Reconocimiento de Palabras Aisladas Utilizando LPC Y DTW, para control de navegación de un mini-robot

H. Borrero *Member IEEE*, Y. Baquero, Z. Alezones

***Resumen*—La utilización de herramientas computacionales, hardware y software con capacidades de desarrollo en torno al manejo de modelos bioinspirados generó la construcción de un minirobot móvil aplicando la caracterización de comandos de voz y el aprovechamiento de las capacidades de procesamiento paralelo ofrecidas por el gene digital para gobernar la correspondiente navegación. A nivel general en la etapa de software se implementó un prototipo de programa en el cual se presenta el reconocimiento de palabras aisladas en castellano emitidas por un locutor, aplicado al control de navegación de un mini-robot. La aplicación cuenta con un desarrollo adelantado con el lenguaje java y consta de cuatro módulos: obtención de la señal hablada, extracción de características, comparación de características, procesamiento de los comandos caracterizados por medio de un gene digital y comunicación de las acciones de control a los actuadores del mini-robot.**

***Palabras clave*— Codificación lineal predictiva (Linear Predictive Coding - LPC), alineación temporal dinámica (Dinamic Time Warping - DTW), robot movil, descriptores.**

I. INTRODUCCIÓN

REproducir las capacidades de los seres vivos para aplicarlas y emularlas en las ciencias de la computación es uno de los grandes retos para investigadores en el tema, como resultado de 50 años de estudios acerca del procesamiento de voz se han desarrollado técnicas que a pesar de no poder competir con la genialidad de los sistemas vivos, son aplicables y útiles para problemas puntuales como la navegación de un mini-robot.

Se agradece de manera especial el apoyo financiero aprobado el instituto de investigaciones de la Orinoquia colombiana en el marco del proyecto de investigación robot móvil prototipo como herramienta de apoyo a la enseñanza de las figuras geométricas planas en niños de educación básica preescolar.

Y. Baquero, Z. Alezones reciben su formación de ingenieros de sistemas, actualmente cursan decimo semestre en la Universidad de los Llanos en Villavicencio meta – Colombia. (yebaquero@hotmail.com, z.u.l.e.i.k.a@hotmail.com).

H. Borrero imparte docencia en la Facultad de Ciencias Básicas e Ingeniería de la Universidad de los Llanos, Km 12 vía Puerto López sede Barcelona Villavicencio – Colombia (h_borrelo@ieee.org).

En el proceso de desarrollo de un software reconocedor de voz se deben tener en cuenta los siguientes aspectos: La frecuencia de muestro de la señal de voz en virtud al criterio de Nyquist [1]; el método de extracción de las características que describen la señal de voz (descriptores); el método de comparación de patrones que puede ser dependiente o independe del aspecto anteriormente mencionado.

Básicamente se implementó un prototipo de programa que por medio de su utilización permite realizar un reconocimiento de palabras aisladas en lengua castellana por parte de un locutor, aplicado al control de navegación de un mini-robot móvil. La aplicación cuenta con un desarrollo adelantado en el lenguaje java y consta de cinco módulos: obtención de la señal hablada, extracción de características utilizando el método de codificación por predicción lineal (LPC (Linear predictive coding)), comparación de características con el método de alineación temporal dinámica (DTW (Dinamyc Time Warping)), procesamiento de los comandos caracterizados por medio de un gene digital y comunicación de las acciones de control a los actuadores del mini-robot.

El documento que se presenta pretende exponer los avances logrados en el marco de un proyecto de investigación oficialmente en ejecución por parte del grupo de investigación en ciencias de la computación de la Universidad de los Llanos.

En la sección II se presentan los aspectos generales teóricos pertinentes a cada una de las etapas en el reconocimiento computacional de comandos de voz y el gene digital. En la sección III se exponen las generalidades sobre la implementación y puesta en marcha.

II. ASPECTOS GENERALES

En el contexto del reconocimiento de comandos de voz es necesario captar la señal que corresponde y realizar una serie de etapas para adecuar la señal, hacerla apta para ser interpretada y así generar acciones de control correspondientes.

A. *Pre-procesamiento*

Al iniciar el análisis de una señal de voz es conveniente tener

en cuenta que esta presenta por naturaleza una atenuación en las frecuencias altas, es por esto que se debe realizar un filtro que permita obtener información suficiente de esas frecuencias para no concentrarse únicamente con la información de las frecuencias bajas, además hay que tener en cuenta que el oído humano es más sensible a frecuencias en la zona de los 3000Hz. Por esta razón se aplica un filtro preénfasis, que además ayuda a que el procesamiento de la señal sea manos susceptible a truncamientos, la ecuación del filtro es la que se muestra en la ecuación 1.

$$y[n] = x[n] - \alpha[n-1] \tag{1}$$

Donde x[n] es el vector de amplitud de la señal de voz de entrada, y[n] es la señal de salida del filtro preénfasis, si $\alpha <$ 0, tenemos un filtro de paso bajo y si $\alpha > 0$ es un filtro de paso alto, para nuestro fin se utiliza α=*0.97*.

