

**IDENTIFICACIÓN DE AVES A PARTIR DE CANTO CARACTERÍSTICO E  
INFORMACIÓN CONTENIDA EN BASES DE DATOS**

**JUAN CARLOS MAYA GONZALES  
ROGER FABIAN LEYTON CALDERON**

**UNIVERSIDAD DE NARIÑO  
FACULTAD DE INGENIERÍA  
PROGRAMA DE INGENIERÍA ELECTRÓNICA  
SAN JUAN DE PASTO  
2011**

**IDENTIFICACIÓN DE AVES A PARTIR DE CANTO CARACTERÍSTICO E  
INFORMACIÓN CONTENIDA EN BASES DE DATOS**

**JUAN CARLOS MAYA GONZALES  
ROGER FABIAN LEYTON CALDERON**

**Trabajo de Grado presentado como requisito para obtener el título de  
Ingeniero Electrónico**

**Asesor: MSC. DARÍO FERNANDO FAJARDO FAJARDO**

**UNIVERSIDAD DE NARIÑO  
FACULTAD DE INGENIERÍA  
PROGRAMA DE INGENIERÍA ELECTRÓNICA  
SAN JUAN DE PASTO  
2011**

NOTA DE RESPONSABILIDAD

“LAS IDEAS Y CONCLUSIONES APORTADAS EN EL TRABAJO DE GRADO, SON DE RESPONSABILIDAD EXCLUSIVA DE LOS AUTORES”

ARTÍCULO 1 DEL ACUERDO No. 324 DE OCTUBRE 11 DE 1966, EMANADO DEL HONORABLE CONSEJO DIRECTIVO DE LA UNIVERSIDAD DE NARIÑO

Nota de aceptación:

---

---

---

---

---

---

---

---

---

---

---

---

Firma del jurado

---

Firma del jurado

## DEDICATORIA

*“A mi familia, ya que siempre han estado ahí,  
en especial a mi madre y hermanos,  
y a mi tío Franco por su apoyo.*

*Roger Leython*

*“A mis padres por su apoyo constante y a mis hermanos  
Por ser los mejores compañeros en este proceso”*

*Juan Carlos Maya*

## RESUMEN

Este trabajo describe el desarrollo de un sistema para la caracterización y reconocimiento más adecuado del canto de ave tomando como referencia una base de datos de 10 aves. Se hace uso de la función PSD (Densidad Espectral de Potencia) como la mejor opción por la marcada diferenciación de los cantos de diferentes especies en pocos datos, así, se implementaron métodos como Redes Neuronales, PCA (Análisis de Componentes Principales) y Correlación, que generaron muy buenos resultados usando cantos de aves de diferentes reservas, tanto de la región como a nivel mundial. Las Redes Neuronales después de su entrenamiento y validación resultan ser las que mejor desempeño presentan en el reconocimiento.

Se desarrolla una aplicación en MATLAB que muestra los resultados de reconocimiento en porcentajes de exactitud después de ingresar un canto determinado, esto apoyado en una base de datos de 10 aves fija para el método de ANN (Redes Neuronales Artificiales) y modificable en PCA y Correlación.

Usando la tarjeta Domino FEZ se desarrolla un sistema embebido que incluye la lectura de un canto de ave a analizar mediante una tarjeta SD, el cálculo de la función PSD y una red neuronal ya entrenada y validada, para finalmente desplegar los resultados en una LCD. Los resultados son muy cercanos a los obtenidos en PC.

## **ABSTRACT**

A more adequate characterization and recognition of the song of the bird is raised on the basis of a data base of ten birds. The function PSD (Power Spectrum Density) is used as the best option for its remarkable difference in the song of different species in a few data; in this order of ideas, methods as Neuronal Networks, PCA (Principal Component Analysis) and Correlation which generated the expected results were implemented, after the use of the songs of the birds from different reserves from the region as worldwide. After training and validation, the Neuronal Networks showed the best management in the recognition.

An application in MATLAB is developed to show the results of recognition in percentages of precision, after inserting a certain song on an already fixed data base of ten birds for the method ANN and modifiable in PCA and Correlation.

