Diseño y desarrollo de un Software para el análisis y procesamiento de señales de voz

H. Laforcada[*], D. Milone, C. Martínez, H. Rufiner
Laboratorio de Cibernética, Departamento de Bioingeniería,
Facultad de Ingeniería, Universidad Nacional de Entre Ríos,
CC47, Suc. 3, (CP 3100), Paraná, Entre Ríos, Argentina

Resumen—En el campo del análisis de señales de voz, el profesional o investigador se encuentra a menudo frente a la necesidad de un software que le permita llevar a cabo estudios específicos, alterar parámetros poco frecuentes y presentar los resultados de manera apropiada. Las herramientas de este tipo actualmente disponibles no se ajustan apropiadamente a las necesidades específicas de profesionales relacionados con la voz (fonoaudiólogos, otorrinolaringólogos, lingüistas, investigadores, etc). Estas razones motivan el desarrollo de un nuevo software que satisfaga los requerimientos mencionados, y que a partir de la disponibilidad del código fuente sea posible realizar modificaciones sobre el mismo para adaptarlo a diferentes condiciones y usuarios.

En el presente trabajo se describe el diseño y desarrollo de un software para el análisis de señales de voz con las características antes citadas. Primeramente se expone una breve introducción al diseño y análisis orientado a objetos y se resaltan aquellos aspectos que son de mayor relevancia, sentando las bases para la comprensión de la nomenclatura e iconografía utilizada. A continuación se presenta el diseño del software, y se revisan conceptos matemáticos relativos al análisis y procesamiento de señales de voz. Luego se describen y exponen las diferentes funciones del software, y se muestran los resultados de la aplicación de esta herramienta al análisis de una señal de voces patológicas.

Todo este desarrollo se completa con la validación de esta herramienta por parte de los usuarios finales. La experiencia recogida permitirá realizar ajustes en el software, y los resultados finales serán objeto de un trabajo futuro.

Palabras clave—Análisis de señales de voz patológicas, diseño de software, procesamiento digital de señales.

I – Introducción

En los últimos años ha crecido el interés por el análisis de señales de voces normales y patológicas como un método alternativo de diagnóstico. Este tipo de análisis demuestra tener ventajas con respecto a los métodos de examinación actuales debido a su naturaleza no invasiva y a su potencial para dar información cuantitativa acerca del estado de las funciones de la laringe y el tracto vocal, con tiempos de análisis razonables [1]. Es sabido que la presencia de patologías en las cuerdas vocales causa cambios significativos en sus patrones de vibración normales, lo que impacta en la calidad de la producción de voz. Los problemas en la producción de voz pueden originarse en [2, 3]:

1) desorden funcional: debido al abuso o mal uso del sistema vocal anatómica y fisiológicamente intacto, corregido por medio de terapia de voz; o
2) patologías laríngeas: nódulos de las cuerdas vocales, pólipos, úlceras, carcinomas y parálisis del nervio laríngeo, corregidas por medio de terapia de voz, cirugía y, en algunos casos, radioterapia.

Se vuelve evidente la necesidad de un software altamente versátil que combine una interfaz amigable y poderosas herramientas de análisis de voz. Nuestro software fue diseñado para satisfacer estas necesidades, dado que siempre se tuvo en mente las visiones tanto del profesional del habla como del investigador. El desarrollo inicial de este software surgió como una evolución natural de nuestro trabajo de laboratorio [4, 5, 6, 7, 8].

Es un hecho que los profesionales del habla usualmente no necesitan conocer los parámetros técnicos que sí importan al investigador, y que usualmente no se interesan en la teoría matemática que yace detrás de un análisis, sino que solamente se interesan por los resultados. Como bioingenieros e investigadores estamos al tanto de estos hechos y, con ellos en mente, diseñamos diferentes perfiles para una fácil adaptación a estas dos diferentes necesidades. El diseño de software y algunos conceptos relacionados a este tema se describen en la Sección II. Se continúa con una breve exposición acerca de las rutinas de procesamiento en la Sección III, y la descripción funcional del software se muestra en la Sección IV.