B. Extracción de características (descriptores)

Teniendo claro que la unidad básica del habla es la que se pretende reconocer (fonemas, vocales, silabas, palabras, frases, etc.) [1], Se cuenta con un conjunto de grabaciones realizadas en un ambiente adecuado (muestras) y después de aplicar a estas muestras de voz el pre-procesado, es en este punto donde se procede a aplicar algún método de extracción de características sobre la señal de voz.

Debido a que las ondas de voz contienen numerosas variaciones (suma de distintas frecuencias), el proceso de extracción de características se realiza a intervalos cortos de tiempo, comúnmente los estudios de voz se realizan a intervalos entre 20ms y 30ms que es donde se considera que la onda de voz no presenta demasiados cambios, a este proceso se le conoce como ventaneo de la señal [1], y se ilustra en la figura 1.

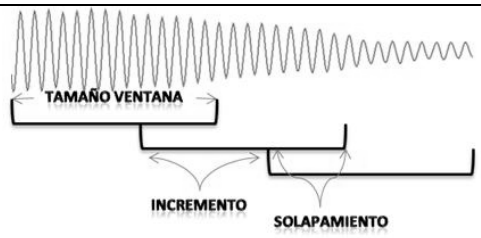

Fig. 1. Ventaneo de una señal.

La elección del tamaño del incremento de las ventanas se realiza teniendo en cuenta el tamaño de la ventana, en el proceso de reconocimiento se deben realizar pruebas con estos parámetros para ajustarlos de manera adecuada al reconocedor, para este caso en el que se utiliza una frecuencia de muestreo de 22050Hz , se realiza un análisis a intervalos de 30ms (lo que equivale aproximadamente a 661 muestras por ventana) con un incremento del 50% (el solapamiento es necesario para no dejar de analizar los bordes de la ventana), aunque el ventaneo hace que el procesamiento sea más lento, mejora la calidad de los resultados, porque el estudio de la señal se hace tomando en cuenta la evolución de las características en el tiempo [1].

Una vez elegido el tamaño de ventana, a cada una se le asigna una función, con el fin de disminuir la importancia de los valores que se encuentran a los extremos de las ventanas, para evitar que características de estos valores varíen la interpretación de la parte central del bloque que es la más significativa. Para este fin los tipos de funciones ventana más utilizados son el tipo Hann y el tipo Hamming [1], en las ecuaciones 2 y 3 respectivamente se puede apreciar los valores y rangos característicos para cada tipo de ventana.

$$\mathrm{V(n)} = \begin{cases} 0.5 - 0.5 \times \mathrm{Cos}\left(\frac{2\pi \mathrm{n}}{\mathrm{N}}\right) & \text{si } 0 \le \mathrm{n} \le \mathrm{N} \\ 0 & \text{en otro caso} \end{cases} \tag{2}$$

$$\mathrm{V(n)} = \begin{cases} 0.54 - 0.46 \times \mathrm{Cos}\left(\frac{2\pi \mathrm{n}}{\mathrm{N}}\right) & \text{si } 0 \le \mathrm{n} \le \mathrm{N} \\ 0 & \text{en otro caso} \end{cases} \tag{3}$$

Después de realizado el correspondiente ventaneo, Los segmentos de voz ya son aptos para la aplicación de técnicas de extracción de patrones como es el caso de LPC (codificación por predicción lineal), el cual está basado en la producción del habla. Se utiliza este modelo debido a que proporciona un modelo adecuado de la señal de voz; sus parámetros se ajustan a las características del tracto vocal; representa la envolvente espectral de la señal de forma comprimida; los parámetros obtenidos mediante predicción lineal muestran un espectro suavizado que proporciona la información más representativa de la voz y es un método preciso, adecuado para computación, tanto por su sencillez como su rapidez de ejecución.

El concepto básico de predicción lineal (LPC) se centra en que una muestra de una señal de voz *x*(n) puede ser predicha por las *k* muestras anteriores de la misma señal, generando una señal aproximada $\tilde{x}(n)$, representada por medio de la ecuación 4.

$$\tilde{x}(n) = \sum_{i=1}^{k} a_i * x(n-i) \tag{4}$$

Se tiene como definición del error de predicción, lo representado en la ecuación 5.

$$e(n) = x(n) - \tilde{x}(n) \tag{5}$$

Para hallar los coeficientes a_i de la ecuación (4) minimizando el error, se aplican mínimos cuadrados al intervalo de N muestras que se deseen considerar, como puede apreciar en el proceso realizado en las ecuaciones 6,7 y 8.