By using the Domino FEZ card, an embedded system including the reading of the bird song is developed for analysis through a SD card, the function PSD and an already trained and validated neuronal network are implemented, in order to show the results in a LCD; the results are very close to those obtained in PC.

## CONTENIDO

	Pág.
INTRODUCCIÓN.....	16
1. DESCRIPCIÓN DEL PROBLEMA .....	17
1.1 PLANTEAMIENTO DEL PROBLEMA .....	17
1.2 FORMULACIÓN DEL PROBLEMA.....	17
2. OBJETIVOS .....	18
2.1 OBJETIVO GENERAL .....	18
2.2 OBJETIVOS ESPECÍFICOS .....	18
3. JUSTIFICACIÓN .....	19
4. REVISIÓN SOBRE EL RECONOCIMIENTO BIOACÚSTICO .....	21
4.1 EL RECONOCIMIENTO DE CANTOS VOCÁLICOS.....	21
4.2 TÉRMINOS Y CONDICIONES EN EL RECONOCIMIENTO DE CANTOS VOCÁLICOS .....	23
4.2.1 Agentes externos .....	23
4.2.2 Variación en la composición del Canto .....	23
4.2.3 Número de cantos presentes en una grabación.....	25
4.2.4 Delimitaciones de la implementación .....	25
4.3 METODOS DE RECONOCIMIENTO CANTO CARACTERISTICO.....	26
4.3.1 Dinamyc Time Warping (DTW) .....	26
4.3.2 Caracterización y reconocimiento mediante Densidad Espectral de Potencial (PSD) .....	28
4.3.2.1 Principal Component Analisis (PCA).....	31
4.3.2.2 Coeficiente de Correlación.....	35
4.3.2.3 Redes Neuronales Artificiales .....	36
4.3.2.4 Entrenamiento y uso de Redes Neuronales.....	37
4.4 ALGORITMO DE CORTE .....	38

5.	DESARROLLO DE LA APLICACIÓN.....	42
5.1	SISTEMA RECONOCIMIENTNO MATLAB.....	42
5.1.1	Lectura de archivos WAV o MP3 .....	43
5.1.2	Algoritmo de corte.....	43
5.1.3	Método de Comparación.....	44
5.1.4	Despliegue de Resultados .....	44
5.2	CREACIÓN DE BASE DE DATOS .....	44
5.3	INTERFAZ GRÁFICA DEL SISTEMA.....	46
5.3.1	Identificación de una Especie.....	47
5.3.2	Creación de una nueva Base de Datos.....	51
6.	IMPLEMENTACION DEL SISTEMA EN MEDIOS EMBEBIDOS.....	55
6.1	SISTEMA RECONOCIMIENTO MEDIO ENBEDIDO.....	56
6.1.1	Lectura del directorio en la memoria micro SD .....	57
6.1.2	Filtrado de los archivos WAV y selección de la grabación .....	57
6.1.3	Lectura y verificación del encabezado del archivo seleccionado .....	57
6.1.4	Lectura decodificación y cálculo del PSD de cada bloque .....	59
6.2	SIMULACIÓN EN LA RED NEURONAL .....	63
6.3	DESPLIEGE DE RESULTADOS.....	66
7.	RESULTADOS.....	67
7.1	RECONOCIMIENTO DE 10 ESPECIES DIFERENTES .....	67
7.2	IMPLEMENTACIÓN DE LA FUNCIÓN PSD EN EL SISTEMA EMBEBIDO .....	76
8.	CONCLUSIONES .....	79
8.1	CONCLUSIONES Y APORTES .....	79
8.2	DESARROLLOS FUTUROS .....	80
9.	BIBLIOGRAFÍA .....	81
10.	ANEXOS .....	84