[*] Correspondencia: cyberlab@fi.uner.edu.ar, hlaforcada@softhome.net

II – Diseño del software

A – Conceptos básicos del diseño orientado a objetos

El análisis y diseño orientado a objetos (DOO) es una poderosa herramienta que permite al programador dividir y manejar problemas complejos. Una *clase* es una abstracción de una parte del dominio del problema, que modela las características y el comportamiento de dicha parte. Las relaciones de *herencia* entre clases permiten el modelado de los aspectos básicos en la clase padre, y los aspectos particulares en las clases hijas. Esto da origen a dos importantes características del DOO: la *especialización*, donde las clases hijas son más específicas que la clase padre y el *polimorfismo* en donde conociendo al padre, se puede pedirle a cualquiera de sus hijos que se comporte como su padre.

Una clase es sólo un modelo, una abstracción. Cuando se desea utilizar sus funcionalidades, es necesario crear (usualmente referido como *instanciar* en la jerga del DOO) o usar un *objeto*. Por lo tanto, se dice que un objeto es la ocurrencia específica de una clase, es la materialización de la abstracción. Más acerca del DOO puede ser encontrado en [9] y [10].

B – Descripción del diseño

Se comienza esta sección con una enumeración de los módulos principales del software, seguido por una breve descripción de cada uno. Un módulo es un bloque de un programa que agrupa variables, objetos y funcionalidades. Las relaciones entre los módulos definen la dinámica del programa. En el marco de la programación orientada a objetos (POO), en general un módulo se corresponde con una clase que agrupa otras clases y funcionalidades para la interrelación entre las clases contenidas y entre estas clases y otros módulos.

El módulo principal del software es *MainForm*, el cual es el responsable de colectar y coordinar la información de otros formularios y de la unidad de procesamiento, así como la manejar e interpretar la interacción del usuario. Para poder mostrar las señales apropiadamente, se diseñaron tres diferentes formularios que contienen y grafican tres clases diferentes de señales. *Signal1DForm*, *Signal2DForm* y *SignalCForm* contienen, grafican y poseen métodos para manipular señales unidimensionales con valores reales, señales bidimensionales con valores reales o complejos y señales unidimensionales con valores complejos respectivamente. Un formulario especial es *RealTimeForm*, el cual está diseñado para la visualización y grabación de señales en tiempo real utilizando un micrófono. Este formulario grafica la señal monitoreada, su espectro de magnitud, su espectro por LPC y su espectrograma. Cuando se necesita del ingreso de parámetros por parte del usuario, se muestran formularios de diálogo especialmente diseñados y se pide al usuario que ingrese datos. Un cuadro de diálogo de configuración general accesible desde el menú principal, permite al usuario configurar parámetros de análisis y del comportamiento del software, y ofrece la opción de establecer valores por defecto para automatizar los análisis.

El manejo de datos necesita de estructuras adecuadas para funcionar correctamente. Para satisfacer los requerimientos impuestos por los diferentes tipos de señales involucradas en el análisis de señales de voz, se diseñó una estructura de clases (Figura 1) que se adapta a cada tipo. Para evitar la incompatibilidad de datos y haciendo uso de la potencia del polimorfismo, el diseño parte de una clase base que modela una señal, *TSignal*, y baja por el árbol de herencia a medida que las características de las señales crecen en complejidad y especialización.

Los dos hijos de primer nivel son *TSignal1D* y *TSignal2D*, que modelan estructuras de datos unidimensionales y bidimensionales respectivamente. Cada uno tiene descendientes que modelan el comportamiento y estructura de señales reales y complejas. Estas clases se adaptan bien cuando se necesita trabajar, por ejemplo, con señales complejas unidimensionales como las señales de espectros, o con señales complejas bidimensionales como las de LPCgramas. En el caso de *TSignal1DReal*, sus descendientes son implementados de acuerdo al tipo de dato usado como unidad fundamental de la señal; si es un valor de coma flotante entonces usamos *TSignal1DSingle*, y si es un valor entero empleamos *TSignal1DInt*. Este último se vuelve importante cuando necesitamos trabajar con señales de voz de diferente cuantización, esto es, 8 bits o 16 bits, para lo cual se diseñaron las clases *TSignal8* y *TSignal16* respectivamente.

