Atencio
acotrin@hotmail.com

Candidatos a Optar Titulo Profesional de la Facultad de Ingeniería Electrónica de la UNMSM, Lima, Perú

Resumen: La exactitud de un Sistema de Reconocimiento de Voz, entre otros factores, depende de la eficiencia en el instante de detectar el inicio y final de la pronunciación de la palabra, mas aún, cuando se encuentra en presencia de ruido de fondo considerable.
El presente trabajo tiene por objeto mostrar un nuevo algoritmo de detección de extremos que incorpora ventajas con relación a los algoritmos utilizados usualmente.

Abstract: The accuracy of the voice recognition system, between others factors, depend of the efficient way of detecting the points of beginning and ending of the word, specially when it is mixed with noise. The main goal of this work is shown a new detector limits algorithm, which has many advantages in front of current algorithms.

Palabras claves: Detección de extremos, energía, cruces por cero, algoritmo COPER.

I. INTRODUCCIÓN

En reconocimiento de voz, la detección de actividad de voz o detección automática de extremos consiste en determinar los instantes de inicio y final de una pronunciación con el fin de entregar al sistema de reconocimiento únicamente el segmento de señal de voz comprendida en dichos instantes.

Los algoritmos de detección de extremos usuales, se basan en el análisis de la evolución en el tiempo de dos propiedades de la señal de voz, estas son la energía y los cruces por cero [Bernal e.t. all, 2000] y funcionan razonablemente bien, cuando la relación señal a ruido es superior a 30 dB, pero fallan considerablemente cuando la voz se encuentra inmersa en un entorno ruidoso [Crespo e.t. all, 2000]. Por lo tanto, se propone un nuevo algoritmo denominado COPER (Cotrina-Peralta), el cual analiza la combinación de energía y cruces por cero para realizar la detección de extremos más robusta.

II. ALGORITMO COMÚNMENTE UTILIZADO

Para detectar el comienzo de la pronunciación de una palabra se exige que la energía o los cruces por cero superen ciertos umbrales durante un período de tiempo y para la detección del final de la pronunciación los niveles de energía y cruces por cero deben caer por debajo de éstos [Flores, 1993]. Los niveles de umbral se

obtienen experimentalmente analizando el contenido de energía y cruces por cero que poseen tanto las palabras pronunciadas, como el ruido de fondo.

La energía de una señal discreta se define como:

$$E = \sum_{m=0}^{N-1} x(m)^2 \tag{2.1}$$

donde: $x(m)$ es la amplitud de la señal y N es el número de muestras.

La fórmula de la densidad de Cruces por Cero es:

$$z = \sum_{m=0}^{N-1} |sign\,[x(m)] - sign\,[x(m-1)]| \tag{2.2}$$

donde: *sign* es la función *signo* y N es el número de muestras.

Básicamente, consiste en adquirir una trama de la señal de voz y luego analizar el contenido de energía y cruces por cero, el cual es un proceso que se repite permanentemente. Esto se ilustra en el diagrama mostrado en la figura 1.

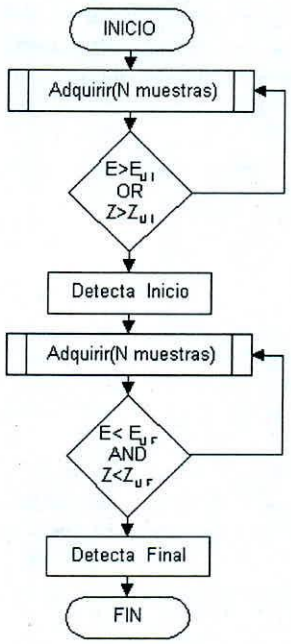

Figura 1. Diagrama de Flujo para la Detección de Extremos

donde:

E : Es la energía de la trama en análisis.
Z : Es la densidad de los Cruces por Cero de la trama en análisis.