$$L = \sum_{n} e^2(n) = \tag{6}$$

$$\sum_n [x(n) - \tilde{x}(n)]^2 = \tag{7}$$

$$\sum_n \left[x(n) - \sum_{i=1}^{k} a_i * x(n-i) \right]^2 \tag{8}$$

Para obtener el valor mínimo de *L,* se deriva parcialmente la expresión 8 respecto a cada una de las variables a_j 1< j <k y se iguala a cero

$$\frac{dL}{da_j} = \frac{d\sum_n \left[x(n) - \sum_{i=1}^{k} a_i * x(n-i)\right]^2}{da_j} = 0 \tag{9}$$

$$\frac{dL}{da_j} = 2\sum_n \left(x(n) - \sum_{i=1}^{k} a_i * x(n-i) \right) * \left(0 - x(n-j)\right) = 0 \tag{10}$$

$$\frac{dL}{da_j} = \sum_n \left(x(n) - \sum_{i=1}^{k} a_i * x(n-i) \right) * \left(x(n-j)\right) = 0 \tag{11}$$

para $1 \leq j \leq k$

Desarrollando la ecuación 11:

$$\sum_n x(n-j) * x(n) - \sum_{i=1}^{k} a_i * \sum_n x(n-j) * x(n-i) \tag{12}$$

Si se define

$$C_{ij} = \sum_n x(n-j) * x(n-i) \tag{13}$$

Reemplazando en (12) se obtiene:

$$C_{j0} - \sum_{i=1}^{k} a_i * C_{ji} \tag{14}$$

Método de autocorrelación: Se puede observar en (14) la correlación existente en i-j Con lo que $C_{ij} = C_{ji} = r_{|i-j|}$, donde los $r_{|i-j|}$ son los coeficientes de correlación, de esta manera la expresión (13) se puede expresar:

$$\sum_n x(n-j) * x(n-i) = \sum_n x(n) * x(n + |i-j|) = r_{|i-j|} \tag{15}$$

Reemplazando $r_{[i-j]}$ en la ecuación (14) obtenemos:

$$\sum_{i=1}^{k} r(|j-i|) * a_i = r(j), \quad 1 \leq j \leq k \tag{16}$$

La ecuación (16) puede ser expresada en forma matricial:

$$\begin{bmatrix} r_n(0) & r_n(1) & r_n(2) & \dots & r_n(k-1) \\ r_n(1) & r_n(0) & r_n(1) & \dots & r_n(k-2) \\ r_n(2) & r_n(1) & r_n(0) & \cdots & r_n(k-3) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ r_n(k-1) & r_n(k-2) & r_n(k-3) & \cdots & r_n(0) \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ \vdots \\ a_k \end{bmatrix} = \begin{bmatrix} r_n(1) \\ r_n(2) \\ r_n(3) \\ \vdots \\ r_n(k) \end{bmatrix} \tag{17}$$

Esta matriz (17) de autocorrelación puede ser resuelta aplicando el algoritmo de Levinson-Durbin estudiado en [6] u otro método de algebra lineal. Con los coeficientes obtenidos del desarrollo de esta matriz, obtenemos los descriptores LPC de cada ventana aplicada, que en conjunto nos darán el grupo de vectores descriptores de la señal de voz.

C. Reconocimiento

Para la etapa de reconocimiento se tiene establecido el vocabulario del sistema (adelante, atrás, izquierda, derecha) y con él los parámetros LPC. La fase de reconocimiento se inicia con la palabra pronunciada por el locutor la cual es parametrizada del mismo modo y el correspondiente patrón LPC es comparado con los patrones de referencia previamente almacenados en memoria usando una medida de similitud (Hamming, euclidiana, distancia máxima); La medida de distancia entre los parámetros usados es la euclidiana (18) estudiada en [2]:
Parámetros $P = (p_1, p_2, \dots, p_n)$ y $Q = (q_1, q_2, \dots, q_n)$

$$d = \sqrt{(p_1 - q_1)^2 + (p_2 - q_2)^2 + \cdots + (p_n - q_n)^2}$$

$$d = \sqrt{\sum_{i=1}^{n} (p_i - q_i)^2} \tag{18}$$

Debido a la variabilidad intralocutor de la señal, hay diferencias no lineales en la duración de los sonidos y la velocidad de pronunciación de los mismos, incluso tratándose de la misma palabra. Por tanto, se realiza un alineamiento temporal de los patrones (los recién calculados y los almacenados en memoria correspondientes a cada palabra del vocabulario), el cual consiste en minimizar la distancia total entre los estos.

Para realizar el alineamiento se utiliza el método de DTW (Dynamic Time Warping) estudiado en [3, 4]. El cual se basa en determinar el patrón más similar a la palabra pronunciada, es decir, el que proporciona una menor distancia (distancia euclidiana) en la etapa de comparación. De manera más explícita, la comparación se realiza a cada palabra del vocabulario generando un plano en el cual un eje se conforma de los parámetros calculados y el otro de los parámetros almacenados, en donde, cada punto o intersección en el plano es la distancia euclidiana calculada.

Teniendo como finalidad encontrar la ruta mínima D desde el origen hasta la última intersección de ambos ejes, mediante la suma de las distancias de la diagonal en el plano.

$$D = \sum d \tag{19}$$

Para lograr mejores resultados y evitar que la ruta siga en forma vertical o diagonal, se realiza un límite diagonal o radio, de manera que los valores seleccionados en el cálculo de las

distancias e implementación del algoritmo están más cerca de la diagonal, como se muestra en la figura 2.