## LISTA DE FIGURAS

	<b>Pág.</b>
Figura 1. Variación entre dos cantos de la misma especie .....	22
Figura 2. Espectrogramas de diferentes dialectos en el agateador norteño .....	24
Figura 3. Variación de trama en longitud y velocidad. ....	27
Figura 4. PSD diferentes aves .....	30
Figura 5. Variacion PSD grabacion mismo individuo. ....	31
Figura 6. Modelo neurona .....	36
Figura 7. Funciones de activación .....	37
Figura 8. Funciones ventana.....	39
Figura 9. Obtención de envolvente de la señal .....	40
Figura 10. Aplicación algoritmo de corte.....	41
Figura 11. Diagrama de flujo aplicación en MATLAB.....	43
Figura 12. Organización de carpetas con especies y respectivos cantos .....	45
Figura 13. Interfaz de usuario principal.....	46
Figura 14. Opciones del menú.....	47
Figura 15. Menú escoger modo de comparación.....	47
Figura 16. Opción cargar base de datos.....	48
Figura 17. Directorio seleccionar base de datos.....	48
Figura 18. Especies en base de datos.....	49
Figura 19. Seleccionar directorio muestrario. ....	49
Figura 20. Ventana escoger track a reconocer. ....	50
Figura 21. Resultados procesamiento track.....	50
Figura 22. Resultados del proceso de identificación y reproducción del canto.....	51
Figura 23. Directorio muestras para nueva base de datos. ....	52
Figura 24. Directorio destino base de datos. ....	52

Figura 25. Directorio cargar base de datos.....	53
Figura 26. Especies encontradas en base de datos. ....	53
Figura 27. Directorio seleccionar muestrario. ....	54
Figura 28. Resultados con nueva base de datos.....	54
Figura 29. Conexión de Pantalla y Memoria al sistema empotrado .....	56
Figura 30. Diagrama de flujo embebido. ....	56
Figura 31. Conversión a 16 bits lectura SD. ....	59
Figura 32. Comparación calculo PSD MATLAB y Sistema Embebido .....	62
Figura 33. Esquema neurona.....	63
Figura 34. Representación de una capa de la red neuronal. ....	64
Figura 35. Representación de una red neuronal de dos capas. ....	65
Figura 36. Comparación reconocimiento ANN, PCA y Correlación. ....	74
Figura 37. Comparación reconocimiento ANN, PCA y Correlación 20 especies. ..	74

## LISTA DE TABLAS

	<b>Pag.</b>
Tabla 1. Formato archivo de audio. ....	57
Tabla 2. Porcentaje de reconocimiento red neuronal.....	68
Tabla 3. Porcentaje de reconocimiento PCA .....	69
Tabla 4. Porcentaje de reconocimiento Correlación .....	70
Tabla 5. Porcentaje reconocimiento Red Neuronal 2.....	71
Tabla 6. Porcentaje reconocimiento PCA 2 .....	72
Tabla 7. Porcentaje reconocimiento Correlación 2 .....	73
Tabla 8. Distorsión MATLAB y Embebido.....	76
Tabla 9. Porcentaje reconocimiento Red Neuronal Tarjeta 1 .....	77
Tabla 10. Porcentaje reconocimiento Red Neuronal Tarjeta 2 .....	78

## LISTA DE ANEXOS

	Pag.
ANEXO A. Eigenvectores y eigenvalores.....	84
ANEXO B. Redes neuronales.....	84

## GLOSARIO

**.NET Framework:** conjunto de nuevas tecnologías, incluidas en los sistemas operativos Microsoft Windows, que provee soluciones pre-codificadas para requerimientos comunes de los programas y que gestiona la ejecución de los mismos.

**Autocorrelación:** se define como la covarianza de la señal consigo misma. La función de autocorrelación resulta de gran utilidad para encontrar patrones repetitivos dentro de una señal

**Análisis espectral:** Se refiere a la acción de descomponer una señal difícil de interpretar en componentes más simples, generalmente componentes de frecuencia.

**ANN:** Son un paradigma de aprendizaje y procesamiento inspirado en el funcionamiento del sistema nervioso del ser humano. La sigla ANN significa *Artificial Neural Network* que en español traduce Red Neuronal Artificial.

**Covarianza:** En estadística la covarianza es una medida de dispersión conjunta de dos variables estadísticas.

**Correlación:** Indica la fuerza y la dirección de una relación lineal entre dos variables aleatorias.

**Desviación estándar:** la desviación típica es una medida (cuadrática) que informa de la media de distancias que tienen los datos respecto de su media aritmética.