E_{UI} : Umbral de Energía de Inicio de Pronunciación.
E_{UF} : Umbral de Energía de Final de Pronunciación.
Z_{UI} : Umbral de Cruces por Cero de Inicio de Pronunciación.
Z_{UF} : Umbral de Cruces por Cero de Final de Pronunciación.

A continuación en la figura 2 se muestra gráficamente el proceso de Detección de Extremos, la palabra usada como ejemplo, delimitada es "*Cinco*", con una relación señal a ruido (SN) de aproximadamente 33 dB. Como resultado se obtiene la palabra correctamente delimitada.

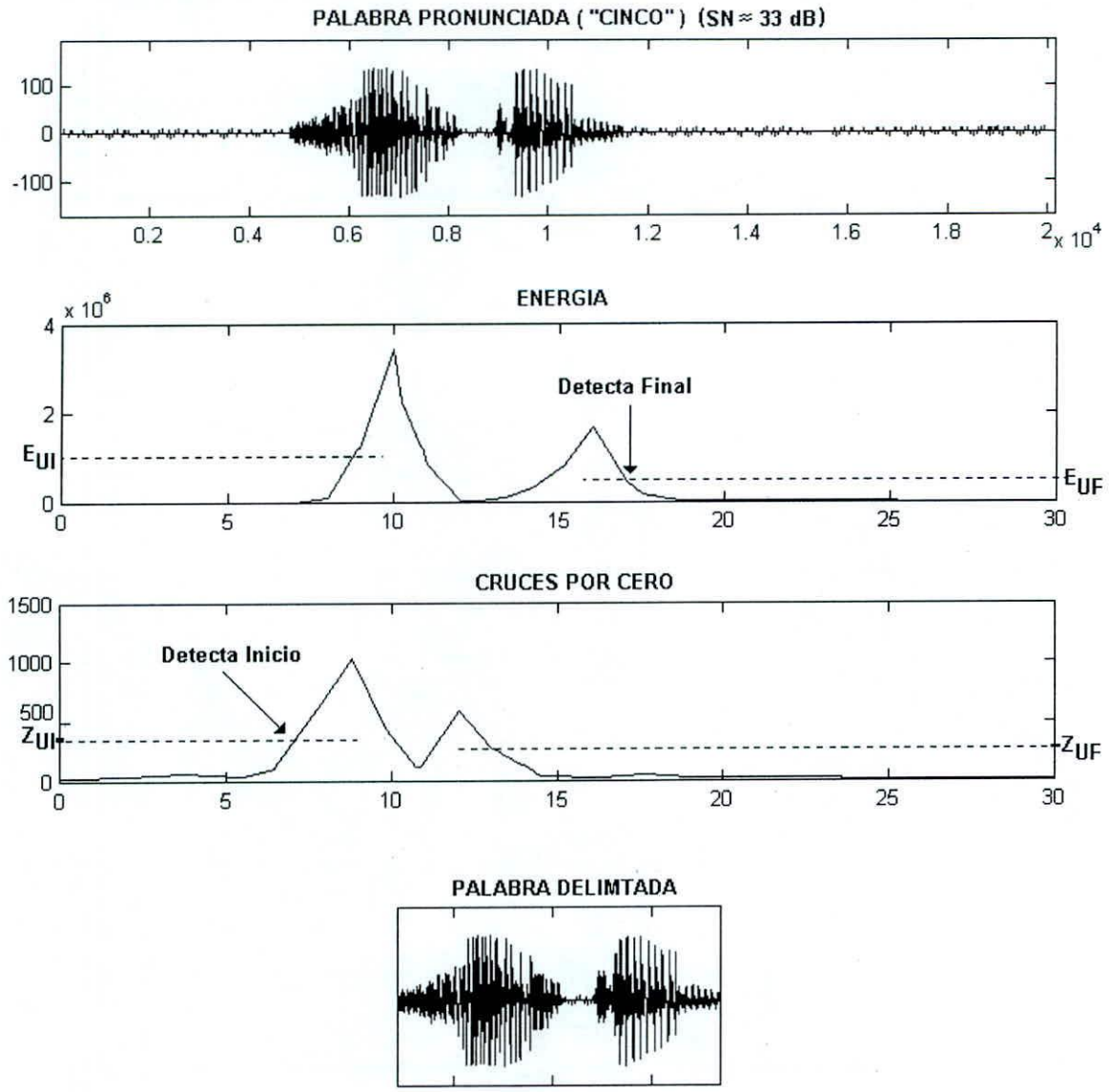

Figura 2. Proceso de Detección de Extremos.