El potencial de procesamiento del software está basado en el módulo de procesamiento digital de señales, denominado *DSP*. Para optimizar el uso de memoria, las rutinas de procesamiento están integradas en una biblioteca de enlaces dinámicos (dll), la que es cargada y utilizada por el módulo de procesamiento cuando es necesario. Esta biblioteca fue escrita en C e incluye tanto rutinas básicas de procesamiento y manipulación de señales así como rutinas complejas y específicas para el análisis y procesamiento de señales de voz. Los conceptos matemáticos que yacen detrás de estos algoritmos de análisis son descriptos brevemente en la siguiente sección.

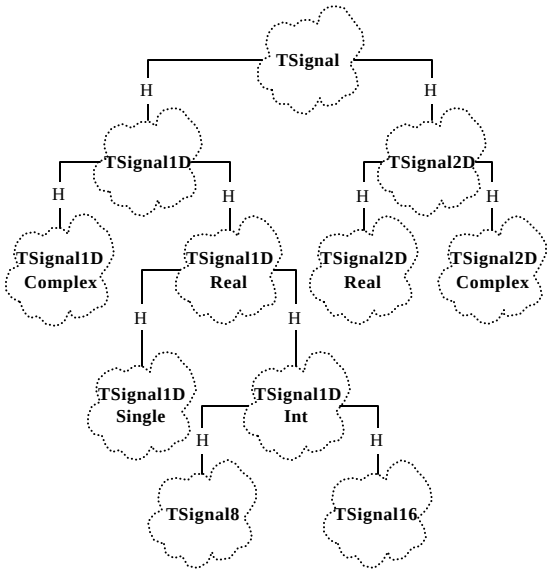

Figura 1: Diagrama de herencia de las clases utilizadas para modelar señales.

III – Procesamiento digital de señales

En esta sección se revisarán aquellos conceptos que son menos frecuentes en el campo del procesamiento de señales en general, y que son de especial interés en el procesamiento de voz. Los conceptos concernientes al análisis convencional serán meramente listados, pero no se profundizará en su descripción. Más acerca de estos conceptos puede encontrarse en [11].

Las transformaciones básicas y análisis que ofrece el software son:

- Análisis espectral
- Integración por bandas
- Análisis de predicción lineal
- Análisis cepstral
- Estimación de la frecuencia fundamental
- Análisis de cruces por cero

A partir de aquí se describirá el análisis de predicción lineal (PL) y el análisis cepstral.

Es posible modelar el tracto vocal utilizando un sistema autorregresivo, como:

$$\hat{v}(n) = -\sum_{j=1}^{Na} a(j)\, v(n-j) + G\, g(n) \quad (1)$$

donde $v(n)$ es la señal a modelar, $\hat{v}(n)$ es la señal estimada por el modelo, $g(n)$ es la entrada del tracto vocal y Na es el orden del sistema. En este modelo se puede considerar inicialmente que la entrada es igual a cero, y (1) puede ser escrita en su forma vectorial como:

$$\hat{v}(n) = -(\mathbf{v}_t^n)^T \mathbf{a}_t$$

donde $\mathbf{a}_t$ contiene los Na coeficientes y $\mathbf{v}_t^n$ contiene las últimas Na salidas $v(n-j)$. El error entre $\hat{v}(n)$ y $v(n)$ puede ser medido usando la distancia euclídea como:

$$e(n)^2 = (v(n) - \hat{v}(n))^2.$$

Para encontrar el vector $\mathbf{a}_t$, debe minimizarse el error cuadrático total entre $\hat{v}(n)$ y $v(n)$:

$$\boldsymbol{x}^2 = \sum_n e(n)^2 = \sum_n (v(n) + (\mathbf{v}_t^n)^T \mathbf{a}_t)^2.$$

