ISSN 0798 1015

REVISTA ESPACIOS

HOME Revista ESPACIOS ÍNDICES A LOS AUTORES

Vol. 38 (Nº 17) Año 2017. Pág. 4

Algoritmo de recocimiento de comandos voz basado en técnicas no-lineales

Speech Recognition Algorithm based on nonlinear techniques

Julieth GÓMEZ-DURÁN 1; José SIMANCAS-GARCÍA 2; Melisa ACOSTA-COLL 3; Farid MELÉNDEZ-PERTUZ 4; Jaime VÉLEZ-ZAPATA 5

Recibido: 10/10/16 • Aprobado: 29/10/2016

Contenido

RESUMEN:

Los algoritmos de reconocimiento de voz son utilizados en aplicaciones de control inteligente especialmente en el área de la medicina. Una de las aplicaciones es el control de equipos de asistencia motora como las sillas de ruedas para pacientes con discapacidad motora total de sus extremidades inferiores y superiores. Las personas en esta condición tienen dificultad para trasladarse de un lugar a otro, dependiendo siempre de ayuda externa. El presente artículo describe el desarrollo de un algoritmo de reconocimiento de comandos de voz aplicando técnicas no lineales para la identificación de instrucciones de control en el desplazamiento de equipos de asistencia motora, con el fin de proporcionar un desplazamiento autónomo a personas con discapacidad motora total que pueden oír, ver y hablar. Las muestras de voz fueron recolectadas en un ambiente sin ruido y con ruido, aplicando un filtro digital para la eliminación del mismo. El procesamiento de la información filtrada se realizó mediante análisis rápido de Fourier y coeficientes cepstrales de Mel, y finalmente se aplicaron redes neuronales para el reconocimiento por voz de los comandos de control.
Palabras claves: Algoritmo de reconocimiento de voz, filtro digital, análisis rápido de Fourier, coeficientes ceptrales de Mel, redes neuronales.

ABSTRACT:

The speech recognition algorithms are used in intelligent control applications especially in the field of medicine. One application is the control of motor assistance equipment such as wheelchairs for patients with total motor disability of their upper and lower extremities. People in this condition have difficulty moving from one place to another, always depending on external aid. This article describes the development of an algorithm to recognize voice commands using nonlinear techniques for identifying control instructions for moving motor assistance equipment, in order to provide an independent movement to people with full motor disabilities who can hear, see and speak. The speech samples were collected in an environment without noise and noise by applying a digital filter for elimination. The filtered information processing was performed using fast Fourier analysis and Mel cepstral coefficients, and finally neural networks were applied for recognition by voice control commands.
Keywords: Voice reconigtion algorithm, digital filter, fast Fourier transform, Mel cepstral coefficients, neural networks.

1. Introducción

De acuerdo a las últimas estadísticas sobre discapacidad publicadas en Colombia por el DANE ("Información Estadística de la Discapacidad," 2004), de un total de 84.283 personas censadas, 1.036 respondieron que tenían deficiencias, de las cuales 32% correspondían a parálisis o ausencia de miembros superiores o inferiores. Estos números no son muy diferentes a nivel mundial, ya que los datos sugieren que más de mil millones de personas viven con algún tipo de discapacidad ("Informe Mundial sobre la Discapacidad: Resumen," 2011), entre las cuales 250.000-500.000 sufren una lesión en la médula espinal cada año y cerca de 190 millones (3,8% de la población mundial) sufren de alguna discapacidad grave (*International perspectives on spinal cord injury*, 2013). Estas personas se enfrentan a diversos obstáculos debido a su condición, pero quizás el principal problema al que se enfrentan es poder ejercer actividades básicas, como desplazarse, sin necesidad de la ayuda de otra persona. Esto hace que a menudo se sientan aislados de la sociedad, llevándolos a estados de depresión difíciles de controlar (M. Smith et al., 2013). Las sillas de ruedas eléctricas han significado una mejora en la calidad de vida de personas con discapacidad motora pero no son de utilidad para todos los discapacitados, ya que dependiendo de la complejidad de la condición, estos pierden la movilidad tanto de extremidades inferiores como superiores, como es el caso de las personas cuadripléjicas. Para dar solución a este problema se han estado desarrollando diferentes alternativas para el movimiento de equipos de asistencia motora y uno de ellos es la implementación de un sistema reconocimiento comandos de voz que permite el movimiento de equipos como la silla de ruedas solo con la voz del paciente (Komiya, Morita, Kagekawa, & Kurosu, 2000).

El proceso de reconocimiento de voz inicia con la conversión de una señal de voz a una secuencia de palabras, por medio de un algoritmo implementado como un programa de ordenador (Tahir & Ashfaque, 2009). Para ello, las técnicas no-lineales para el reconocimiento de voz son altamente utilizadas para sistemas que presentan no linealidades en la adquisición de la señal, en el canal de transmisión, y mecanismos de percepción humana como es el caso de adquisición y procesamiento de la voz ("On the relevance of bandwidth extension for speaker identification," 2002).

Este artículo describe en cuatro secciones el proceso para el reconocimiento de comandos de voz simples: "adelante", "atrás", "izquierda", "derecha", utilizando técnicas no lineales para el control de equipos de asistencia motora en pacientes con discapacidad total motora. La sección 2 describe los conceptos teóricos para el desarrollo de un algoritmo de reconocimiento de voz; la sección 3 detalla la metodología y se explican las etapas del algoritmo; luego la sección 4 presenta las pruebas y los resultados del algoritmo, y finalmente en la sección 5 se desarrollan las conclusiones.

2. Marco de referencia

A. Síntesis y reconocimiento de voz

El procesado digital de voz ha sido una de las áreas de mayor trabajo en el campo del procesamiento digital de señales. Para entender cómo reconocer los patrones de una señal de voz, se debe primero entender el funcionamiento del aparato fonador humano.

Las señales de voz están formadas por secuencias de sonidos. Estos sonidos y la transición entre ellos llevan la información que necesita ser transportada (Latinus & Belin, 2011). Estas secuencias se rigen por ciertas reglas, estudiadas por la lingüística. El estudio de la clasificación de los sonidos básicos pertenece a la fonética.

El aparato fonador humano está compuesto de dos partes principales: las cuerdas vocales (o glotis) y el tracto vocal. A su vez, este último se compone de 3 partes principales:

- La faringe – La cual conecta el esófago con la boca.
- La cavidad oral – La boca.
- El tracto nasal – Inicia en el velo del paladar y termina en las fosas nasales.

La fuente de energía viene de la presión de aire expulsada por los pulmones, bronquios y tráquea. La voz se produce cuando la onda acústica es irradiada de este sistema vocal, cuando el aire es expulsado de los pulmones y su flujo se ve perturbado por las constricciones en el tracto vocal. Cuando el velo del paladar se baja, el tracto nasal se acopla acústicamente al tracto vocal para producir sonidos nasales.

Clasificación de los sonidos

Las unidades básicas de sonido son denominadas fonemas y hay de dos tipos principales: vocales y consonantes. Las primeras son producidas cuando el tracto vocal se excita por pulsos de aire causados por las cuerdas vocales. La vibración es periódica por naturaleza y el período es el tono del sonido. La forma del tracto vocal determina las frecuencias resonantes del tracto, denominadas formantes. Las vocales tienen tres formantes entre las frecuencias de 200 Hz a 3 kHz, que varían de persona a persona ("Filtering for Vowels," n.d.).

En la producción de consonantes, las cuerdas vocales están relajadas, pero hay excepciones. Allí, el aire fluye hasta el tracto vocal sin la excitación periódica generada por las cuerdas.

Para poder sintetizar la voz artificialmente, se necesita un modelo de producción de voz tal como el que se describe en la figura 1:

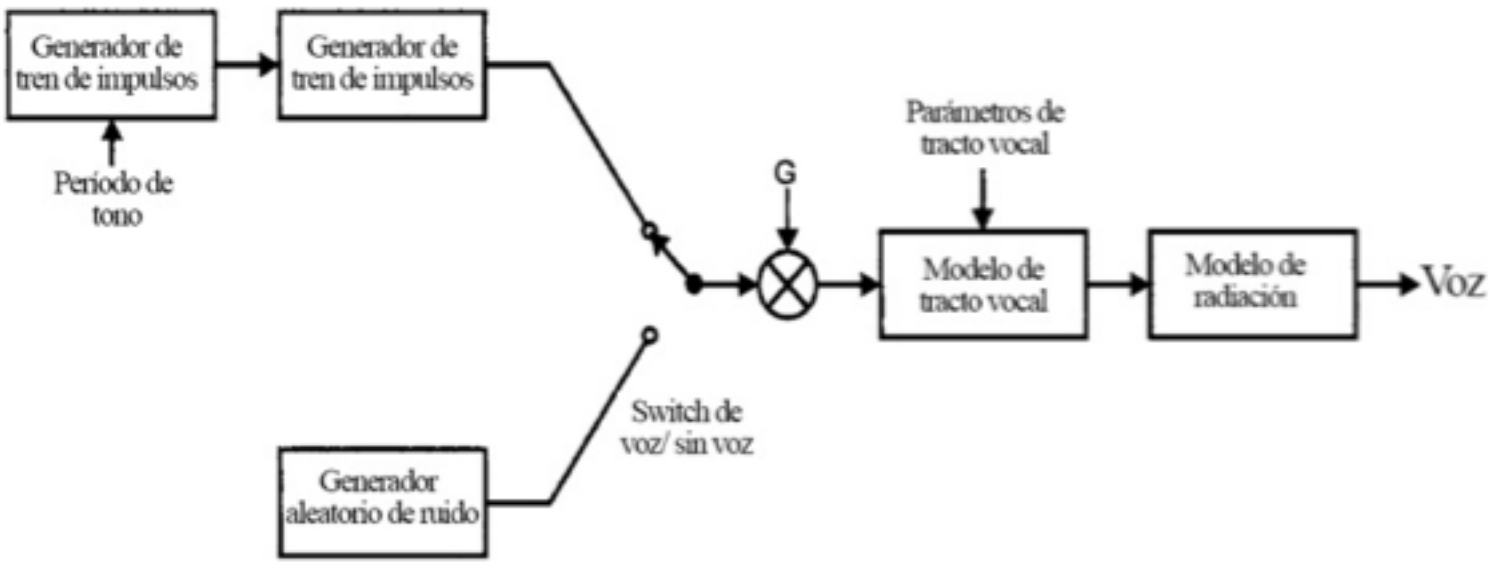

Figura 1: Modelo de producción de voz (Lai, 2003).

Los modelos de pulso glotal, vocal y de radiación son sistemas lineales en tiempo discreto, es decir, son filtros en tiempo discreto. Para sintetizar el sonido, el *switch* de voz/sin voz se debe conmutar a la fuente para el sonido en un determinado tiempo. Los parámetros de tracto vocal también requieren variar en el tiempo.