**Etología:** Rama de la biología y de la psicología experimental que estudia la conducta, el instinto y las actividades inatas o aprendidas de los animales.

**Fast Fourier Transform (FFT):** Transformada rápida de Fourier es el término utilizado para referirse al cálculo de un análisis espectral mediante un algoritmo ejecutado de manera eficiente.

**Filtro electrónico:** Es un elemento capaz de discriminar determinada frecuencia o gama de frecuencias de una señal eléctrica modificando tanto su amplitud como su fase.

**Periodograma:** Densidad espectral de energía por tiempo muestreado.

**Power Spectral Density (PSD):** Densidad Espectral de Potencia de una señal es una función matemática que nos informa de cómo está distribuida la potencia de dicha señal sobre las distintas frecuencias de las que está formada, es decir, su espectro. Aunque la densidad espectral no es exactamente lo mismo que el espectro de una señal, a veces ambos términos se usan indistintamente, lo cual, en rigor, es incorrecto.

**MIPS:** (Millones de instrucciones por segundo) Sigla utilizada para referirse a la cantidad máxima de instrucciones que un procesador puede realizar durante un periodo de un segundo.

**RNA:** (*Redes Neuronales Artificiales*) Sistemas de procesamiento de información cuya arquitectura y funcionamiento están inspirados en las redes neuronales biológicas y en la forma en que funciona el sistema nervioso.

**Varianza:** Es una medida de su dispersión definida como la esperanza del cuadrado de la desviación de dicha variable respecto a su media.

## INTRODUCCIÓN

Nuestra fauna colombiana ha sido privilegiada en cuanto a número y variedad de especies se refiere. En este trabajo vamos a encaminarnos al estudio y caracterización de algunas de las aves que habitan nuestro territorio, con el fin de proveer una herramienta a la hora de realizar estudios de biodiversidad los cuales son utilizados por los biólogos en trabajos de planeación, manejo o políticas de conservación de diferentes ambientes.

Las personas involucradas en este tema, tanto estudiantes, docentes e investigadores del área de biología de la universidad de Nariño, han desarrollado un extenso banco de datos acerca de las diferentes especies presentes en nuestra región, han realizado diferentes investigaciones, labor de la cual han brotado frutos que ayudan en la tarea de conocer y describir nuestro medio ambiente.

En el presente trabajo se propone desarrollar un sistema inteligente capaz de identificar y clasificar especies de aves a partir de grabaciones de audio hechas por los investigadores, la identificación de las aves iría acompañada de una fotografía y el nombre científico del ejemplar; se propone además implementar el mismo sistema en un dispositivo embebido, con el fin de facilitar la movilidad y la carga de equipos en el campo, y proveer la posibilidad de hacer la identificación en tiempo real. Esta necesidad surge de las prácticas de manejo de la información que actualmente realizan los investigadores, labor que resulta ser ardua y dispendiosa debido a que cada grabación obtenida debe ser escuchada por un humano en su totalidad y comparada con una larga lista de referencias probables; las listas pueden ser de alrededor de 100 especies para determinada ubicación, y para cada especie habrá en promedio cinco cantos de referencia, lo cual al realizarse sin un sistema computacional o electrónico de apoyo se torna poco eficiente y confuso para la persona que clasifica ya que después de escuchar cierto número de cantos se alcanza cierto nivel de fatiga en el oído y la tarea se puede entorpecer.

## **1. DESCRIPCIÓN DEL PROBLEMA**

### **1.1 PLANTEAMIENTO DEL PROBLEMA**

La importancia para los biólogos de saber reconocer certeramente una especie determinada de nuestra fauna silvestre, ha llevado a recolectar y compilar información referente a cerca de especies ya estudiadas que habitan reservas presentes en nuestro departamento; dado que la habilidad para saber reconocer una especie ya sea por su apariencia, rasgos, comportamiento o canto particular, sólo se puede adquirir con la práctica apoyada en el material de consulta.

Esta labor generalmente se realiza de la mano de un experto en el tema, quien debido a la práctica tiene la facilidad de identificar los cantos y especies de donde provienen.