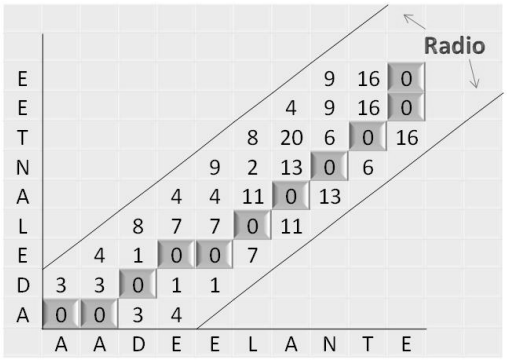

Fig. 2. Representación espacial en el plano generado por las características de la palabra adelante.

D. ADN y Gene digital

El ADN está constituido por moléculas denominadas nucleótidos, cada nucleótido está compuesto de un fosfato, un azúcar de cinco carbonos (desoxirribosa) y una base nitrogenada [5, 6]. Los nucleótidos derivan su nombre de la base que poseen las cuales se clasifican en dos pirimidinas {Citosina (C), Timina (T)} y dos purinas {Adenina (A), Guanina (G)}. En la figura 1(a) se representan los cuatro nucleótidos constituyentes del ADN cada fosfato (cruz) se enlaza con la desoxirribosa (óvalo) y el grupo fosfato – azúcar se enlaza a una base nitrogenada

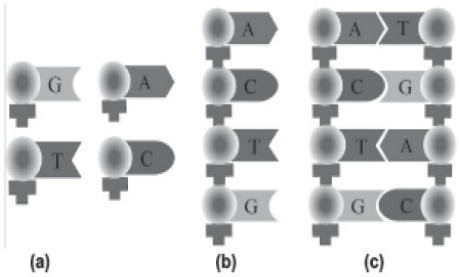

Fig. 3. Representación básica del ADN. (a) los nucleótidos, (b) cadena sencilla de ADN, (c) doble hélice de ADN.

Una cadena de ADN está conformada por una secuencia de nucleótidos como se representa en la figura 1 (b). Existe un principio natural denominado complemento Watson – Crick en el cual dos nucleótidos son complementarios si sus bases son complementarias, la Adenina complementa la Timina y la Citosina complementa la Guanina, el complemento corresponde a la unión por enlaces de hidrógeno. Es importante tener en cuenta que para el caso de las cadenas sencillas de ADN (ver figura 2(b)) se cuenta con 4 elementos de base de modo que si se podría en principio que si dispone de *n* bases disponibles para formar una sola cadena, es posible que se forme alguna de 4^n posibles combinaciones.

El ADN es la molécula de la vida, es una molécula compleja que además de almacenar información genética, juega un papel activo en los organismos. Los genes son segmentos de ADN que codifican una o más proteínas, en la figura 4 se muestra un diagrama de bloques de las regiones constituyentes de un gene.

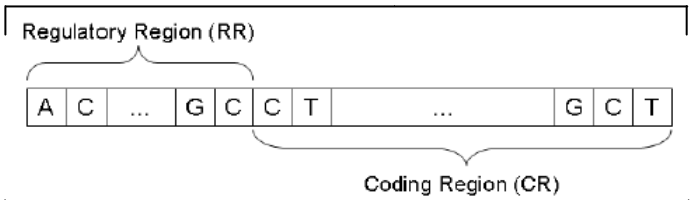

Fig. 4. Representación general de un gen

La región reguladora (RR) es donde las proteínas y otras moléculas se fijan para iniciar o parar la expresión de los genes, l segunda región corresponde a la región codificadora (CR) que es el segmento de DNA que se transcribe en RNA [5, 6].

En esencia, los genes son segmentos de ADN transcritos en ARN. Su composición se basa en cadenas de tres moléculas conocidas como nucleótidos. Desde el punto de vista de la emulación electrónica, encontramos varias posibilidades de programación e implementación del gene biológico, conocido como gene digital [7].

Basados en los aspectos mencionados, se plantea el gene digital como una configuración de registros de forma tal que se logre asociar una acción a un comportamiento determinado en la entrada del sistema, ver figura 5. El robot incorpora en un solo registro binario, las lecturas de todos sus sensores (detectores, sensores digitales y análogos). Con el registro de sensores u se obtiene, de manera paralela, el registro de la acción requerida y [8].

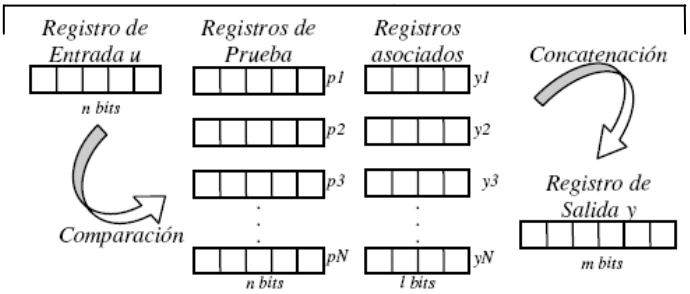

Fig. 5. Diagrama de registros en el gene digital.