A partir de $\nabla \boldsymbol{x}^2 = 0$ se obtiene:

$$\left(\sum_n \mathbf{v}_t^n (\mathbf{v}_t^n)^T \right) \mathbf{a}_t = -\sum_n \mathbf{v}_t^n v(n)$$

el cual es conocido como el sistema de Wiener-Hopf y es representado comúnmente como:

$$\mathbf{R}_t \mathbf{a}_t = -\mathbf{r}_t \qquad (2)$$

donde $\mathbf{r}_t$ es el vector de autocorrelación y $\mathbf{R}_t$ es la matriz de autocorrelación de $v(n)$. Puede mostrarse que $R_{ij} = r_{i-j}$ y entonces $\mathbf{R}_t$ es una matriz Toeplitz. El algoritmo de Levinson-Durbin aprovecha este hecho para simplificar la resolución del sistema. En cuanto a la determinación del orden óptimo para el sistema, se pueden utilizar varios métodos para resolver el compromiso entre la complejidad del sistema y el error total. Estos métodos están basados en medidas del error durante la predicción, de (1) y (2) podemos obtener:

$$E(N_a) = r_0 + \mathbf{r}_t^T \mathbf{a}_t$$

y una vez que se ha encontrado el sistema más simple que implique el mínimo $E(N_a)$ es posible determinar el orden apropiado para la estimación. Otros métodos [12] usan criterios basados en la teoría de la información. Asumiendo una distribución Gaussiana de la señal, puede medirse el error de acuerdo a:

$$I(N_a) = \log E(N_a) + \frac{2N_a}{N_e}$$

donde N_e es el número efectivo de muestras, el cual, en el caso de una ventana Hamming es igual a 0,4.

El software posee rutinas para el análisis de PL, cálculo de los coeficientes de PL, de los coeficientes de reflexión, la energía residual para el orden dado y la energía total de la señal. También permite calcular la respuesta en frecuencia del sistema a partir de los coeficientes de PL, la cual es utilizada en un análisis tiempo-frecuencia similar al análisis de Fourier de tiempo corto (espectrograma) llamado LPCgrama.

El cepstrum real (CR) de una señal $v(m)$ puede ser definido utilizando la transformada discreta de Fourier (TDF) como:

$$c(m) = T_F^{-1}\left\{ \log\left| T_F\left\{ v(m) \right\} \right| \right\}. \qquad (3)$$

Esta definición puede ser extendida reemplazando la TDF en (3) y su inversa (TIDF) por su definición:

$$c(k) = T_F^{-1}\left\{ \log\left| \sum_{n=1}^{Nv} v(n)\, e^{-j(2p/N_v)(k-1)(n-1)} \right| \right\}$$

$$= \frac{1}{N_u} \sum_{k=1}^{Nu} \log\left| u(\hat{e}) \right| e^{j(2p/N_u)(k-1)(k-1)}.$$

Finalmente, si se considera que el argumento de la TIDF es una secuencia par de valores reales, puede simplificarse su cálculo usando la transformada coseno:

$$c(k) = \frac{2}{N_u} \sum_{k=1}^{Nu/2-1} \log\left| u(\hat{e}) \right| \cos((2\boldsymbol{p}/N_u)(\hat{e}-1)(k-1))$$

Si se considera al tracto vocal modelado por (1), se puede representar a la señal de voz para fonemas sonoros como la convolución:

$$\hat{v}(n) = g(n) * h(n)$$

donde la entrada al sistema es el tren de pulsos glóticos $g(n)$ y $h(n)$ es la respuesta impulsiva del sistema. La convolución de dos señales en el tiempo se corresponde con el producto de los espectros de dichas señales en la frecuencia, y dado que en el dominio frecuencial las componentes de los pulsos glóticos varían con la frecuencia mucho más rápidamente que la respuesta en frecuencia del tracto vocal, al aplicar la TIDF se separa el espectro de la entrada del espectro de la respuesta del sistema. Una vez aplicado el CR se puede utilizar una técnica de filtrado lineal para buscar el punto donde los espectros se separan, y en el caso de

señales de voz, este pico se da en un valor de frecuencia correspondiente con la frecuencia fundamental de la señal de voz, F_0, lo que permite estimar el valor de esta frecuencia. De esta manera se puede obtener información relativa a $h(k)$ en las primeras muestras e información relativa a $g(k)$ a partir de $1/F_0$.