Seguidamente en la figura 3, se muestra la delimitación de la misma palabra pero bajo un entorno ruidoso, donde la SN es de aproximadamente 20 dB. Se puede apreciar que la energía sigue proporcionando información

correcta sobre el inicio de los sonidos sonoros, mientras que los cruces por cero no se pueden distinguir entre los sonidos fricativos y el ruido de fondo, de esta forma se produce una delimitación incorrecta, tanto para el inicio y final de la pronunciación de la palabra, debido a que siempre se mantiene una alta densidad de cruces por cero.

Si se analiza la fórmula de la Densidad de Cruces por Cero, ecuación 2.2, se observa que ésta fórmula acumula los cambios de signo de la señal y si el ruido de fondo es de alta frecuencia tendrá una alta densidad de cruces por cero, por lo que se hace dificultoso distinguir entre el ruido de fondo y la palabra pronunciada. Por esta razón, se propone en el presente artículo, un nuevo algoritmo, que no sólo relaciona los cambios de signo, sino que a la vez proporciona información sobre los cambios de energía de la señal, el cual se ha denominado COPER, el cual se describe en la siguiente sección.

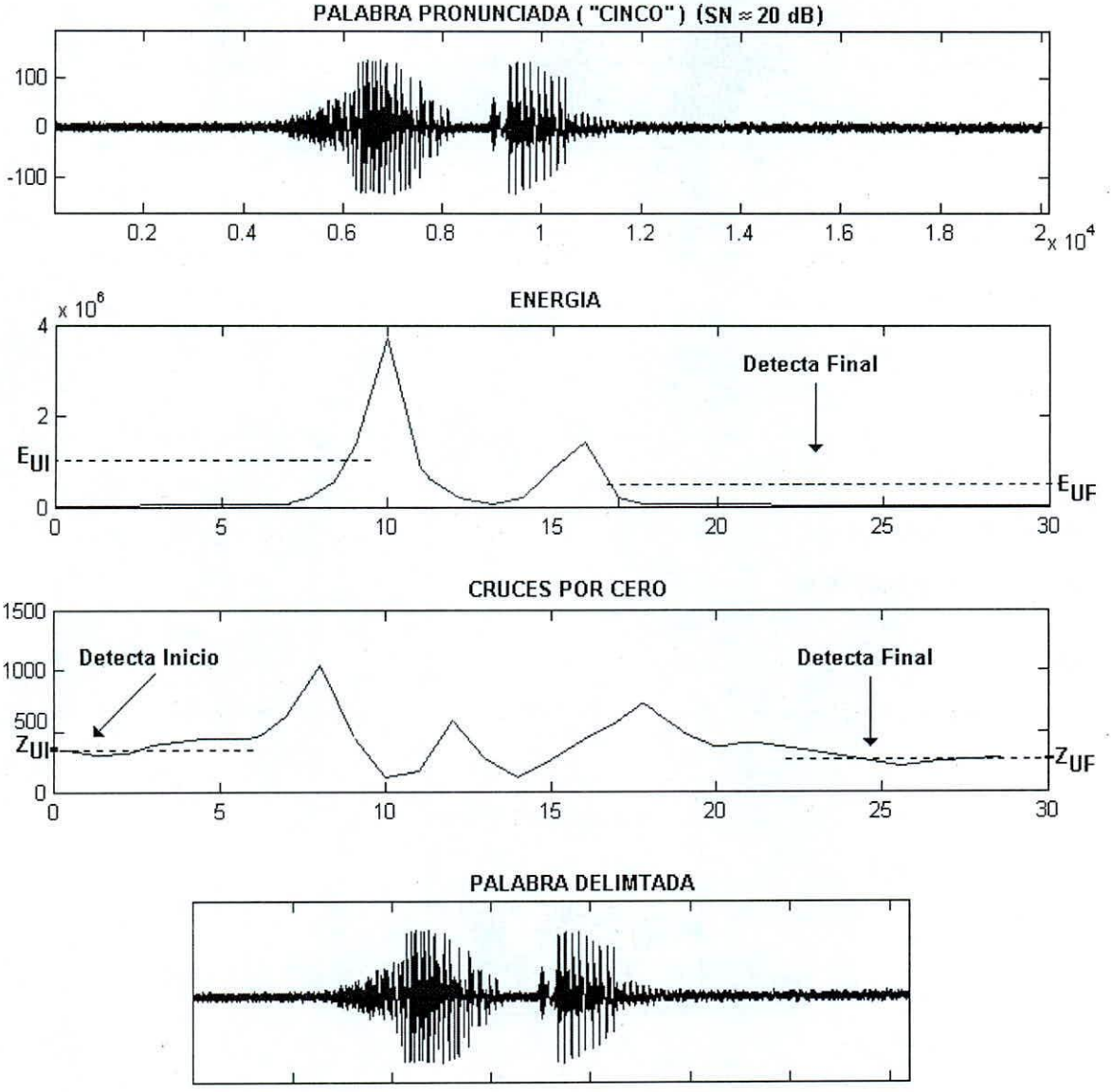

Figura 3. Proceso de Detección de Extremos.

III. ALGORITMO DE COPER

Este algoritmo que se propone, surge como parte de los resultados de nuestro trabajo de tesis sobre Reconocimiento de Voz Independiente del Locutor. La formulación es similar a la ecuación 2.2, pero a las funciones *signo*, se les ha multiplicado por la energía de la muestra analizada, de esta manera, la densidad acumulada no sólo dependerá del cambio de signo de las muestras sino también de la amplitud, por consiguiente, el ruido de fondo no logrará gran acumulación de densidad, debido a que posee una pequeña amplitud comparada con las palabras pronunciadas.

La fórmula matemática simplificada obtenida se muestra a continuación:

$$COPER = \sum_{m=0}^{N} \left| x[\mathrm{m}] \cdot |x[\mathrm{m}]| - \mathrm{x}[\mathrm{m}-1] \cdot |x[\mathrm{m}-1]| \right| \qquad (3.1)$$

Los resultados obtenidos, al realizar el análisis en el tiempo de la misma palabra del ejemplo, mostrada en la figura 2, basada en la fórmula de COPER, es mostrado en la figura 4, para una SN=33dB y en la figura 5, se muestra el mismo análisis para una SN=20dB.