El modelo glotal de pulsos de mayor uso es el de Rosenberg (Rosenberg, 1971), cuya respuesta de impulso está dada por:

$$g(n) = \begin{cases} \frac{1}{2}\left[1 - \cos\left(\frac{\pi n}{N_1}\right)\right] & 0 \le n \le N_1 \\ \cos\left[\frac{\pi(n-N_1)}{2N_2}\right] & N_1 \le n \le N_1 + N_2 \\ 0 & \end{cases} \quad (1)$$

Donde *g(n)* es la respuesta al impulso del sistema lineal que tiene la forma de onda glotal deseada, y N1+N2+1 son los puntos que componen el pulso glotal. En la mayoría de los casos, el modelo de radiación es ignorado. El modelo de tracto vocal es usualmente un modelo

predictivo lineal porque la muestra de actual voz es generada de un número de muestras pasadas más la actual excitación. Esto puede ser descrito así:

$$s(n) = \sum_{k-1}^{y} a_k s(n-k) + u(n) \qquad (2)$$

Donde a_k es el coeficiente para el modelo y cambia de un fonema a otro, *u(n)* es la muestra de entrada al modelo de tracto vocal y *s(n)* es la salida del filtro adaptativo que implementa el tracto vocal, que equivaldría a la señal de voz (Lai, 2003).

B. Procesamiento no lineal de audio

El filtrado digital puede mejorar las señales de audio debido a que separa las frecuencias de la señal de las frecuencias de ruido. Este tipo de técnicas forman el eje central del procesamiento digital de señales. Se usa para reducir el ancho de banda del ruido en las señales de voz, lo cual incluye: ruido electrónico de circuitos analógicos, viento soplando en micrófonos, murmullo, entre otros. El filtrado lineal no es adecuado, ya que las frecuencias en el ruido se traslapan con las de la voz, en el rango de 200 Hz a 3.2 kHz, así que para hacerlo se debe revisar la amplitud de cada frecuencia. Si la amplitud es grande, pertenece probablemente a la voz y debe ser retenida. Si la amplitud es pequeña, correspondería a ruido y debería ser descartada; así mismo, los que se encuentran entre ambos extremos deben ser ajustados. Una de las soluciones a este problema sería con la implementación de un filtro adaptativo de Wiener, siempre y cuando se conozca el espectro de la señal y del ruido previamente, para que la respuesta pueda ser determinada. Esta técnica no lineal parte de la misma idea del filtrado lineal pero recalcula la respuesta en frecuencia del filtro en cada segmento de la señal.

La desventaja de esta técnica es que no es válido agregar-traslapar señales largas, ya que la respuesta en frecuencia del filtro cambia y la forma de onda en el dominio del tiempo no se alineará con los sistemas vecinos. Por ello es necesario dividir la forma de onda de la señal original en el dominio del tiempo entre los segmentos superpuestos, y una vez finalizado el procesado y antes de ser recombinados, se aplica una ventana suave a cada segmento.

La segunda técnica no lineal a considerar, se denomina procesado homomórfico de la señal que significa "la misma estructura" y consiste en separar las señales que han sido combinadas de forma no lineal convirtiéndolas a un sistema lineal. Esto puede ser modelado como una señal de audio *a[]*, que se multiplica por una señal lentamente variante *g[]*, que representa la ganancia variable. Lo anterior se puede realizar a través de un control automático de ganancia o con procesamiento digital no lineal.

Tal como se muestra en la figura 2, la señal de entrada []xg[], se pasa a través de una función logarítmica, resultando dos señales que son combinadas por adición, es decir, el logaritmo es la transformada homomórfica que vuelve el problema no lineal de multiplicación en un problema lineal de adición. Luego, las señales sumadas son separadas por un filtro lineal convencional. Para el control automático de ganancia, la señal *g[]*se compone de frecuencias por debajo del rango de la señal de voz (200 Hz a 3.2 kHz). El logaritmo de estas señales tendrá un espectro que sigue el mismo principio de eliminar el componente variante de la ganancia de la señal. Por último, para producir la señal de salida se aplica la función exponencial o antilogaritmo(S. W. Smith, n.d.).

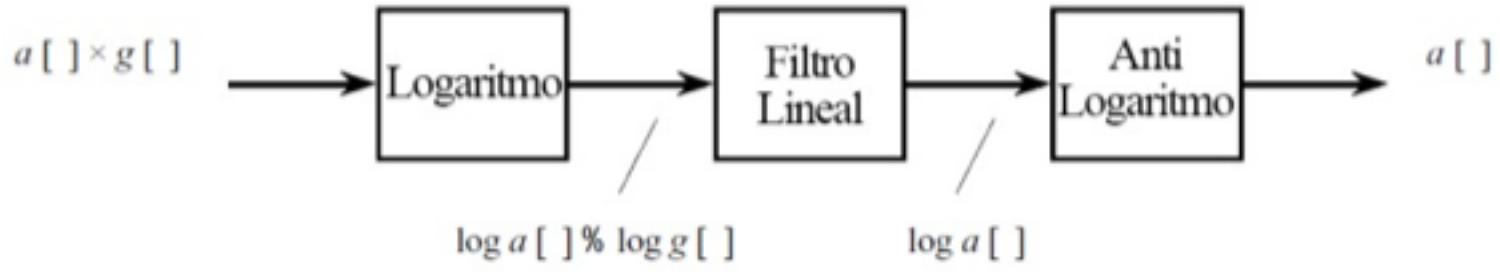

Figura 2: Separación homomórfica de señales multiplicadas (S. W. Smith, n.d.).

C. Redes neuronales

Las redes neuronales son sistemas de procesamiento, hardware o software, que copian esquemáticamente la estructura neuronal del cerebro para tratar de reproducir sus capacidades. Por ello, son capaces de aprender de la experiencia a partir de señales provenientes del exterior, dentro de un marco de computación paralela y distribuida. (Del Brío & Sanz Molina, 2007).

Estos sistemas se basan en el sistema neuronal biológico, y por tanto puede realizarse con una estructura jerárquica similar en donde el elemento principal es la neurona artificial o perceptrón.

D. Recursos disponibles

En el presente proyecto se utilizó la herramienta MATLAB®, y el Neural Network Toolbox. Este *toolbox* provee funciones y aplicaciones para modelar sistemas no lineales complejos, que no pueden ser fácilmente modelados con la ayuda de una ecuación. El *toolbox* permite la implementación de aprendizaje supervisado, con redes dinámicas, de alimentación directa o de base radial. También permite aprendizaje no supervisado, con mapas auto-organizados y capas competitivas. Así mismo, se puede diseñar, entrenar, visualizar y simular redes neuronales. Las aplicaciones van desde ajuste de datos, pasando por reconocimiento de patrones, clustering, predicción de series temporales hasta modelado y control de sistemas dinámicos ("Neural Network Toolbox - MATLAB - MathWorks España," n.d.).

---

3. Algoritmo de reconocimiento de patrones de voz

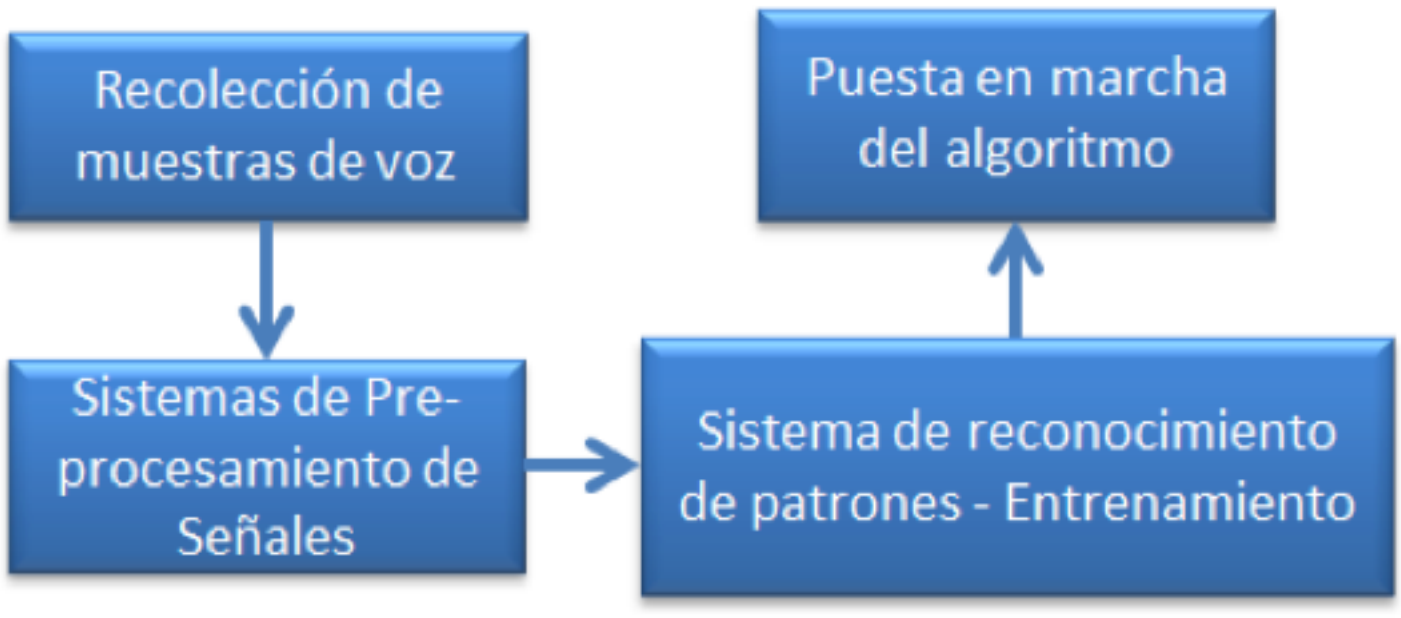

Figura 3. Algoritmo

Recolección de muestras de voz: Se toman varias muestras de voz, tanto en ambientes silenciosos como con ruido. Cada muestra debe hacerse con los siguientes comandos: Alto, derecha, izquierda, atrás, adelante, en archivos separados '*.wav*', de tal manera que puedan ser leídos fácilmente por MATLAB®. El uso de este algoritmo en implementaciones futuras en dispositivos físicos implicará la realización de esta etapa para cada uno de los usuarios del sistema durante el entrenamiento.

Pre-procesamiento de Señales: Consiste en un filtro digital que procesa las señales por medio de la Transformada Rápida de Fourier, con el fin de eliminar el ruido externo de las

señales obtenidas en la etapa de recolección, así como el análisis de los coeficientes cepstrales de Mel.

Sistema de reconocimiento de patrones: Está formado por redes neuronales (Hopfield ó RBF). Se entrena el algoritmo con las muestras recolectadas y procesadas, para enseñarle a reconocer los patrones de los comandos de voz que reciba.