El gene digital está compuesto por cuatro secciones: (i) un registro de entrada (u); (ii) una serie de registros de prueba (pi); (iii) registros asociados a los anteriores (yi); (iv) un registro de salida (y). La operación consiste en realizar comparaciones, en paralelo, entre el registro de entrada y cada uno de los registros de prueba. A partir de esta comparación y según alguna restricción, el registro asociado correspondiente, es o no concatenado al registro de salida. Como método para realizar la comparación, se propone el uso de la distancia de Hamming, en la que se mide el número de bits diferentes entre

los registros. Para determinar la condición de concatenación, definimos un parámetro llamado umbral de Hamming, si la distancia es menor al umbral, se concatenará el registro asociado correspondiente.

En la figura 6 se muestra un diagrama general sobre la arquitectura en la que se basa el desarrollo del programa prototipo, básicamente se emula el paralelismo disponible en el gene digital, como se aprecia, la palabra de entrada (muestra) que corresponde a una cadena de bits asociada a un comando de voz caracterizado a vector binario es comparada de manera paralela con un cierto número de palabras cargadas en los registros P_{r1} a P_{rN}. Las palabras almacenadas en los registros P_{r1} a P_{rN} corresponden al complemento de cada una de las palabras caracterizadas como comandos de voz. Como se aprecia la muestra se procesa paralelamente con las palabras almacenadas en los registros, se debe detectar la distancia haming entre cada una de estas palabras y la muestra. Si alguna de las palabras almacenadas genera sobrepasamiento de un umbral θ_N habilita el bloque "*Enable*" que le corresponde y por lo tanto la palabra e_N (acción de control) pasa al exón

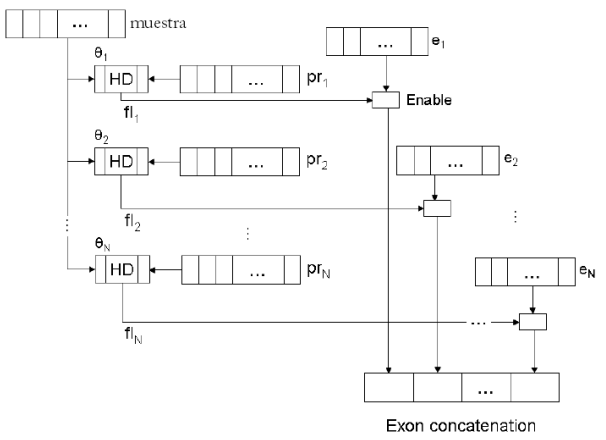

Fig. 6. Diagrama gene digital

Las palabras cargadas en cada uno de los registros e_N corresponde las acciones control para navegación del mini-robot.

E. Cinemática del robot

La locomoción del mini-robot se basa en un modelo trípode de movimiento, igual como sucede con los seres vivos el mini-robot debe ser capaz de soportar su propio peso y superar la fuerza de gravedad. Este modelo trípode básicamente consiste en mantener tres patas en el suelo y darle libertad de movimiento a las demás; una ventaja de este modelo basado en patas es la estabilidad que se genera para el mini-robot y que permite aislar el cuerpo del terreno empleando puntos discretos de soporte. Así mismo, mediante patas, es posible conseguir cierta omnidireccionalidad y el deslizamiento en la locomoción es mucho menor [6].

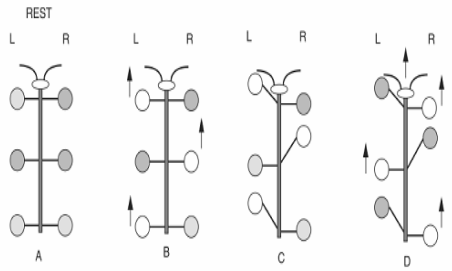

Fig. 7. Representación sobre el modelo trípode para la locomoción del hexápodo. [6].

Como se muestra en la figura 7 (A), la posición inicial para el prototipo será la de mantener todas sus patas en el suelo, seguido de esto (figura 7 (B)) se reafirma la posición fija para tres de las patas las demás avanzan; el siguiente paso se fijar las patas que avanzaron lo que permitirá el avance de las demás. Esta dinámica de movimiento será reiterativa hasta el momento en el que se reinicie el algoritmo que controla la cinemática que realiza el robot. En la figura 8 se una imagen de la estructura mecánica implementada para el funcionamiento del robot móvil.

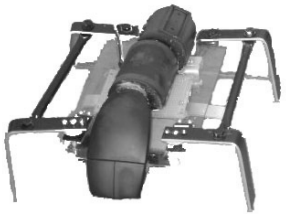

Fig. 8. Prototipo mini-robot.

III. IMPLEMENTACION

La aplicación, que es un prototipo se encuentra desarrollada en lenguaje JAVA, se entreno con las palabras *"adelante", "atrás", "derecha", "izquierda"* de un mismo locutor, cada palabra reconocida se caracterizo en forma de un vector binario que corresponde a la palabra de entrada (muestra) al gene-digital. En cuatro registros (P_{RN}) del gene-digital se encuentran almacenadas las palabras (vectores binarios) que generan un umbral adecuado para activar las acciones de control recidentes en los registros e_n como se puede apreciar en la figura 6.