IV – Resultados

En la Figura 2 se muestra un segmento de una voz normal correspondiente al fonema vocálico /aa/ sostenido y en la Figura 3 un segmento de una voz patológica correspondiente al mismo fonema. Se pueden apreciar las diferencias morfológicas entre estas señales. En la Figura 4 se observa el espectro de magnitud correspondiente a la señal de la Figura 2, y en la Figura 5 el de la señal de la Figura 3. A primera vista podemos apreciar la aparición de componentes de alta frecuencia en el caso patológico y una mayor energía en los armónicos correspondientes.

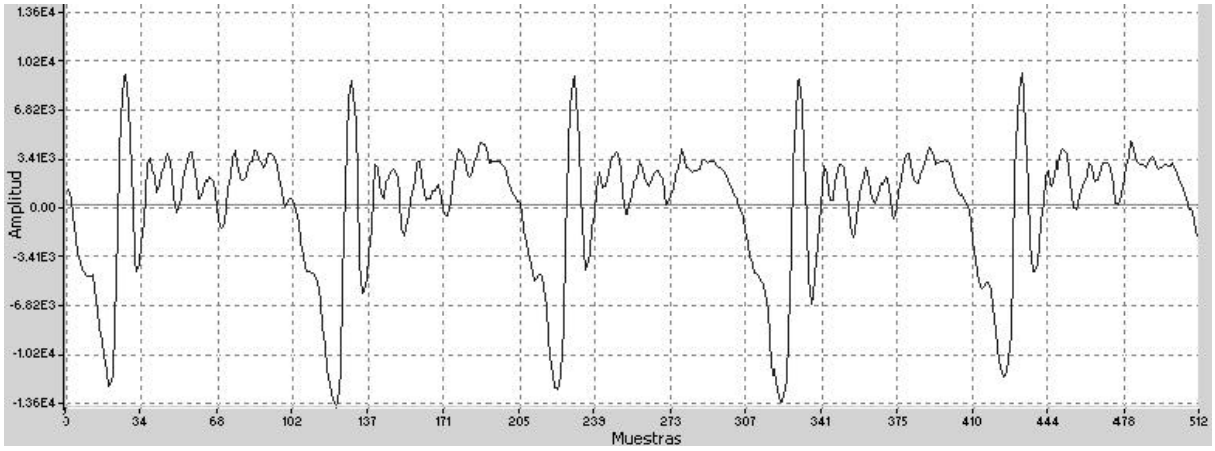

Figura 2: Secuencia temporal de una voz normal.

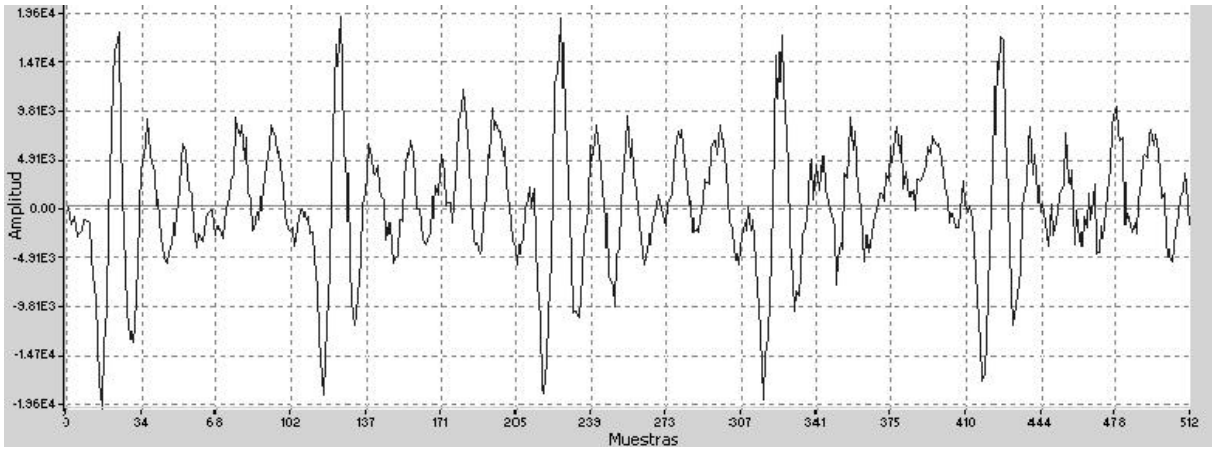

Figura 3: Secuencia temporal de una voz patológica.