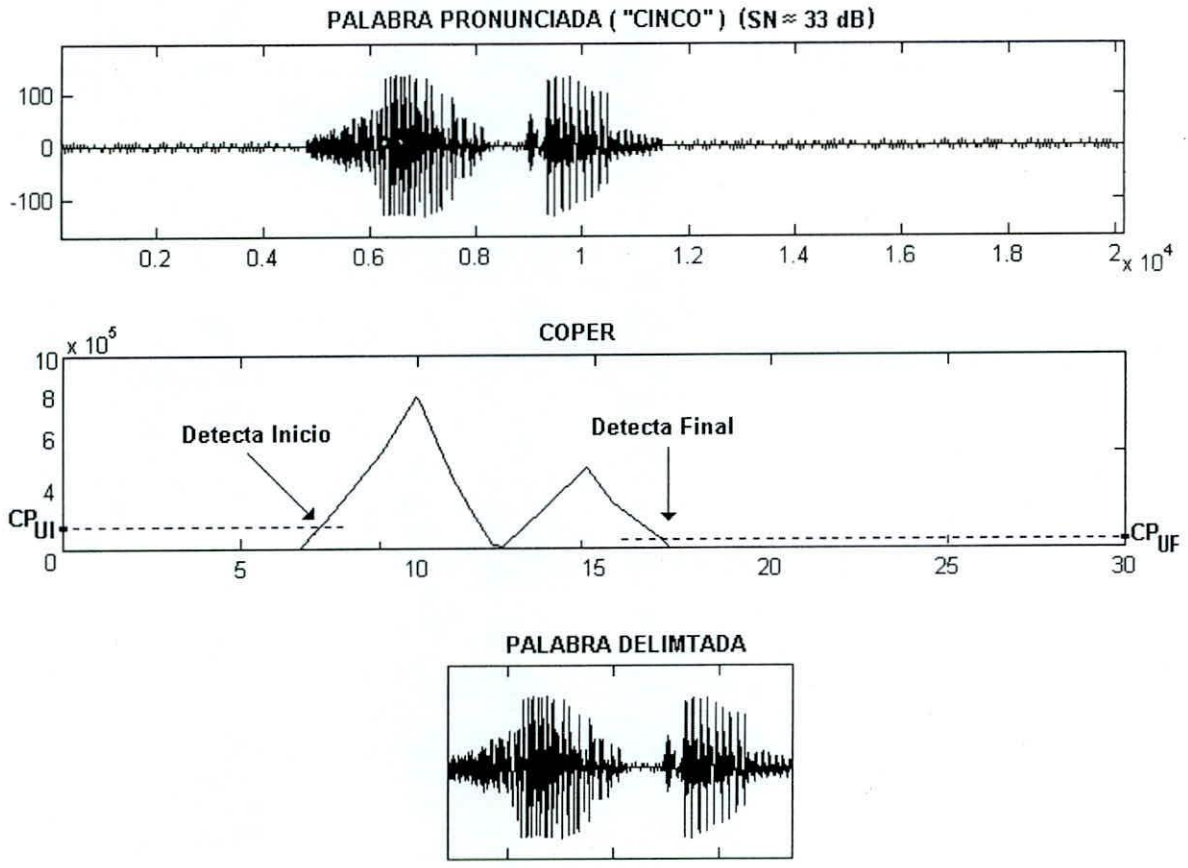

Figura 4. Proceso de Detección de Extremos.

donde

CP_{UI} : Es el umbral de inicio de pronunciación.
CP_{UF} : Es el umbral de final de pronunciación.

En las figuras 4 y 5, se aprecia que sólo existe una gran acumulación del parámetro COPER entre los instantes que dura la pronunciación y que el ruido no contiene información significativa. Por lo tanto en ambos casos se produce una correcta delimitación de la palabra pronunciada, en contraste con el resultado obtenido con un algoritmo comúnmente utilizado. Observando con detenimiento la figura 4 y 5 se puede notar como el parámetro COPER va siguiendo con detalle los cambios temporales que se llevan a cabo durante una pronunciación, demostrando así que el algoritmo COPER proporciona la suficiente información para ser utilizado como único parámetro en la Detección de Extremos.

Experimentalmente se ha determinado que al algoritmo COPER proporciona una respuesta aceptable, hasta figuras de ruido de 15 dB. En la figura 6, se muestra la delimitación de 10 palabras (los 10 dígitos del castellano) utilizando el algoritmo COPER, con una figura de ruido de aproximadamente 15 dB.