Con el fin de ayudar en la tarea de una identificación de especies aviarias a partir de una señal auditiva, se propone desarrollar un sistema que este en capacidad de analizar la señal y comparar patrones con bases de datos previamente adjuntas para ayudar en la identificación de la proveniencia de determinada señal de audio.

### **1.2 FORMULACIÓN DEL PROBLEMA**

¿Es factible realizar un sistema inteligente que apoyado en una base de datos de 10 especies de aves existentes en la región con su canto característico, pueda identificar un ejemplar contenido en ésta a partir de una nueva señal de audio?

## **2. OBJETIVOS**

### **2.1 OBJETIVO GENERAL**

Desarrollar un sistema inteligente capaz de reconocer y diferenciar cantos de 10 especies de aves diferentes.

### **2.2 OBJETIVOS ESPECÍFICOS**

- Sintetizar la información que se posee acerca de las especies presentes en las diferentes reservas, tales como fotografías, peso promedio, canto característico, etc.
- Crear una base de datos con los patrones extraídos de la información recolectada por el grupo de investigación en Ecología evolutiva- GAICA adscrito al departamento de biología de la Universidad de Nariño.
- Procesar la señal de audio que contiene la información acerca del canto del ave que se pretende identificar y extraer de este los patrones más significativos para su posterior identificación.
- Elaborar una interfaz gráfica la cual desplegara una información acerca de la posible especie de ave que emite el sonido analizado, acompañado de una representación visual, y una información adicional.
- Implementar un sistema embebido de identificación de 10 aves del sistema inteligente desarrollado en lenguaje de bajo nivel.

### 3. JUSTIFICACIÓN

#### INVENTARIOS BIOLÓGICOS Y BIOACÚSTICA

Al realizar estudios en biodiversidad es importante considerar que esta se compone de diferentes elementos que le otorgan su calidad jerárquica organizacional característica y la cual está estrechamente relacionada con atributos como la composición, estructura y funcionalidad (Gastón, 1996) En este sentido el inventariar los elementos de la biodiversidad es un punto crucial y la base que permitirá direccionar los estudios científicos y su aplicación en la toma de decisiones sobre el uso de los recursos así como en el diseño y planteamiento de políticas y estrategias de conservación y manejo.

Una estrategia utilizada al momento de realizar estudios de biodiversidad ha sido el trabajo con grupos indicadores los cuales brindan un estimativo de la diversidad en un área determinada y extrapolar esta información a otros grupos afines así como también permiten evaluar los cambios ambientales y los impactos generados en el ecosistema (Halffter *et al.*, 2001). Dentro de estos grupos se destacan los inventarios de avifauna, los cuales brindan información confiable sobre el estado de conservación de un hábitat determinado, dado a que algunos organismos tienen especificidad de hábitat y son sensibles a los cambios (Halffter *et al.*, 2001); por otra parte, las relaciones inter e intraespecíficas y las funciones ecológicas de esta comunidad les permiten desempeñar un papel dominante en mantener la heterogeneidad espacial y la diversidad taxonómica de los bosque mediante procesos como la dispersión de semillas y la polinización (Stiles, 1983).

Adicionalmente características como el comportamiento llamativo, la identificación, la detección y el estado del conocimiento en este grupo son características que hacen que las aves sean un grupo ideal para inventariar gran parte de la comunidad con buenos resultados de certeza y la caracterización de sus hábitats (Stotz *et al.*, 1996).

Para realizar inventarios de la avifauna se utilizan diversas metodologías entre la que se destacan métodos intrusivos y no intrusivos, dentro de los primeros es conocido el empleo de redes de niebla, equipos ideales para la captura de individuos; esta metodología permite tomar diversos datos biológicos de los individuos capturados que aportan al estado de conocimiento de la especie como información sobre el estado del plumaje y condiciones reproductivas (Ralph *et al.*, 1996) que son un indicativo importante para entender aspectos poblacionales y de la dinámica de la comunidad, aunque cabe resaltar que este método permite inventariar especies que prefieren hábitats específicos como el sotobosque (Ralph *et al.*, 1996).