Un análisis de las características frecuenciales dinámicas a lo largo del tiempo puede ser llevado a cabo utilizando el espectrograma. En la Figura 6 puede apreciarse un espectrograma de 0,8 segundos para una señal de voz normal, mientras que en la Figura 7 se muestra uno correspondiente a una señal patológica. Una vez más se observa la aparición de bandas de alta frecuencia en el caso patológico, y puede verse que la energía se concentra en diferentes bandas de frecuencia (zonas más brillantes).

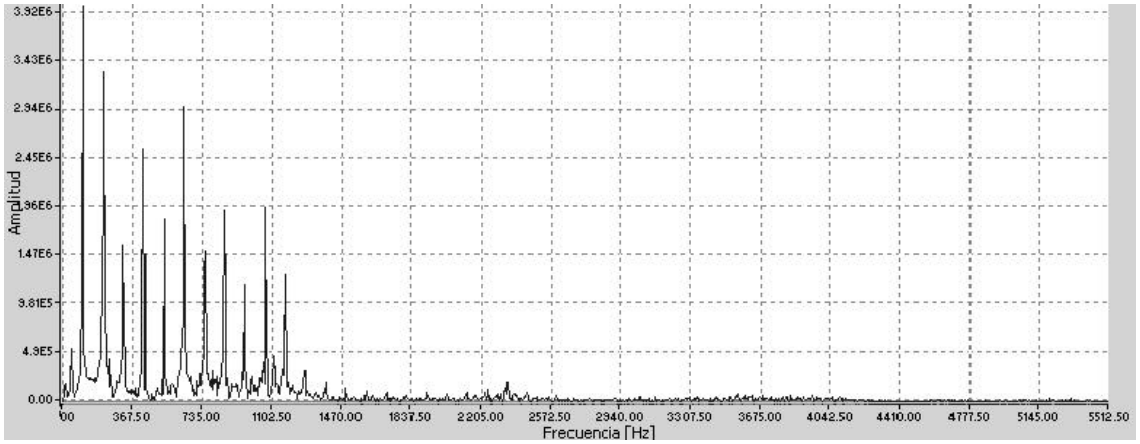

Figura 4: Espectro de magnitud de la voz normal de la Figura 2.

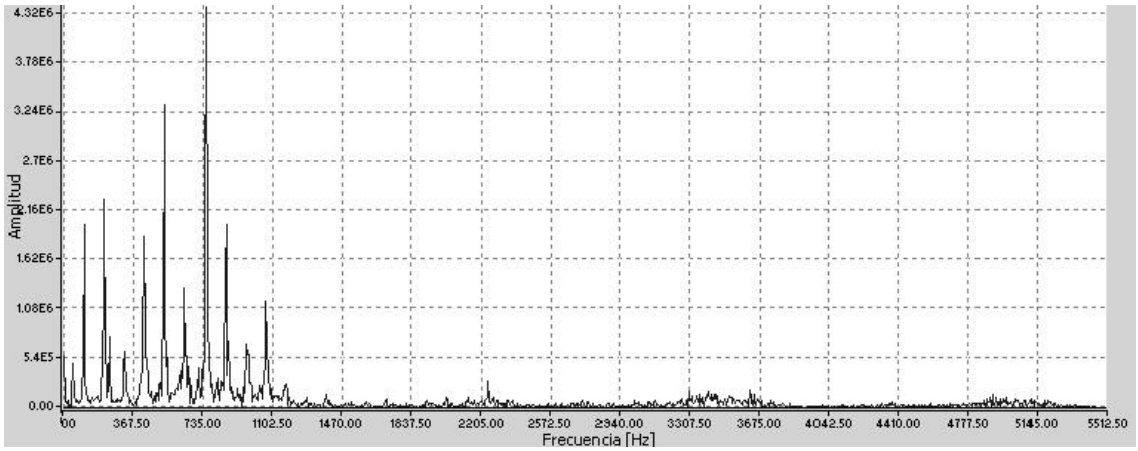

Figura 5: Espectro de magnitud de la voz normal de la Figura 3.