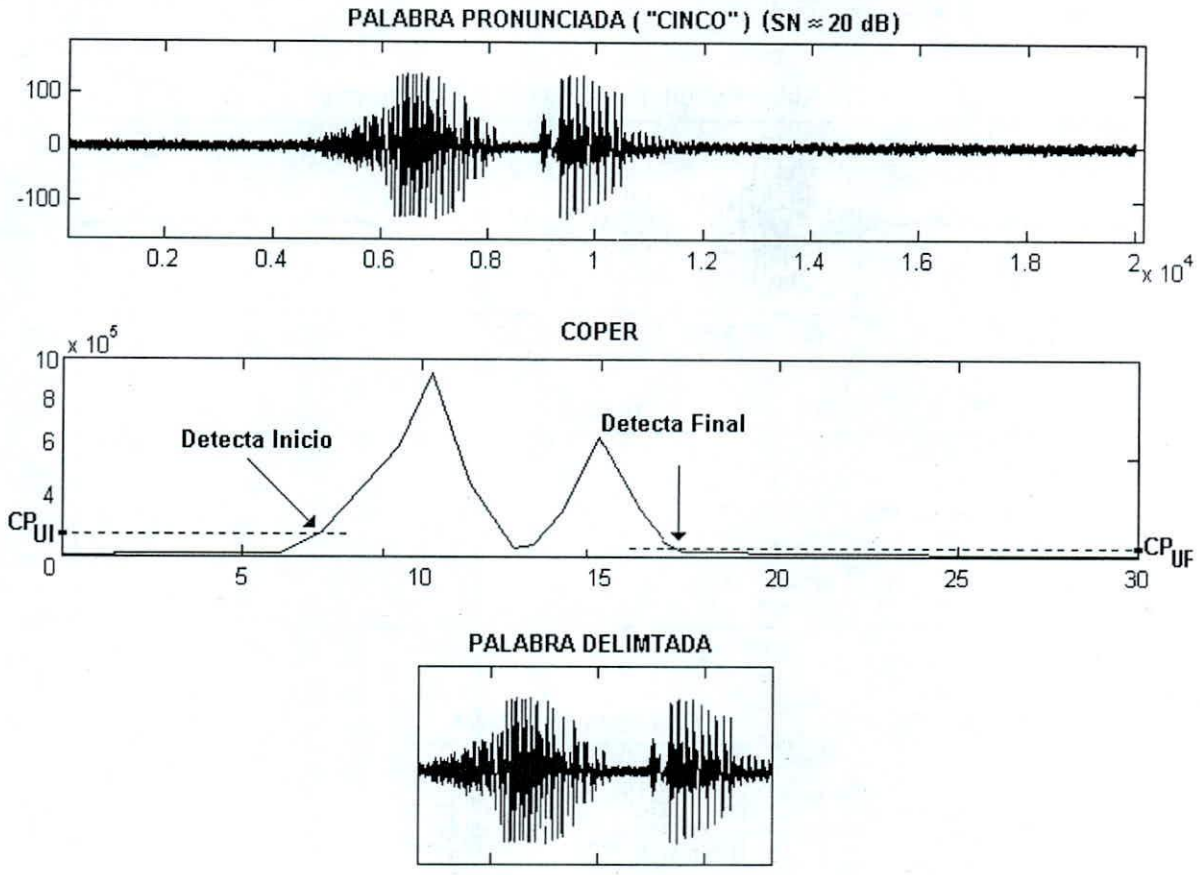

Figura 5. Proceso de Detección de Extremos.