Las palabras de control que residen en el exón de salida del gene digital (figura 6) se deben trasmitir a las respectivas entradas de los actuadores del mini-robot (motores).

Es importante resaltar que las palabras de control residentes en el exón de salida corresponden con las acciones de desplazamiento del prototipo. De acuerdo con el diagrama de la figura 9, las salidas del gene digital corresponden al exón de salida del gene digital de manera que se habilita cada uno de los bloques que habilita alguna de las dinámicas de desplazamiento que se indica en la figura, así mismo, se puede apreciar que cada uno de los bloques (adelante, atrás, derecha,

izquierda) es habilitado desde el gene digital y estos habilitan el funcionamiento de los actuadores de salida que en principio son tres servos – motores como actuadores de un robot hexápodo (tres grados de libertad).

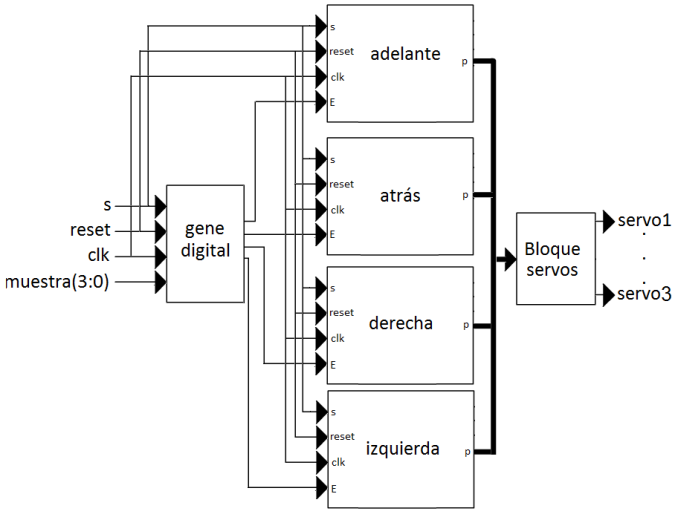

Fig. 9. Diagrama de bloques relacionado con desplazamientos del mini-robot.

Retomando el diagrama de la figura 9 se puede apreciar que los bloques correspondientes a la ejecución de los desplazamientos del mini-robot corresponden en sí a una entidad que tiene como puertos de entrada: el puerto *s* que permite poner en funcionamiento la dinámica del robot; el puerto *reset* simplemente permite reiniciar la dinámica del mir-robot de manara asincrónica; se tiene también el puerto de entrada *muestra* el cual corresponde a las palabras caracterizadas en el bloque de reconocimiento de comandos de voz; se cuenta con 3 puertos de salida por medio de los cuales se transmiten señales de onda cuadrada con ciclo útil controlado necesarias para el posicionamiento de los ejes de 3 servos que hacen parte fundamental de los 3 grados de libertad del sistema.

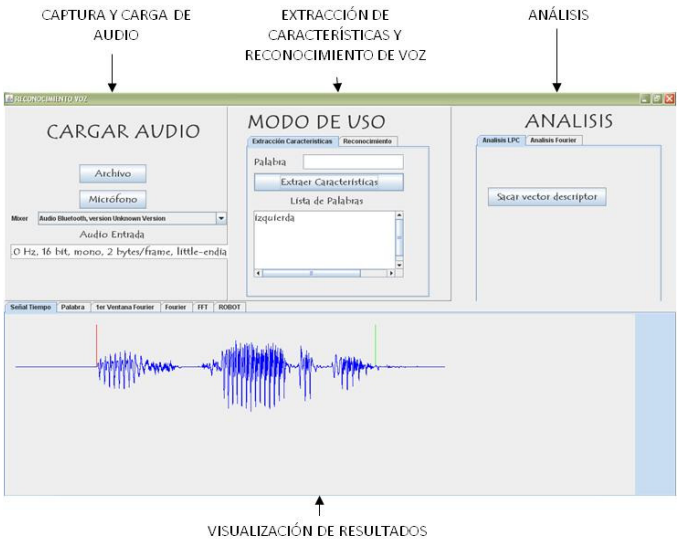

Fig. 10. Interface gráfica, aplicación de Reconocimiento de voz.

En cuanto a la aplicación software, la interfaz gráfica consta de tres componentes, el primero destinado a la captura o carga de audio, el segundo para extracción de características y reconocimiento de voz y el último de análisis. Adicionalmente cuenta con un panel inferior, en el cual se visualizan los resultados de aquellos procesos que lo requieran (figura 10).

Mostrando a nivel interno (figura 11) las etapas en el reconocimiento de voz, con los tipos de datos que se manejan en forma específica para la aplicación:

En la primera parte obtenemos el vector de voz en el tiempo, ya sea desde el modulo del micrófono o desde archivo, esta información es entregada en forma binaria, con lo cual hay que realizar el respectivo cast para poder ser trabajado directamente en las posteriores etapas. En la siguiente etapa realizamos el filtro preénfasis sobre los datos de voz.

Posteriormente se obtienen los descriptores por cada ventana de voz que en conjunto nos darán una matriz de elementos descriptores de la palabra, dependiendo si el siguiente paso es reconocimiento las características son guardadas o no, para ser comparadas.