Puesta en marcha del algoritmo: Una vez puesto en marcha el algoritmo, se deben hacer varias pruebas con el fin de detectar la tasa de error del algoritmo implementado, y así compararlo con otras técnicas utilizadas en proyectos similares. Las pruebas deben hacerse, al igual que la recolección de muestras, tanto en ambientes silenciosos como en ambientes con ruido.

A. Recolección de muestras de voz

Se tomaron 20 muestras para cada comando, almacenadas en archivos independientes .wav, con una velocidad de muestreo de 16 kHz y un factor de ganancia 1.4x. Las señales fueron tomadas utilizando la aplicación *Smart Voice Recorder* para tecnología*Android*.

Cada muestra tiene diferente longitud, amplitud y empiezan en diferentes puntos, tal como se puede observar en la figura 4, donde se colocan sobre un mismo plano:

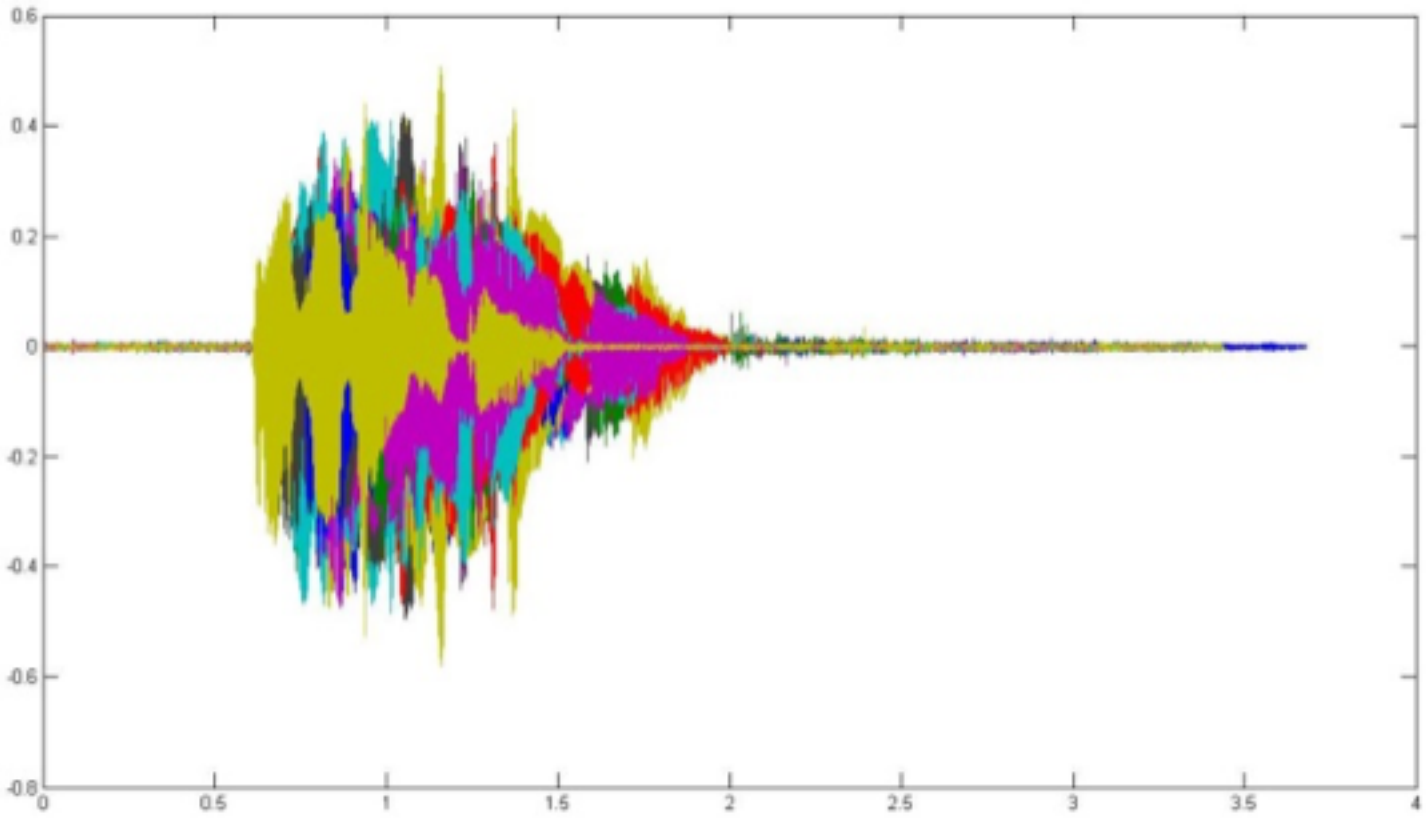

Figura 4: Representación en el dominio del tiempo de todas las muestras del comando "adelante"

B. Sistema de pre-procesamiento de señales

Luego de la toma de muestras es necesario pre-procesar las señales obtenidas. Aquí se debe entender cómo se representa mejor la información en las señales a procesar, con el fin de extraer los datos más relevantes para alimentar la red neuronal. Para ello se utiliza el Toolbox de procesamiento digital de señales de MATLAB®.

Para que el uso de la red neuronal funcione, es necesario asegurarse que el procesamiento de la señal arroje la menor cantidad de características suficientes como para que la red pueda identificar los sonidos y su implementación sea menos compleja (Lyons, n.d.). Por esta razón, se utiliza el análisis de los coeficientes cepstrales de Mel.

El próximo paso es calcular el espectro de frecuencia de cada tramo. El periodograma identifica las frecuencias presentes en la señal, aun así, el periodograma contiene información que no se

necesita para el reconocimiento automático de voz. Por este motivo, se toman grupos contenedores de periodograma y se suman para tener una idea de la energía presente en cada región de frecuencias. Esto se realiza por medio del banco de filtros Mel. El primero es angosto y da una indicación de cuánta energía existe cerca a los 0 Hz. A medida que las frecuencias aumentan, los filtros se ensanchan y las variaciones se vuelven menos significativas y es necesario tener una información aproximada de cuánta energía existe en cada punto. La escala de Mel explica exactamente cómo espaciar los filtros y qué tan anchos hacerlos.

Una vez se tienen las energías del banco de filtros, se debe calcular su logaritmo. Esto también está inspirado en el oído humano, ya que éste no escucha en una escala lineal y grandes variaciones de energía pueden no sonar tan diferente si el volumen es alto. Esta operación de compresión emula la forma en la que los humanos escuchan. Además, el logaritmo permite hacer una sustracción de la media cepstral como técnica de normalización de canal.

El último paso es computar la Transformada de Coseno Discreta (DCT del inglés *Discrete Cosine Transform*) al logaritmo de las energías del banco de filtros. Esto se hace porque los bancos se superponen, y el DCT des-correlaciona las energías para que las matrices de covarianza diagonal puedan ser utilizadas con redes neuronales. Hay que tener en cuenta que sólo 12 de los 26 coeficientes DCT se usan (Lyons, n.d.).

La escala de Mel relaciona la frecuencia percibida de un tono puro a su frecuencia medida real. Los humanos diferencian mejor pequeños cambios de tono en bajas frecuencias, que en altas frecuencias. La escala hace que las características se ajusten más a lo que los humanos escuchan, por medio de la siguiente ecuación:

$$M(f) = 1125 \ln\left(1 + \frac{f}{700}\right) \qquad (7)$$

Donde f es la frecuencia de los tonos componentes de la señal de voz. Los pasos necesarios para la implementación de esta técnica son:

Muestrear las señales en tramos de 20-40 ms (25 ms es estándar recomendado). Es decir, una señal de 16 KHz tendrá *0.025*16000 = 400* muestras. Si el archivo no se divide en un número parejo de tramos, se debe rellenar lo que falte con ceros. Esto se hace porque una señal de audio está constantemente cambiando, para simplificar el proceso estadísticamente se asume que en cortos períodos de tiempo la variación es poca.

Los próximos pasos se deben aplicar a cada tramo con el fin de extraer 12 MFCC coeficientes de cada uno. *S(n)* = Señal en el dominio del tiempo. Se debe tener que: *Si(n)* = Señal muestreada, donde *n* va de 1-400 e *i* oscila sobre el número de tramos; *Si(k)* = La señal resultante del cálculo de la transformada discreta compleja de Fourier, donde *i* denota el número del tramo que corresponde al tramo en el dominio del tiempo; *Pi(k)* = Es el espectro de potencia del tramo *i*.

Para calcular la Transformada Discreta de Fourier del tramo, se debe hacer lo siguiente:

$$S_i(k) = \sum_{n=1}^{N} s_i(n) h(n) e^{-j2\pi kn/N}, \ 1 \leq k \leq K \qquad (9)$$

h(n) = Es una ventana de análisis de muestra con *N* longitud. *K* = Es la longitud de la Transformada Discreta de Fourier.

El Periodograma estimado del espectro de potencia se obtiene de:

$$P_i(k) = \frac{1}{N} |S_i(k)|^2 \qquad (10)$$

Se debe tomar el valor absoluto de la transformada compleja de Fourier con 512 puntos y calcular el cuadrado del resultado, para mantener los primeros 257 coeficientes.

Calcular el filtro espaciado a Mel. Este es un set de 20-46 filtros triangulares que se aplican al

periodograma. Este filtro viene en la forma de 26 vectores con longitud 257. Cada vector es más que todo ceros excepto por cierta secciones del espectro. Para calcular las energías del filtro, se debe multiplicar cada uno por el espectro de potencia y después sumar los coeficientes. Como resultado se obtendrán 26 números que indican la energía presente en cada banco de filtros, tal como se observa en la figura 5:

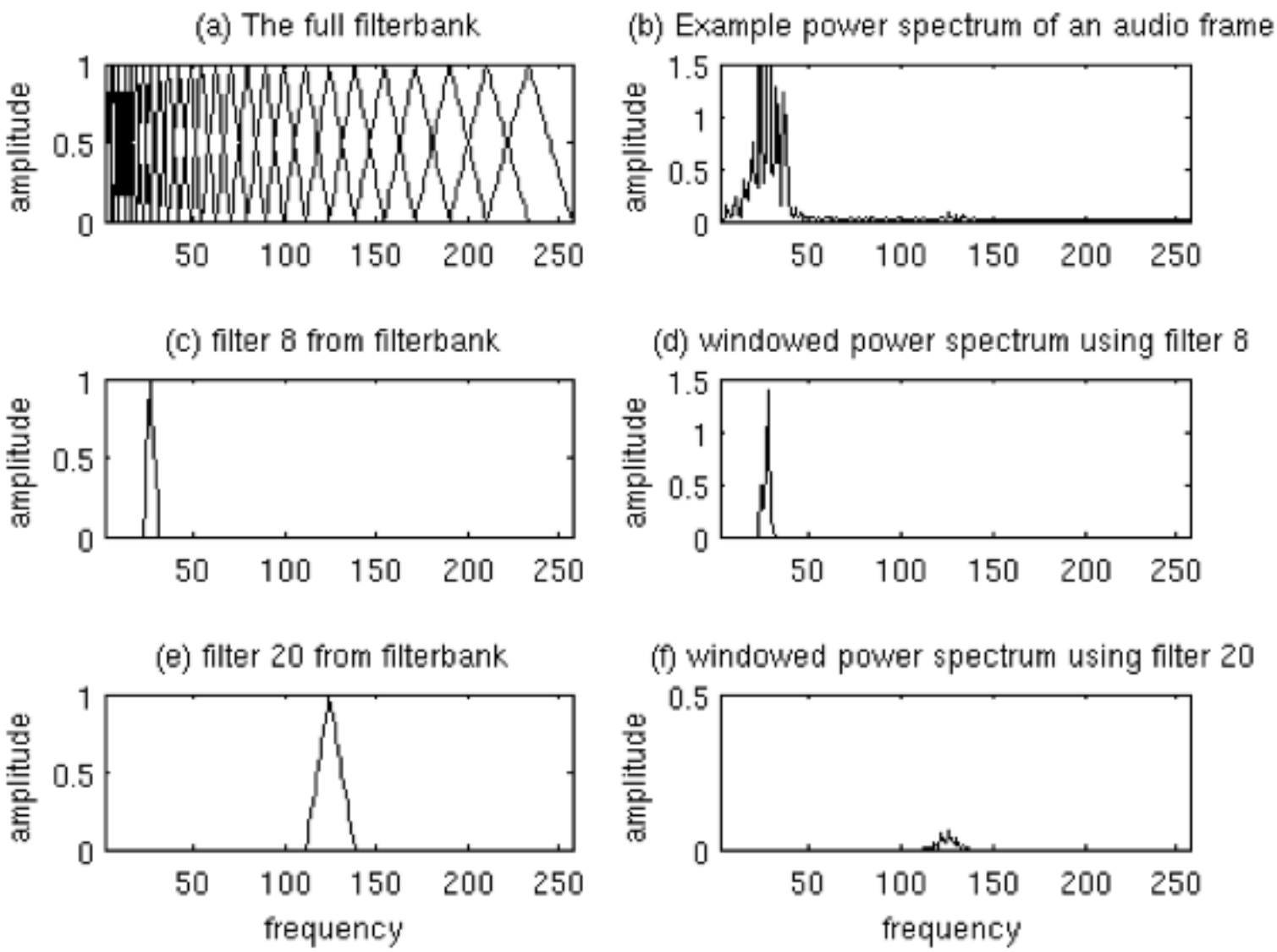

Figura 5: Banco de filtros de Mel y ventana de espectro de potencia.

Calcular el logaritmo de cada una de las 26 energías obtenidas. Calcular la Transformada de Coseno Discreta a las 26 energías para obtener 26 coeficientes Cepstrales. Para reconocimiento de voz, sólo son necesarios los primeros 12-13 coeficientes. Estos son los denominados Coeficientes Cepstrales en las Frecuencias de Mel (Lyons, n.d.).

Imitando la forma en la que los humanos escuchan, la escala de frecuencia Mel tiene un espaciamiento linear de frecuencia por debajo de los 1000 Hz, y un espaciamiento logarítmico por encima de los 1000 Hz. Las señales de voz tienen más energía en las frecuencias más bajas. La siguiente fórmula se usa para calcular los mels de una frecuencia dada en Hz:

$$mel(f) = 2595.^{10} \log\left(1 + \frac{f}{700}\right) \qquad (11)$$

Para cada tono con una frecuencia actual *f* Hz, un tono subjetivo se mide en la escala de mel. El pitch de un tono de 1 kHz, 40 dB por encima de la audiencia perceptual se conoce entonces como 1000 mels. En este caso sería:

$$mel(f) = 2595.^{10} \log\left(1 + \frac{f}{700}\right) = 181.312.111{,}042623$$

El cepstrum es la transformada directa de Fourier de un espectro. Es decir, es el espectro de un

espectro, y tiene ciertas propiedades que lo hacen útil en muchos tipos de análisis de señales. Uno de los atributos más destacados es el hecho de que cualquier periodicidad, o patrones repetidos, en un espectro serán detectados como uno o dos componentes específicos del cepstrum. Si el espectro contiene varios grupos de armónicos, puede ser confuso debido a la superposición. Pero en el cepstrum, se separarán de una manera similar a la que el espectro separa patrones repetitivos de tiempo en la forma de onda.

Para la implementación de la Red Neuronal, se usó el *MATLAB Neural Network Toolbox* disponible, para crear, entrenar y simular la red. Para cada palabra se utilizaron 30 muestras grabadas, de las cuales 20 fueron usadas para el entrenamiento y 10 para pruebas.

Definición de matriz de entrada y capas de la red

Esta red neuronal consta de 13 entradas, correspondientes a los 13 primeros coeficientes cepstrales de Mel calculados para cada señal, de acuerdo a la recomendación en (Lyons, n.d.). Para un óptimo desempeño de la red, se asignaron tres capas ocultas. Esto se evaluó por medio de ensayo y error hasta obtener la arquitectura de red más simple posible, con resultados satisfactorios. Además, se asignaron cuatro perceptrones a cada capa oculta de acuerdo a la respuesta que se desea obtener de la red.

Definición de matriz target

Alimentar la red neuronal con todos los puntos del espectrograma se podría convertir en una tarea bastante tediosa, ya que este consiste de aproximadamente 30.300 puntos, es decir 30.300 entradas de la red. Por lo tanto, se deben usar los trece (13) coeficientes obtenidos con el algoritmo MFCC. Los valores de entrada están en el rango de -5 hasta 1.5, y para cada neurona de entrada, se configura este parámetro. Todos estos valores se agregan a una matriz de entrada *input NN.* La matriz objetivo (o *target*) estaría conformada por cuatro bits de salida que activarían la respuesta de cada motor en la silla de ruedas, tal como se muestra en la tabla 1:

Tabla 1: Matriz Target de la red neuronal

Adelante	Izquierda	Derecha	Atrás
1	0	0	0
0	1	0	0
0	0	1	0
0	0	0	1

Entrenamiento de la red

Una vez definidas las matrices de entrada y target de la red, se puede proceder con la creación de la red en MATLAB® y su entrenamiento. Para ello se usan comandos especiales, disponibles para el *Neural Network Toolbox*. A continuación, se crea la red *feed-forward* de tres capas ocultas y una de salida, cada una de las cuales con cuatro neuronas, con sus respectivas funciones de aprendizaje. Además, se asigna la matriz de entrada y de target a la red. Hecho esto, se procede a inicializar y entrenar la red. De acuerdo a la gráfica en la figura 6, esta red logró la identificación plena de los comandos en la época 115.

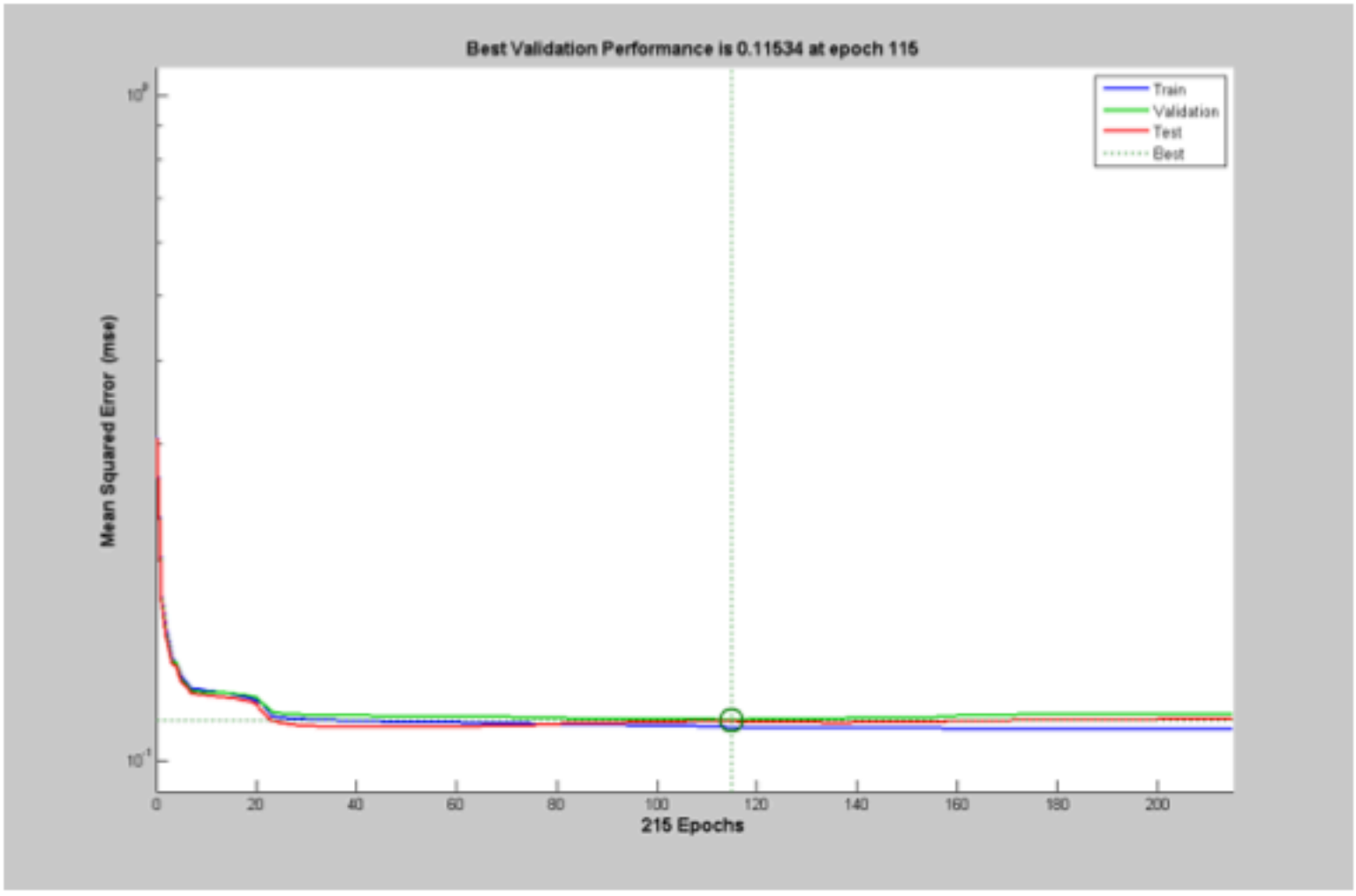

Figura 6: Rendimiento de la red neuronal creada.