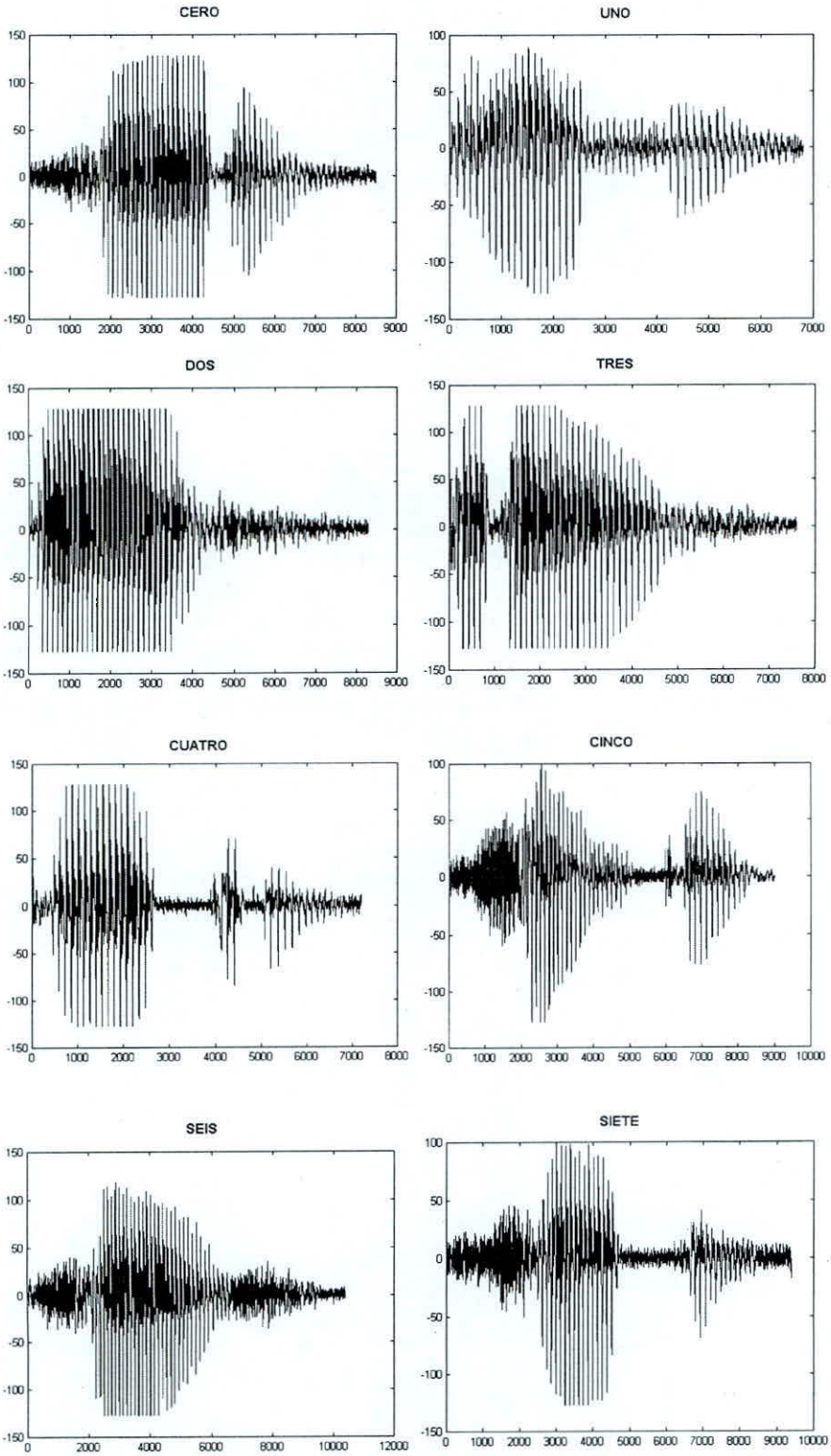

Figura 6. Gráficas de palabras delimitadas

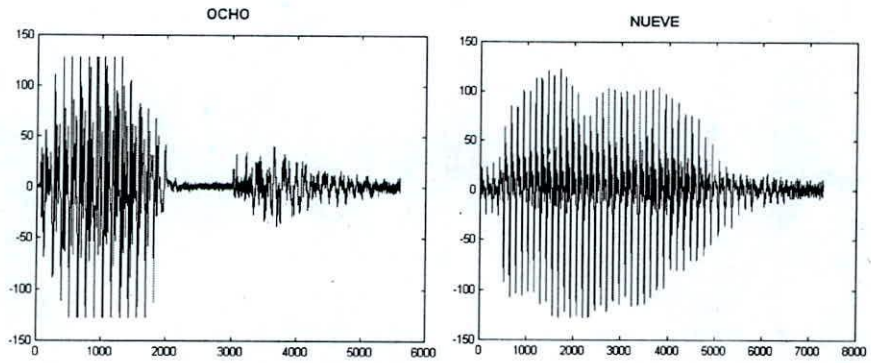

Figura 6 (Continuación).