Por último se realiza una comparación entre las características de las palabras que se tienen guardadas y las que se desean comparar, lo cual nos mostrara quien de estas nos arroja la menor distancia y por consiguiente será la palabra reconocida.

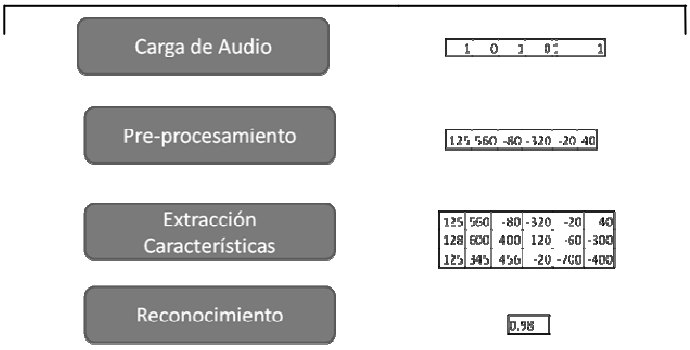

Fig. 11. Salida de la información en cada fase del sistema.

IV. RESULTADOS

La aplicación fue probada en una población de 25 hombres y 25 mujeres (6 menores de 12 años, 11 de 12 a 19 años, 12 de 20 a 35 años y 21 mayores a 35 años); de los cuales se tomaron 5 muestras de cada palabra (*"adelante", "atrás", "derecha"* y *"izquierda"*, adicionando además el reconocimiento de la palabra *"alto"* para un desarrollo posterior de una asociación para respuesta). Teniendo en total 25 muestras por persona (adelante 1-2-3-4-5, atrás 1-2-3-4-5, izquierda 1-2-3-4-5, derecha 1-2-3-4-5 y alto 1-2-3-4-5). Probando el sistema mediante el entrenamiento de las muestras n° 1 inicialmente y recopilando los resultados mediante el reconocimiento de las 4 restantes, posteriormente

se realiza la misma acción de entrenamiento para las muestras 2, 3, 4, 5 y sus respectivas pruebas con las muestras sobrantes.

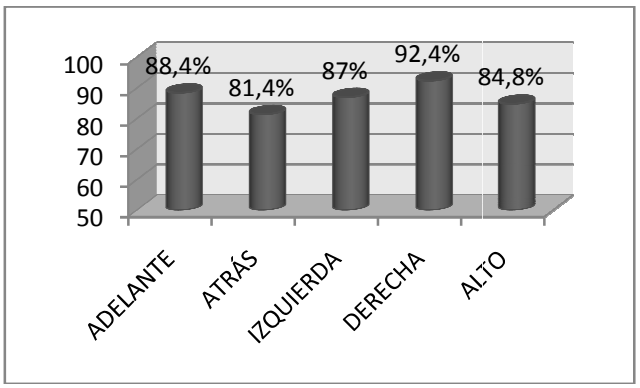

Fig. 12. Porcentaje de reconocimiento para hombres.

Obteniendo de los análisis resultados de reconocimiento de 86,8% para los hombres (figura 12), 82% para las mujeres (figura 13) y un total de 84,45% (figura 14) de reconocimiento teniendo en cuenta la población en general.

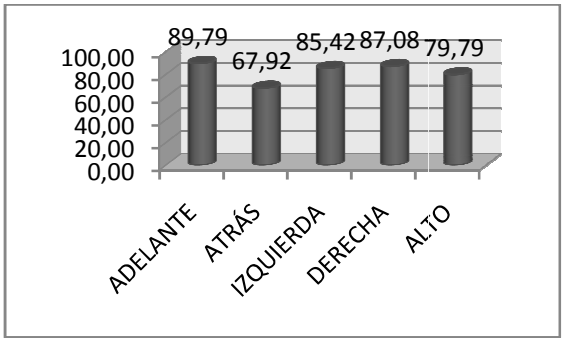

Fig. 13. Porcentaje de reconocimiento para mujeres.

Las muestras fueron tomadas en un ambiente bastante natural, por tanto la efectividad de la aplicación que reside en un 84,45% puede aumentar considerablemente si se tiene en cuenta condiciones lo mas ideales posibles.

Al momento de tomar las muestras, estas fueron tomadas de manera continua, es decir, grabaciones en las cuales el locutor repetía una palabra varias veces y posterior a ello fueron recortadas las palabras para formar conjunto de muestras; lo que genero como resultado posibles errores humanos al momento de manipularlas.

A pesar de las condiciones adversas en que fueron tomadas las muestras y manipuladas, se nota un porcentaje de reconocimiento bastante aceptable para el sistema, teniendo en cuenta además la poca experiencia de los desarrolladores en el tema.

Al momento de desarrollar un reconocedor de voz sin filtros de ruido, hay que tener en cuenta el ambiente al cual se va a aplicar; ya que en este caso el ruido sería necesario, porque al momento de reconocer las muestras presentaran un mayor grado de semejanza.