Por otra parte uno de los métodos más utilizados y que no genera un contacto directo o de manipulación de los organismos son las observaciones directas (Villarreal *et al.*, 2006), que con ayuda de herramientas ópticas como binoculares y teleobjetivos permiten apreciar en mayor detalle los individuos y obtener información sobre algunos aspectos del comportamiento. Sin embargo esta técnica presenta dificultades en hábitats tropicales con vegetación densa (Villarreal *et al.*, 2006).

Teniendo en cuenta estas consideraciones sobre las metodologías utilizadas en los inventarios de avifauna, actualmente se está trabajando en el reconocimiento de las vocalizaciones ya que las aves y algunos grupos específicos han desarrollado evolutivamente un complejo sistema de comunicación (Kroodsma, 1996.).

El empleo de técnicas bioacústicas tiene aplicaciones en la sistemática, la conservación y el manejo de las poblaciones de aves silvestres (Tubaro, 1999), destacándose el uso de los cantos en la diagnosis de nuevas especies y el estudio de sus relaciones macro sistemáticas, donde las diferencias del canto han permitido esclarecer diferencias entre especies (Becker, 1982; Payne, 1986; Tubaro, 1999). La bioacústica es una herramienta poderosa de bajo costo en términos de tiempo y dinero para el monitoreo de la biodiversidad (Tubaro, 1999). Más allá de la composición de especies un detallado análisis bioacústico permite estimar las densidades poblaciones relativas y acompañada de la técnica del playback que consiste en la reproducción de cantos naturales, es una ayuda en la identificación específica e individual (Baptista & Gaunt, 1997), el conteo y captura de individuos y el mapeo de territorios (Falls, 1981).

En este sentido, una de las mayores dificultades en campo está relacionada con la identificación rápida y segura de los cantos y la especie que representan. Un programa que permita comparar los registros fonográficos obtenidos en campo con una base de datos preestablecida permitirá incrementar los estimativos de riqueza avifaunística de una zona y a la acumulación de información bioacústica importante en el proceso de caracterización ecológica evolutiva de las especies.

## **4. REVISIÓN SOBRE EL RECONOCIMIENTO BIOACÚSTICO**

Empezaremos con una revisión acerca tanto de conceptos fundamentales relacionados con la identificación de patrones como de los métodos más utilizados para el análisis de estos patrones.

### **4.1 EL RECONOCIMIENTO DE CANTOS VOCÁLICOS**

Sin duda alguna el oído humano es el sistema más eficiente y complejo a la hora de identificar, extraer, discriminar y separar sonidos, labor que a un humano le resulta bastante fácil debido a la familiarización que se tiene con este; si quisiéramos que un sistema artificial lograra el mismo desempeño que el oído humano debemos empezar por analizar en qué se soporta este sistema y nos daremos cuenta que en una situación como lo es escuchar tocar una banda completa, el oído sabe diferenciar la trompeta, el bajo, el tambor, la guitarra, la voz etc., y esta habilidad se debe a que previamente durante toda su vida ha podido escuchar cada instrumento por aparte varias veces y ha almacenado esta información en el cerebro, también debemos notar que a pesar de que el oído nunca haya oído una trompeta en particular, pero haya oído otras decenas de éstas, inmediatamente la identificaría, a pesar de no ser del mismo material, medida etc.. Si hacemos una analogía con un sistema artificial, podríamos decir que si contamos con una etapa de captura de una señal y una base de datos de referencia se podría buscar las semejanzas que hay entre la señal de entrada y las almacenadas en la base de datos; claramente el oído humano es un sistema flexible y supera enormemente cualquier sistema actual para el reconocimiento acústico, pero al ser el oído el sistema más eficiente entonces es el modelo a seguir.

Debido a la semejanza que presenta el sistema de reconocimiento de cantos vocálicos con los sistemas de reconocimiento de voz, nos cimentaremos en las investigaciones y procesos que se hayan utilizado en el reconocimiento de voz con las respectivas adecuaciones o modificaciones a las que haya lugar.