En la figura 7 se puede ver que el valor final del coeficiente de gradiente después de la época 215 es 0.0047, lo cual es bastante aproximado a cero. Entre más cercano a cero se encuentre este valor, mejor será el entrenamiento y evaluación de las redes.

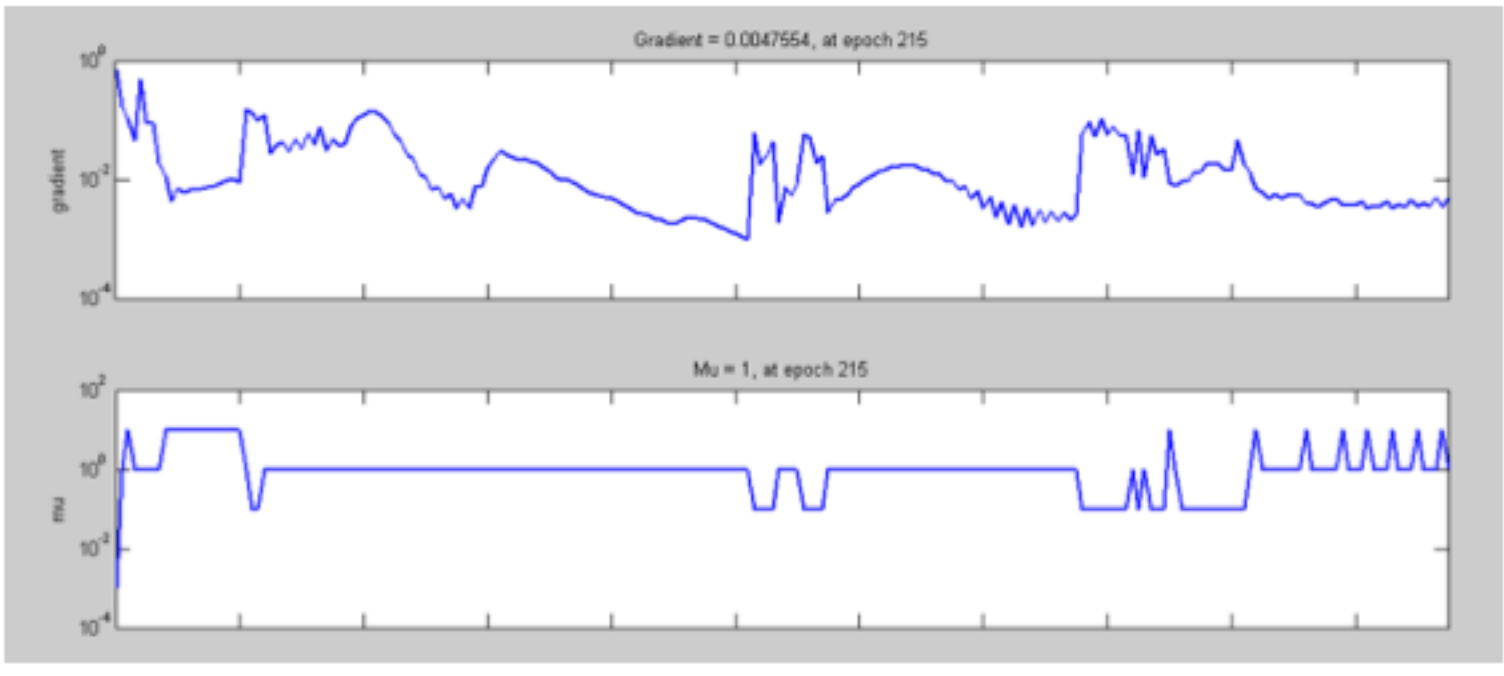

Figura 7: Estado del entrenamiento de la red creada.

---

4. Pruebas y resultados

A. Pre-procesamiento de la señal

Este es un paso muy importante, ya que de la calidad de las entradas dependerá la calidad de la red neuronal. El ruido y diferencias en amplitud de una señal pueden distorsionar la integridad de una palabra. Estos problemas tienen solución teniendo un adecuado pre-

procesamiento de la señal, compuesto de las siguientes etapas: Filtrado, Detección de entropía y Coeficientes Cepstrales en las Frecuencias Mel (Gevaert, Tsenov, & Mladenov, 2010).

Filtrado:

Las muestras fueron grabadas con un celular de micrófono estándar, por lo que contienen ruido causado por la respiración o el entorno. En esta etapa se debe eliminar el ruido de alta y baja frecuencia. La voz se sitúa principalmente en el rango de 300-3750 Hz, así que se puede aplicar un filtro FIR pasabanda sobre la señal, para una frecuencia de muestreo de 16 kHz.

Primero, se debe representar la señal en el dominio de la frecuencia. Para ello se utiliza la Transformada Rápida de Fourier, y se procede a representar la frecuencia normalizada (πrads/muestra). En la figura 8 se puede observar una comparación entre las tres señales obtenidas.

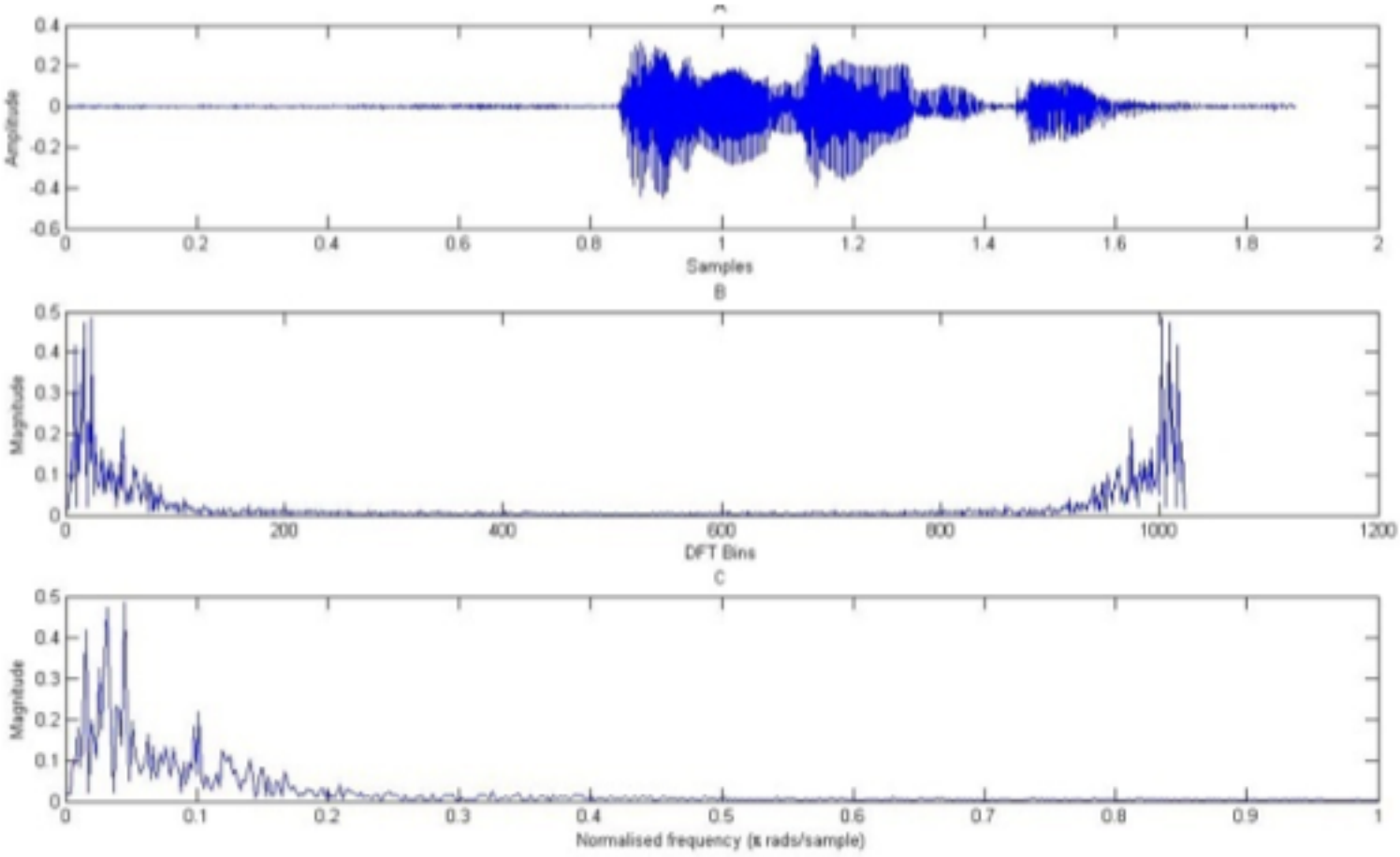

Figura 8: A. Forma de onda de la señal Adel; B. Representación de la señal en el dominio de la frecuencia; C. Representación de la frecuencia normalizada de la señal.

Para diseñar el filtro FIR a utilizar, se debe tener en cuenta que la herramienta de Matlab para el diseño de filtros requiere hacer la representación de la Frecuencia de acuerdo a Nyquist, FNyquist = Fs/2 = 8000 Hz, y las frecuencias de corte del filtro digital se deben normalizar con respecto a FNyquist.

Estos datos se utilizan para la implementación del filtro en MATLAB, obteniendo como resultado la respuesta en frecuencia que se observa en la figura 9:

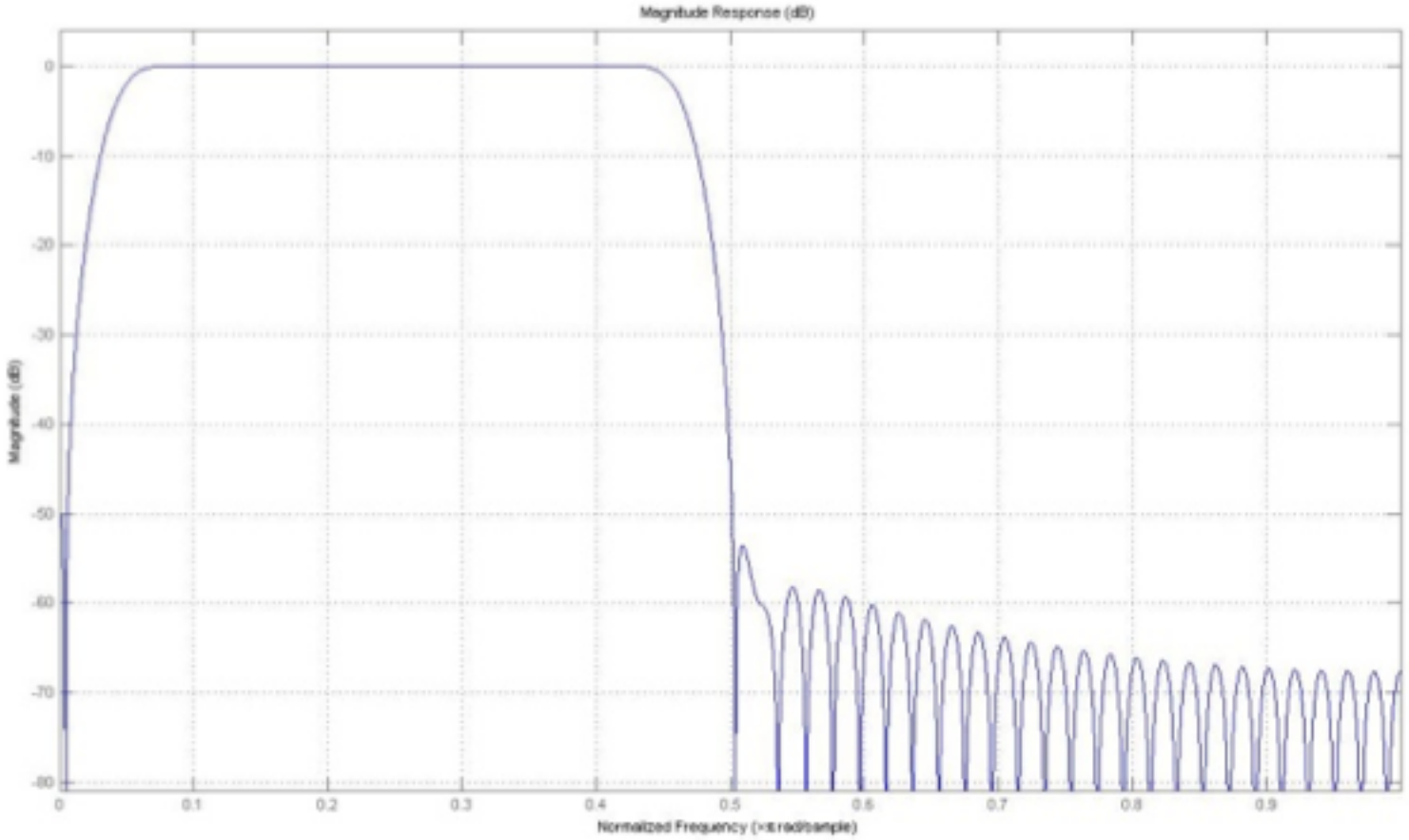

Figura 9: Respuesta en frecuencia del filtro pasabanda diseñado para remover frecuencias inaudibles o correspondientes a ruido genéricos del entorno.

Una vez diseñado el filtro, se procede a pasar la señal. El resultado obtenido para el primero comando *Adel1* se muestra en la figura 10.

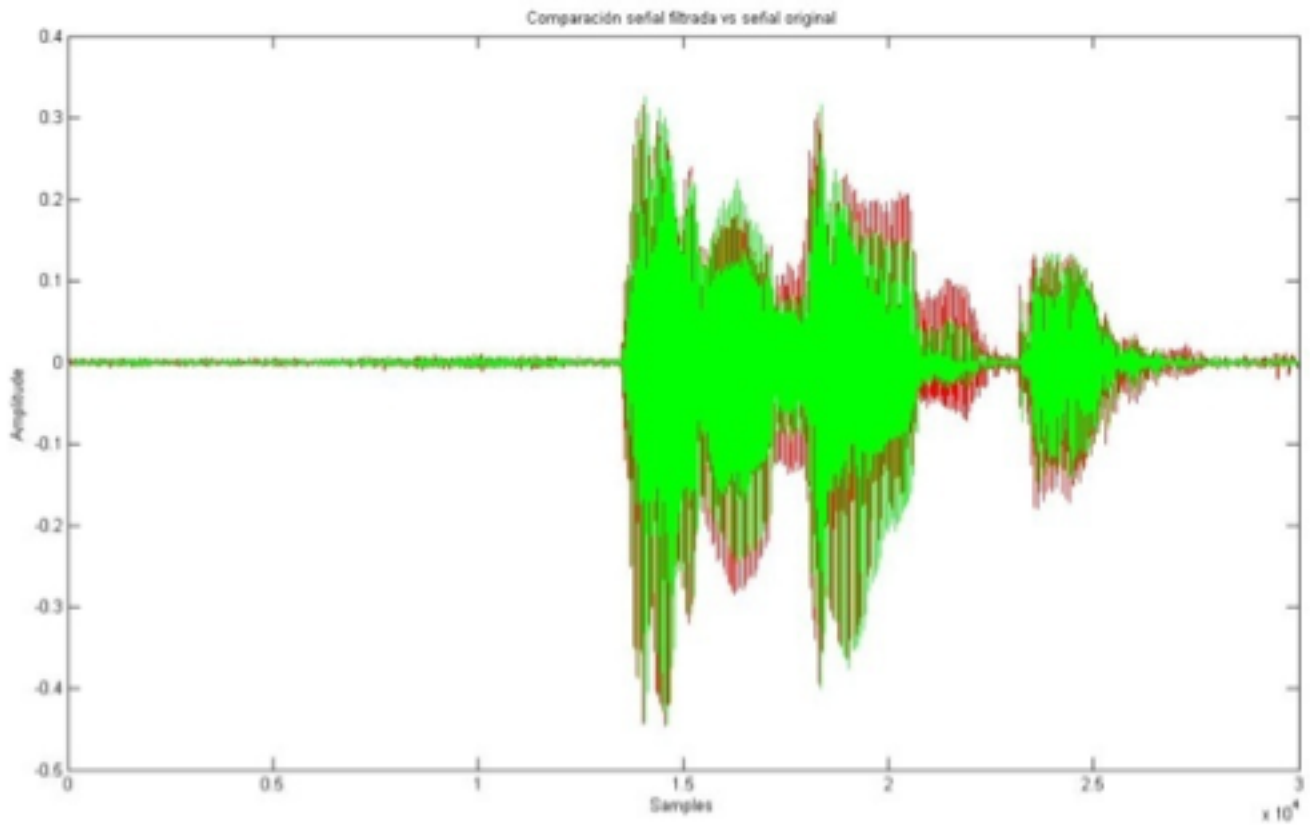

Figura 10: Comparación formas de onda de señal original (en rojo) con señal filtrada (en verde).

Este proceso se debe realizar con cada una de las señales de audio para los comandos adelante, atrás, izquierda, derecha y alto. En las figura 16se puede observar las señales filtradas vs. Originales para los comandos “adelante”.

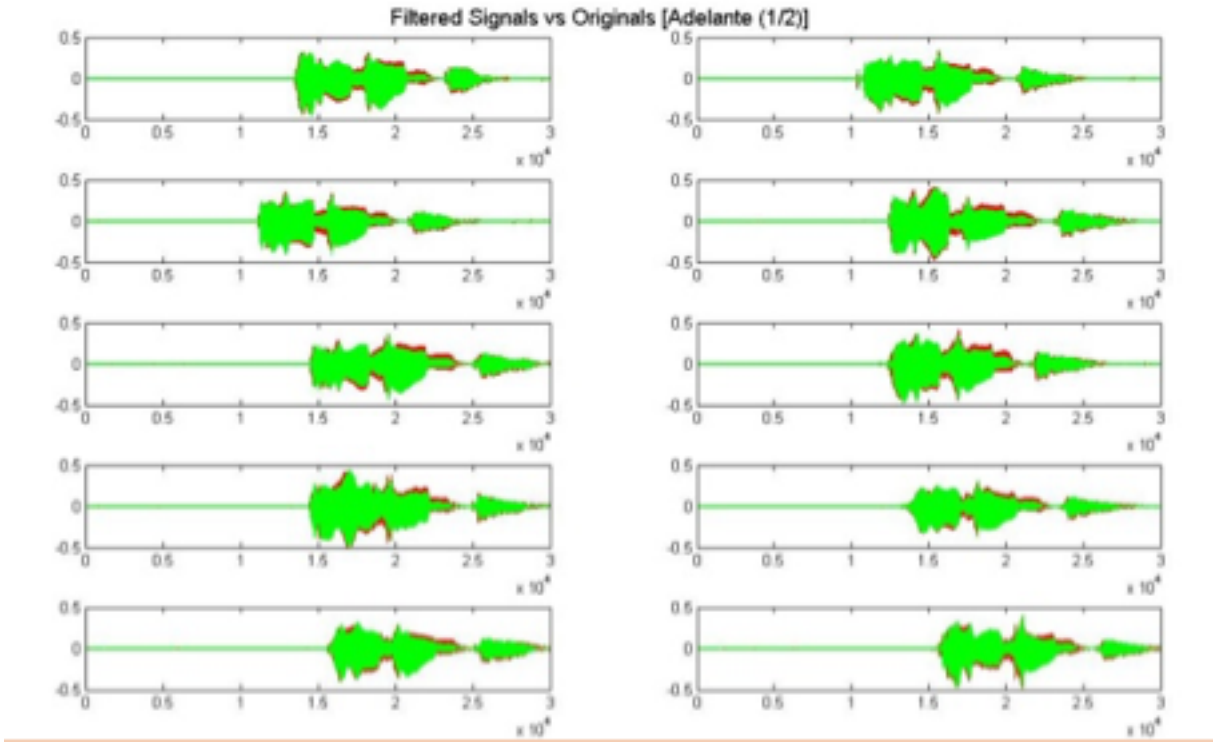

Figura 11: Señales filtradas del comando Adelante.

Detección de punto final

Una de las partes más difíciles del reconocimiento de voz es la detección del inicio y fin de una palabra. La detección basada en el cálculo de la desviación estándar de la señal, es un método sencillo y efectivo para encontrar el inicio de la información relevante de los comandos (Saha, Chakroborty, & Senapati, n.d.)(Tiwari, Pandey, & Shrestha, 2011).