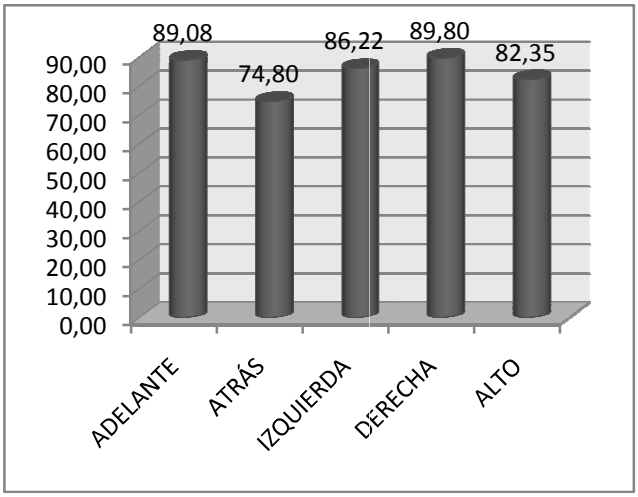

Fig. 14. Porcentaje de reconocimiento total de la aplicacion.

V. CONCLUSIONES

Es necesario analizar las ondas de manera segmentada para comprender su evolución en el tiempo; ya que si se usan tamaños de ventana demasiado grandes se omiten cambios locales; contrario a si se toman tamaños de ventanas demasiado pequeños ya que se reflejan demasiado los cambios puntuales.

El tamaño del incremento entre ventanas influye directamente en los tiempos de repuesta de los algoritmos y a su vez en la calidad de los resultados, con un incremento demasiado pequeño, el tiempo de respuesta es mayor y los resultados poco favorables.

LPC es un método adecuado al tratamiento del habla ya que presenta una aproximación a la producción de la misma.

El uso del algoritmo de programación dinámica (DTW), es ideal para reconocimiento de señales de voz, porque trata de reducir las diferencias temporales naturales del habla.
El algoritmo DTW ofrece buenos resultados para un conjunto pequeño de palabras a reconocer, si se requiere realizar el reconocimiento para un vocabulario extenso, esta solución no es la más óptima computacionalmente.

El entrenamiento con características de uno o pocos hablantes, hace que el reconocimiento de voz sea dependiente del hablante, para un reconocedor de voz general, se deben realizar estudios con una muestra considerable de distintas voces, tratando de analizar características generales.

La primera etapa de realización del reconocimiento de comandos de voz y gene digital es efectuada en Java como prototipo para posteriormente ser implementado en hardware.

Se considera que vale la pena explotar las capacidades del chip ADN emulado electrónicamente, el gene digital y los algoritmos genéticos en la caracterización y reconocimiento de comandos de voz en u sistema embebido aprovechando la disponibilidad de dispositivos lógicos reconfigurables como lo son las FPGA pues cuentan con la posibilidad de procesar la información de comando de manera paralela y asociativa.

Es recomendable a futuro trabajos como la aplicación de filtros para el tratamiento adecuado del ruido en pre-procesamiento, implementación de algoritmos diferentes en los diversos procesos de extracción de características y reconocimiento, no ser dependiente en gran escala del hablante, aplicación a diversos fines, entre muchos más trabajos futuros.

AGRADECIMIENTOS

Los autores agradecen el soporte ofrecido por el instituto de investigaciones de la Orinoquia colombiana, la valiosa colaboración recibida por parte del grupo de ciencias de la computación de la Universidad de los Llanos, al Ing. Juan Fajardo e Ing. Rubén Darío Ángel por sus valiosas orientaciones y de manera especial al grupo CIS (Control Inteligente de Sistemas) Universidad Nacional, por ser pionero en este tipo de proyectos.

REFERENCIAS

[1] Bernal, J., Bobadilla S., Gómez, P., "Reconocimiento de voz y fonética acústica". Ed RA-MA. 2000.

[2] Bregón, A.; Alfonso A., "Un Sistema De Razonamiento Basado En Casos Para La Clasificación De Fallos En Sistemas Dinámicos". Universidad de Valladolid. 2005.

[3] Keogh, J. "Derivative Dynamic Time Warping". 2001.

[4] Alvarado, J. "Reconocimiento De Palabras Aisladas Utilizando MFCC y Dinamic Time Warping". Universidad Nacional De Trujillo. 2008.

[5] Campbell, A.M. y Heyer, L.J.: "Discovering genomics, proteomics, and bioinformatics, Pearson Education", 2003.
Alberts B, Bray D, Johnson A, Lewis J, Raff M, Roberts K, Walter, P ."Essential cell biology." Garland Publishing, 1998.

[6] Prieto, J., Ramos, O. y Delgado, A.: "Diseño de un gene digital en FPGA y MATLAB con aplicaciones en robótica móvil," XIII Taller Iberchip IWS-2007, Lima – Perú, Marzo 14-16, 2007.

[7] Farfan, A., Herreño J. y Delgado, A.: "Gene digital y chip ADN electrónico: aplicaciones en robótica móvil," 3rd Colombian Workshop on Robotics and Automation (CWRA), Universidad Tecnológica de Bolivar, Cartagena, Agosto 21- 22, 2007.