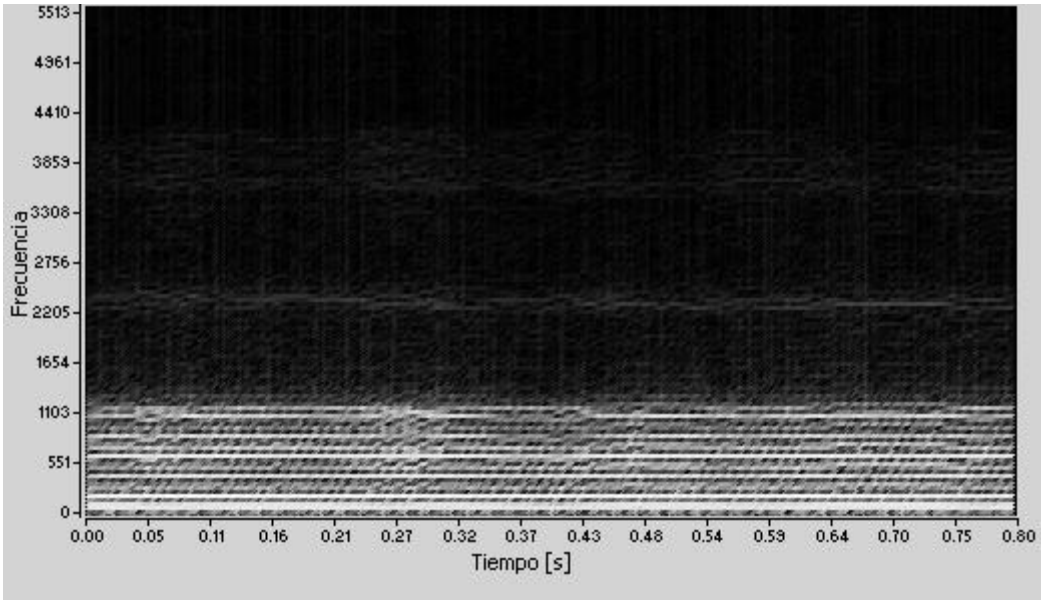

Figura 6: Espectrograma de una señal de voz normal.

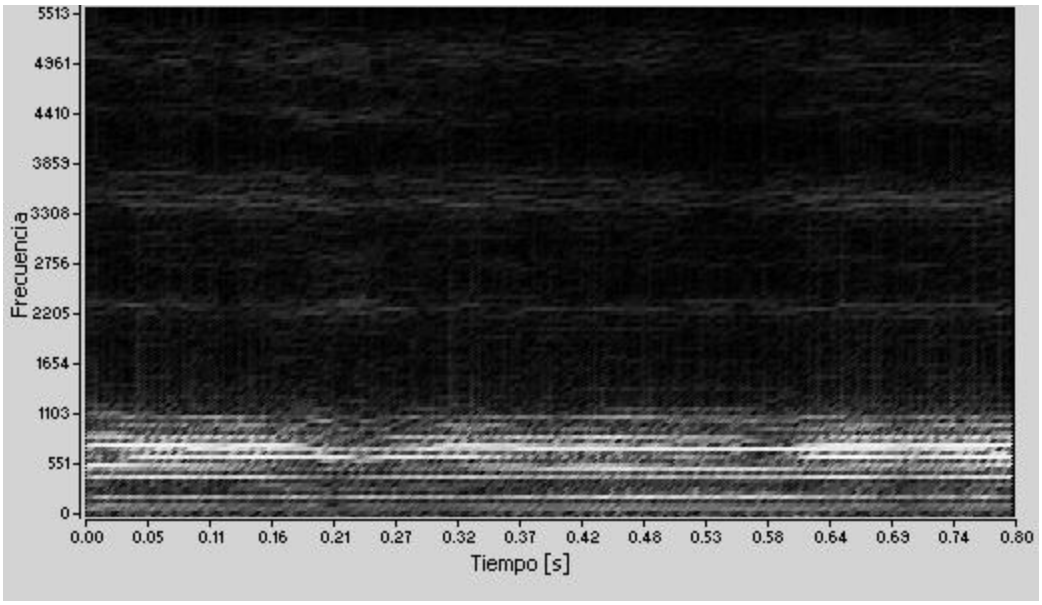

Figura 7: Espectrograma de una señal de voz patológica.