En la figura 12 se pueden observar los resultados de la detección en las formas de onda de los comandos Adelante:

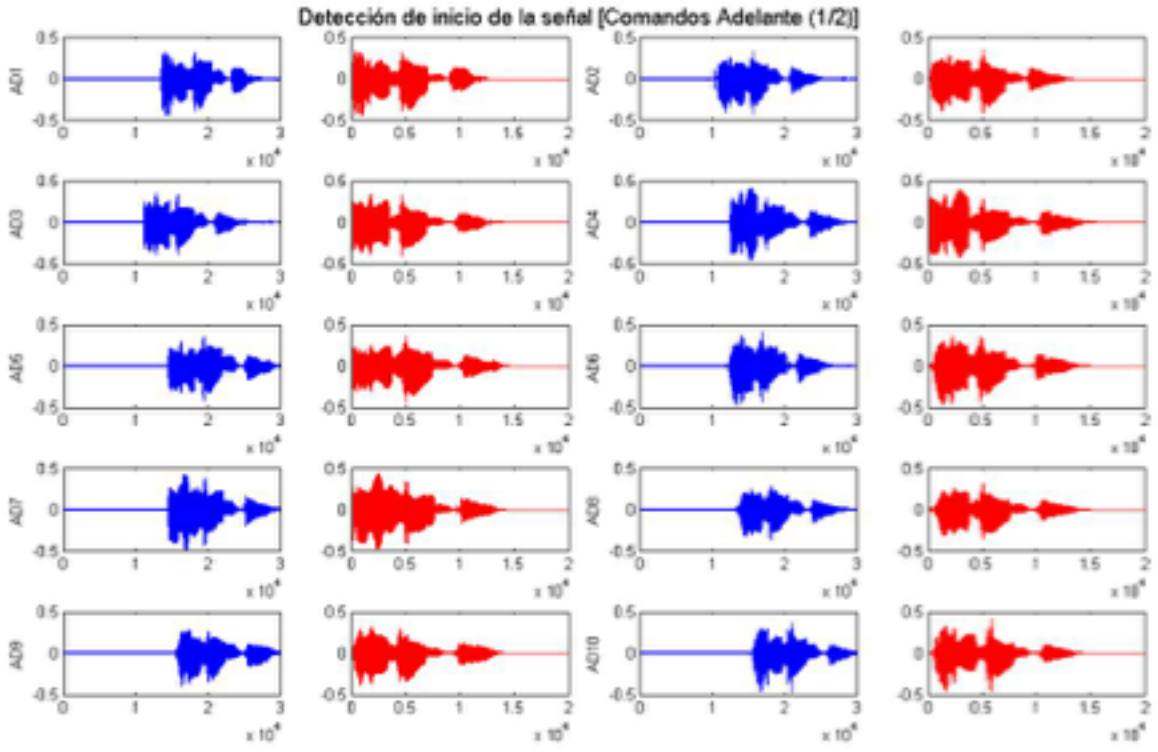

Figura 12: Detección de inicio de la señal en comandos “Adelante” (1/2).

Desplazamiento y espectrograma

Una vez se conoce el punto de inicio de la información relevante de la señal, se puede calcular el. En la figura 13 se compara la señal representada en el dominio del tiempo con su respectivo espectrograma. Allí se puede observar con mayor claridad la distribución de frecuencia de la señal en el tiempo.

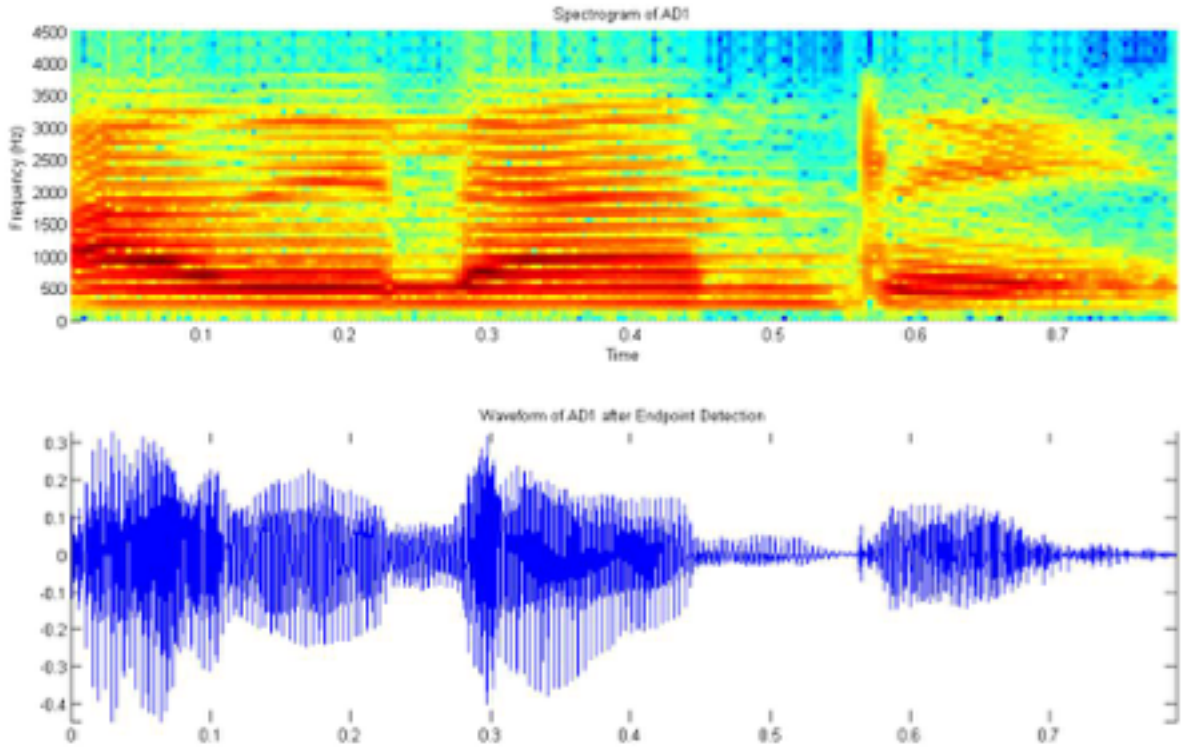

Figura13: Espectrograma de la señal "Adelante" y su representación en el dominio del tiempo.

Sin embargo, estos espectrogramas contienen 101 tramos de tiempo con alrededor de 300 frecuencias cada uno, es decir aproximadamente 30.300 puntos como entrada para la red neuronal. Para la selección de los puntos que servirán como entrada a la red, se procede a calcular los coeficientes cepstrales.

Coeficientes Cepstrales en las Frecuencias Mel:

Para calcular estos coeficientes, se utilizó la función *mfcc* del *Auditory Toolbox* (Slaney, 1998). La cantidad de coeficientes que devuelve, es un parámetro que se puede ajustar en la función. A mayor número de coeficientes, la red será más sensitiva a pequeñas variaciones en la señal. Un menor número de coeficientes resulta en una aproximación menos exacta, pero en este caso, más efectivo. La red neuronal necesita de 10 a 20 coeficientes para un óptimo funcionamiento.

Este banco de filtros está construido con 13 filtros espaciados linealmente, seguidos de 27 filtros espaciados logarítmicamente. Estos 40 filtros tienen la respuesta en frecuencia mostrada en 14:

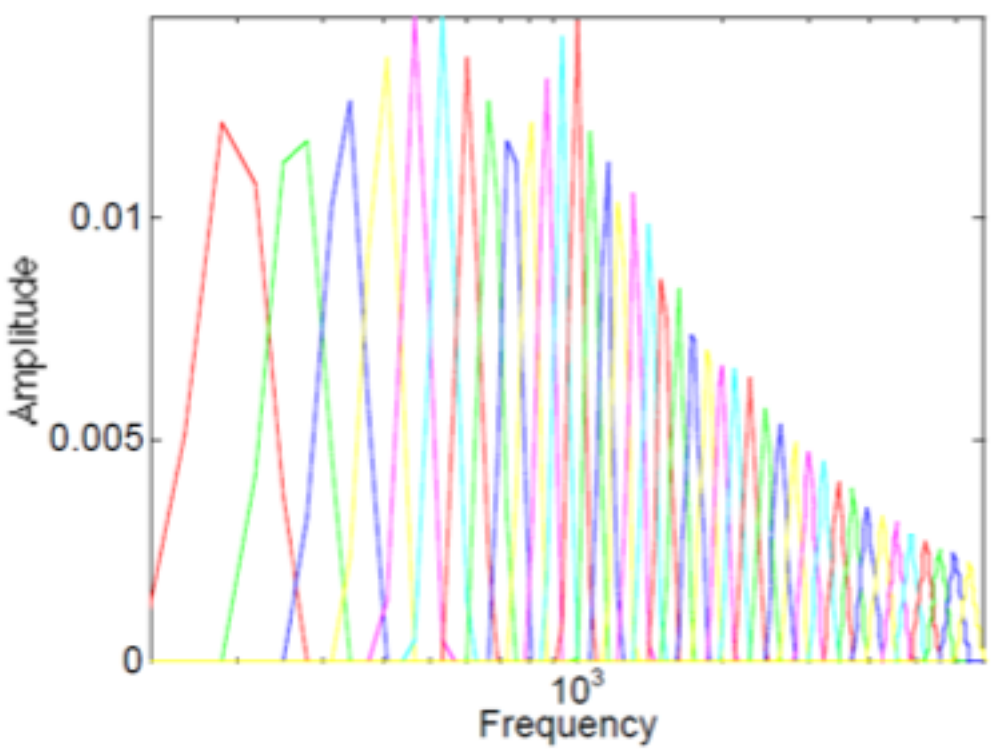

Figura14: Respuesta en frecuencia del banco de filtros para coeficientes cepstrales de Mel.

En la figura 15 se puede observar el resultado de la extracción de características para el primer comando *Adelante*:

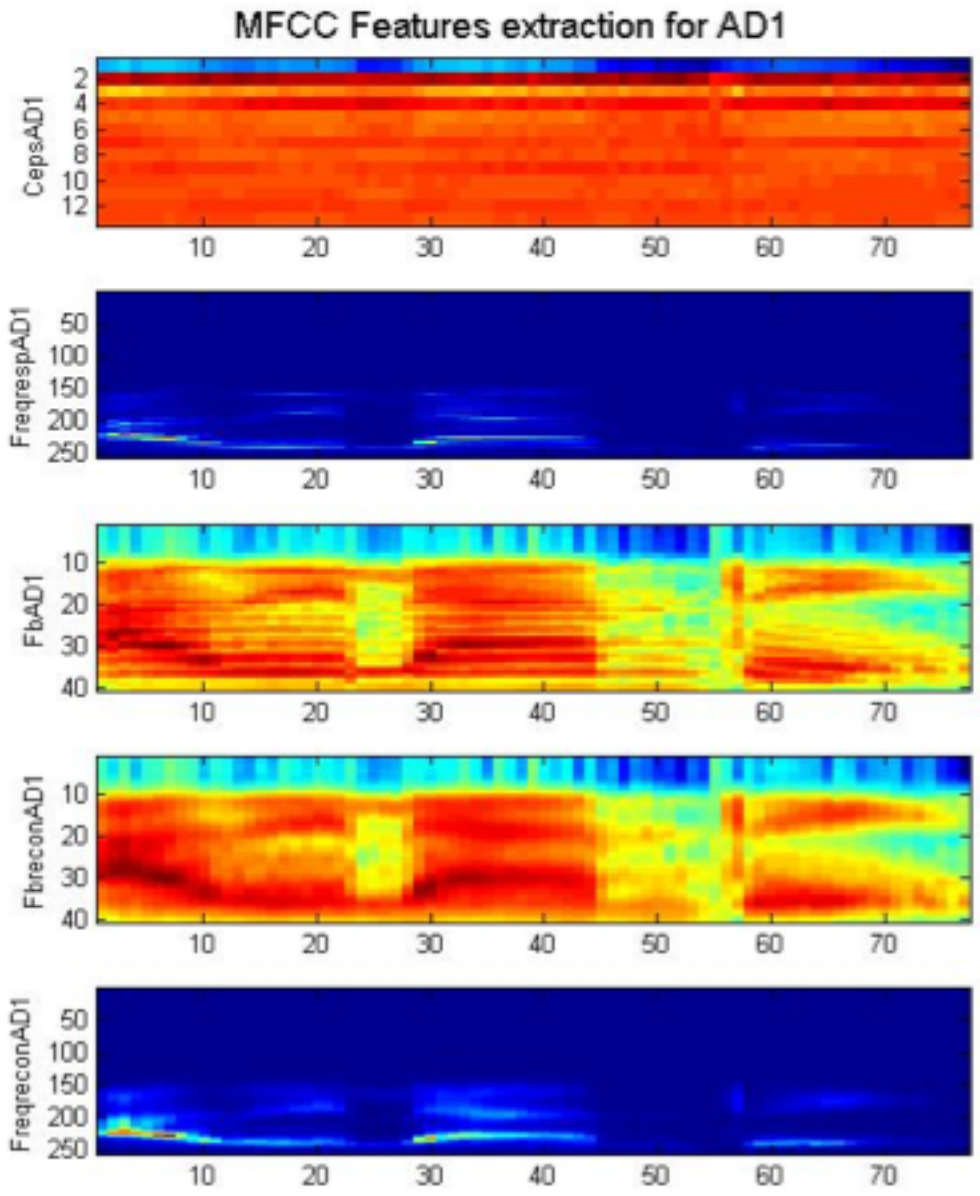

Figura 15: Resultado de la extracción de características del primer comando "Adelante".
En la gráfica FreqreconAD1 se pueden observar los coeficientes cepstrales extraídos.

B. Red neuronal

Prueba de la red

Para evaluar el funcionamiento de la red creada, se necesitan datos diferentes a los usados para la etapa de entrenamiento. En este caso, se tienen 20 señales para la evaluación de cada comando (80 en total). Estas señales deben también pasar por una etapa de pre-procesamiento con el fin de reducir el ruido, detectar el punto de origen de la información relevante y extraer los coeficientes cepstrales. Con el comando de voz Izquierda #17 en la muestra, se evalúa la red neuronal. El resultado se puede ver en la figura 16.

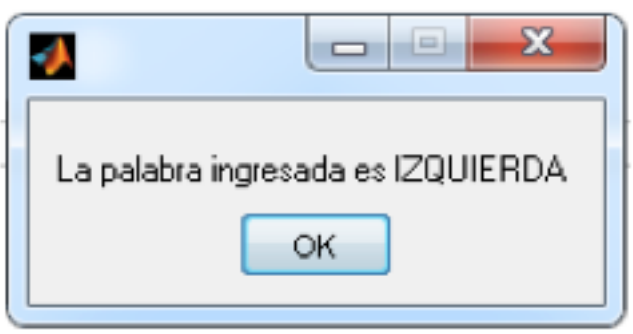

Figura 16: Identificación de un comando "Izquierda" aleatorio.

Sólo cambiando la señal en la entrada, es decir, la muestra de voz del algoritmo, se pudo

evaluar el resto de los 80 comandos. Los resultados obtenidos se presentan en la tabla 2.

Tabla 2: Resultados de las pruebas de identificación de los comandos de voz tomados posterior al entrenamiento.

Comandos	**Aciertos**	**Fallos**
Adelante	17	3
Atrás	16	4
Izquierda	18	2
Derecha	19	1
Total	70	10
87.5 % de aciertos.		

En (Camargo Serrano, 2010) se explica que para la cantidad de muestras por comando utilizada en estas pruebas (20 muestras), una taza de aproximadamente 80% se considera aceptable, y es claro que el algoritmo propuesto la supera significativamente (87.5%). Con lo anterior se puede validar el algoritmo propuesto en este artículo.

5. Conclusiones

Se diseñó una base de datos con comandos: “adelante”, “atrás”, “izquierda”, “derecha” en condiciones normales, para entrenar y evaluar la red, con el fin de controlar el desplazamiento de un equipo de asistencia motora para personas en condiciones de cuadriplejía.

La extracción de características relevantes de las señales es determinante en la efectividad del algoritmo. En la etapa de pre-procesamiento se evaluó la opción de preparar las señales por medio de análisis rápido de Fourier, pero el desempeño no fue el mejor. En el dominio de la frecuencia, todas las señales eran muy similares y las entradas de la red neuronal demasiadas, lo que haría impráctica su implementación. Por ello, se optó por la extracción de coeficientes cepstrales, obteniendo resultados satisfactorios.

Las redes neuronales se han mostrado una vez más como un método útil y fácil de implementar en la identificación de patrones. Un 87.5% de aciertos en la identificación de los comandos de voz es una estadística aceptable, sin embargo en futuras investigaciones se puede evaluar el sistema con diferentes tipos de redes en diferentes condiciones, para así evaluar cuál es la más eficiente dependiendo de la aplicación y mejorar la tasa de errores. También se propondrá la implementación en tiempo real.

Como trabajo futuro, se creará una base de datos mayor, en condiciones normales, controladas y con ruido, y con voces de diferentes personas, con el fin de construir una red más robusta y que pueda identificar comandos de cualquier persona en tiempo real. Así mismo, se implementará al sistema sensores de obstáculos, teniendo en cuenta que los pacientes en condiciones de cuadriplejía pueden no percatarse de muros y otros impedimentos para movilizarse.

Referencias

Camargo Serrano, J. (2010). *Sistema de reconocimiento de voz humana por hardware*. Universidad Pontificia Bolivariana, Bucaramanga.

Del Brío, B. M., & Sanz Molina, A. (2007). *Redes neuronales y sistemas borrosos* (Tercera).

México: Alfaomega.

Departamento Administrativo Nacional de Estadística. (2004). Información Estadística de la Discapacidad. Recuperado a partir de http://www.dane.gov.co/files/investigaciones/discapacidad/inform_estad.pdf

Filtering for Vowels. (s. f.). Recuperado 13 de septiembre de 2016, a partir de http://sail.usc.edu/~lgoldste/General_Phonetics/Source_Filter/SFb.html

Gevaert, W., Tsenov, G., & Mladenov, V. (2010). Neural networks used for speech recognition. *Journal of Automatic Control, University of Belgrade*, 7.

Komiya, K., Morita, K., Kagekawa, K., & Kurosu, K. (2000). Guidance of a wheelchair by voice. En *Industrial Electronics Society, 2000. IECON 2000. 26th Annual Confjerence of the IEEE* (Vol. 1, pp. 102-107). IEEE.

Lai, E. (2003). *Practical digital signal processing, for engineers and technicians* (1ra Edición). Great Britain: IDC Technologies. Recuperado a partir de http://read.pudn.com/downloads142/ebook/619751/practical_digital_signal_processing.pdf

Latinus, M., & Belin, P. (2011). Human voice perception. *Current Biology*, *21*(4), R143–R145. http://doi.org/10.1016/j.cub.2010.12.033

Lyons, J. (s. f.). Mel Frequency Cepstral Coefficient (MFCC) tutorial [Practical Cryptography]. Recuperado 24 de junio de 2015, a partir de http://practicalcryptography.com/miscellaneous/machine-learning/guide-mel-frequency-cepstral-coefficients-mfccs/

M. Faúndez-Zanuy, M. Nilsson, & W. Bastiaan Kleijn. (2002). On the relevance of bandwidth extension for speaker identification. *Signal Processing Conference, 2002 11th European*, 1-4.

Neural Network Toolbox - MATLAB - MathWorks España. (s. f.). Recuperado 5 de agosto de 2014, a partir de http://www.mathworks.es/products/neural-network/index.html?s_tid=gn_loc_drop

Organización Mundial de la Salud, & Banco Mundial. (2011). Informe Mundial sobre la Discapacidad: Resumen. Recuperado 13 de septiembre de 2016, a partir de http://www.who.int/disabilities/world_report/2011/summary_es.pdf

Proakis, J. G., & Manolakis, D. G. (2007). *Digital signal processing*. Pearson Prentice Hall.

Rosenberg, A. (1971). Effect of glottal pulse shape on the quality of natural vowels. *The Journal of the Acoustical Society of America*. Retrieved from http://www.ncbi.nlm.nih.gov/pubmed/5541751

Saha, G., Chakroborty, S., & Senapati, S. (s. f.). *A New Silence Removal and Endpoint Detection Algorithm for Speech and Speaker Recognition applications*. Indian Institute of Technology, India.

Slaney, M. (1998). Auditory Toolbox: A MATLAB Toolbox for Auditory Modeling Work. Recuperado 12 de julio de 2015, a partir de https://engineering.purdue.edu/~malcolm/interval/1998-010/

Smith, M., Saunders, R., Stuckhardt, L., McGinnis, J. M., America, C. L. H. C. S., & Medicine, I. (2013). *Best Care at Lower Cost: The Path to Continuously Learning Health Care in America*. National Academies Press. Recuperado a partir de https://books.google.de/books?id=_wUw6XCFqGwC

Smith, S. W. (s. f.). *The Scientist and Engineer's Guide to Digital Signal Procesing* (Second Edition). San Diego, California: California Technical Publishing. Recuperado a partir de www.DSPguide.com

Tahir, M., & Ashfaque. (2009). Voice Controlled Wheelchair Using DSK TMS320C6711. *Signal Acquisition and Processing, 2009. ICSAP 2009. International Conference on*, 217-220. http://doi.org/10.1109/ICSAP.2009.48

Tiwari, G., Pandey, M., & Shrestha, M. (2011). *Text-prompted remote speaker authentication*. Tribhuvan University, Nepal. Recuperado a partir de https://www.scribd.com/doc/145261803/TEXT-PROMPTED-REMOTE-SPEAKER-AUTHENTICATION-Project-Report-GANESH-TIWARI-IOE-TU#

Waibel, A., & Lee, K.-F. (1990). *Readings in speech recognition*. Morgan Kaufmann Publishers.

Wiener, N. (1949). Extrapolation, interpolation, and smoothing of stationary time series. Retrieved from http://www.ulb.tu-darmstadt.de/tocs/129776289.pdf

World Health Organization, & International Spinal Cord Society. (2013). *International perspectives on spinal cord injury*. World Health Organization.

---

1. Email: jgomez16@cuc.edu.co
2. Email: jsimanca3@cuc.edu.co
3. Email: macosta10@cuc.edu.co
4. Email: fmelende1@cuc.edu.co
5. Email: jvelez@cuc.edu.co

Revista ESPACIOS. ISSN 0798 1015
Vol. 38 (Nº 17) Año 2017

[Índice]

[En caso de encontrar algún error en este website favor enviar email a webmaster]