IV. TIEMPO DE PROCESAMIENTO

Matlab, fué el lenguaje de programación que se utilizó para desarrollar el programa que nos permitió comparar el tiempo de procesamiento del algoritmo COPER y un algoritmo comúnmente utilizado, obteniendo como resultado que el algoritmo COPER requiere un menor tiempo de procesamiento. Se utilizó como muestra la palabra "Uno" que se almacenó en un vector de 8000 muestras. Seguidamente se lista el programa ***ComparaTiempos.m*** implementado, así como los resultados obtenidos.

ComparaTiempos.m

```
close all
clear all
clc

E=0;
Z=0;
CP=0;

EUI=30;
ZUI=20;
CPUI=6;

y=100*wavread('Uno.wav');

% Algoritmo Comúnmente Utilizado

disp('Algoritmo Comúnmente Utilizado')

TiempoInicio=cputime;
for n=2:1:length(y)
   E=E+y(n)*y(n);
   Z=Z+abs(sign(y(n))-sign(y(n-1)));
end
```

```
if (E>EUI | Z>ZUI)
   TiempoFinal=cputime;
   TiempoTotalEZ=TiempoFinal-TiempoInicio;
   sprintf('InicioDetectado. Tiempo=%2.2d',TiempoTotalEZ)
end
% Algoritmo COPER
disp('Algorimo COPER')

TiempoInicio=cputime;
for n=2:1:length(y)
   CP=CP+abs(y(n)*abs(y(n))-y(n-1)*abs(y(n-1)));
end

if (CP>CUI)
   TiempoFinal=cputime;
   TiempoTotalC=TiempoFinal-TiempoInicio;
   disp('Algorimo COPER')
   sprintf('InicioDetectado. Tiempo=%2.2d',TiempoTotalC)
end

disp('Resultado')

if TiempoTotalEZ<TiempoTotalC
   sprintf('Algoritmo Energia y Cruces por Cero es más rápido')
elseif TiempoTotalEZ>TiempoTotalC
   sprintf('Algoritmo COPER es más rápido')
elseif TiempoTotalEZ==TiempoTotalC
   sprintf('Ambos Algoritmos demoran el mismo Tiempo')
end
```

Resultados :

Algoritmo Comúnmente Utilizado
ans =
InicioDetectado. Tiempo=**14.23 seg**

Algoritmo Propuesto - COPER
ans =
InicioDetectado. Tiempo=**10.91 seg**

Observándose finalmente que el Algoritmo COPER es más rápido.

V. CONCLUSIONES

Se ha desarrollado un nuevo parámetro para realizar la Detección de Extremos más robusta. En las pruebas experimentales el algoritmo COPER ha demostrado mejoras notables en comparación al algoritmo de detección de extremos comúnmente utilizado. Además se demostró la reducción del tiempo de procesamiento, el cual permite que sea utilizado eficientemente en aplicaciones de tiempo real.

VI. BIBLIOGRAFÍA

- Jesús Bernal Bermúdez, Jesús Bobadilla Sancho, Pedro Gómez Vilda, "*Reconocimiento de voz y fonética acústica*". 2000 ALFAOMEGA GRUPO EDITOR. S.A.
- Andrés Flores Espinoza, "*Reconocimiento de Palabras Aisladas en Castellano*", Inictel. Dirección de Investigación y Desarrollo. 1993.
- C. Crespo Casas, C. de la Torre Munilla, J.C. Torrecilla Merchán. *"Detector de extremos para reconocimiento de voz"*. Comunicaciones de Telefónica I+D. Publicación de Telefónica I+D. S:A. Madrid España .2001. Ediciones OnLine: http://www.tid.es/presencia/publicaciones/comsid/esp/home.html
- Vinay K. Ingle, Jhon G. Proakis. *"Digital Signal Processing" Using Matlab V.4.* PWS Publising Company. 1997.
- The Math Works Inc. *"Matlab. Edición de estudiante"*. Prentice Hall. 1996.