VI - Conclusión

Los profesionales consiguen resultados de alta calidad, rápidos y precisos cuando combinan su experiencia con el uso de herramientas poderosas. La detección y el tratamiento de patologías de la voz no son excepciones. Para poder conseguir estos resultados, las herramientas diseñadas e implementadas en nuestro software se basan en conceptos matemáticos sobre el procesamiento de señales y en técnicas de diseño y programación. Es por esto que nuestro software es de gran utilidad como herramienta de apoyo al diagnóstico para el profesional del habla así como la investigación en el área de señales.

Referencias

[1] C.E. Martínez, H.L. Rufiner, “Acoustic Analysis of Speech for Detection of Laryngeal Pathologies”, *Proceedings of the Chicago 2000 World Congress IEEE EMBS*, Paper # TH-Aa325-07, Chicago, USA, Julio 2000.

[2] P.W. Flint, C.W. Cummings, “The John Hopkins Center for Laryngeal and Voice Disorders”. Department of Otolaryngology-Head & Neck Surgery, John Hopkins University. Baltimore, Maryland, Agosto 1997.Available: http://www.med.jhu.edu/voice.

[3] James A. Koufman, Gregory N. Postma, “Center for Voice Disorders”. Wake Forest University, Noviembre 13, 1998. Available: http://www.bgsm.edu/voice.

[4] M. Argot, H.L. Rufiner, D. Zapata, A. Sigura, "Análisis Digital de Señales de Voz Orientado a la Rehabilitación Fonoarticulatoria". Fonoaudiológica. Asociación Argentina de Logopedia, Foniatría y Audiología. Tomo 39, Nº2, pp. 57-72, Octubre 1993.

[5] H.L. Rufiner, D. Zapata, A. Sigura, “Sistema de Adquisición, Procesamiento y Análisis de Señales de Voz”, *Anales del Primer Congreso Conjunto de Bioingeniería y Física Médica, VIII Congreso Argentino de Bioingeniería y III Workshop de Física Médica*, (SABI/SAFIM), Argentina, Octubre 1992.

[6] M. Argot, H.L. Rufiner, D. Zapata, A. Sigura, “Una Herramienta para la Investigación Fisiológica y Clínica de la Voz y el Habla”, *Anales del XVIII Congreso Latinoamericano de Ciencias Fisiológicas*, Uruguay, Abril 1994.

[7] H.L. Rufiner, D. Zapata, A. Sigura, “Análisis Digital de Señales de Voz Orientado a la Fonoaudiología y la Rehabilitación Foniátrica”, *Anales del X Congreso Nacional de Informática y Telecomunicaciones*, pp. 430-450, Capital Federal, Mayo 1993.

[8] H.L. Rufiner, J.M. Cornejo, M. Cadena, E. Herrera, “Laboratorio de Voz”, *Anales del VIII Congreso de la Asociación Mexicana de Audiología, Foniatría y Comunicación Humana*, Veracruz, México, Marzo 20-23, 1997.

[9] G. Booch, *Object-Oriented Analysis and Design with Applications*. Addison-Wesley, 1996.

[10] R. Wirfs-Brock, B. Wilkerson, and L. Wiener, *Designing Object-Oriented Software*, Englewood Cliffs, New Jersey, Prentice Hall, 1990.

[11] J. R. Deller, J. G. Proakis and J. H. Hansen, *Discrete Time processing of speech signals*, Macmillan Publishing, New York, 1993.

[12] H. Akaike, “A new look at the statistical model identification”, *IEEE Trans. on Automatic Control*, Vol. 19, No. 6, pp. 716–723, 1974.