En un canto de ave en general se diferencian algunas partes como lo son la introducción y el trino, sin embargo los sonidos emitidos por el espécimen no son siempre los mismos, se puede presentar en ocasiones únicamente el trino o únicamente la introducción, repetirse una de las dos partes o presentarse solo un segmento de estos, etc. También se encuentra en ocasiones un registro de un canto completo e instantes más tarde el mismo individuo emite otro canto con una variación en alguna de las notas.

Para observar las variaciones de los cantos de la misma especie resulta de gran utilidad comparar los Espectrogramas, pues dan una idea visual del canto en el tiempo, frecuencia y amplitud.

Los espectrogramas resultan de un gráfico en tres dimensiones graficando por tramos el espectro de la señal; la señal se parte en tramos iguales, se aplica una ventana a estos datos para evitar aparición de frecuencias en el corte y se calcula y grafica la transformada de Fourier, el proceso se repite tomando el siguiente tramo superpuesto al 50% con el anterior para nuevamente calcular el espectro y graficarlo en el tiempo correspondiente.

El espectrograma se presenta generalmente en un grafico 2D, con los ejes de tiempo y frecuencia diferenciando la amplitud del espectro por el color, un punto con color cercano al azul determina que en determinado tiempo la frecuencia correspondiente no tenía gran energía comparada con un punto rojo. El matiz de colores corresponde al espectro de la luz visible desde el azul oscuro al rojo, siendo al rojo el que denota mayor energía. En la figura 1 se aprecia un ejemplo de dos espectrogramas con variación en notas para una misma especie.

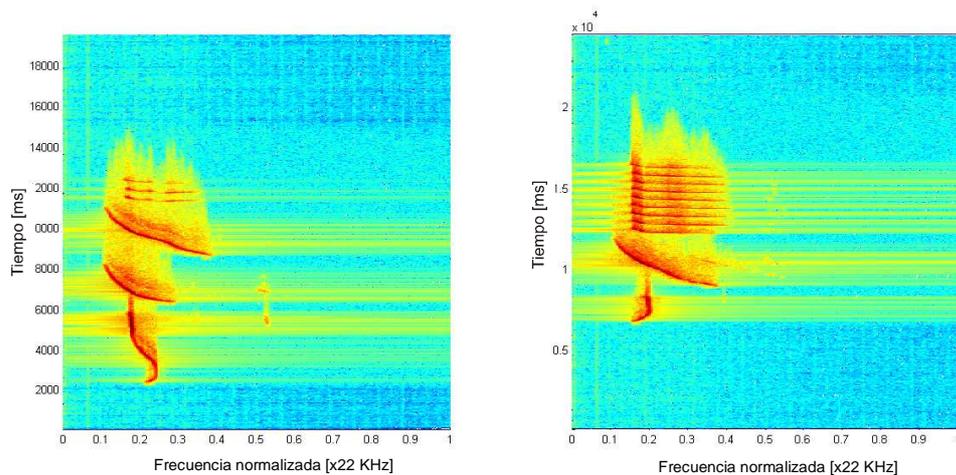


Figura 1. Variación entre dos cantos de la misma especie

La variación geográfica del canto es bien conocida en numerosas especies de aves, observándose variaciones en poblaciones de la misma especie con respecto al ancho de la banda, el número de picos, la frecuencia dominante y fundamental, así como en el intervalo entre notas.<sup>1</sup>

---

<sup>1</sup> Hernandez, Jose M. La bioacústica como instrumento en los estudios de biodiversidad y conservación. [artículo de internet]. <http://jmhweb.wordpress.com/docencia/master-en-biologia-de-la-conservacion-ucm/la-bioacustica-como-instrumento-en-los-estudios-de-biodiversidad-y-conservacion/> [Consulta 1 abril 2011]

Este tipo de variaciones se estudian actualmente como un tipo de evolución en la forma de comunicarse entre los mismos individuos, haciendo la analogía con sistemas de reconocimiento de voz se podría decir que a un sistema que está entrenado para reconocer la palabra "**ADELANTE**", se ingresase en su lugar la palabra "**ADELANTA**" y debe ser reconocido ya que la diferencia entre las dos palabras es mínima.

## **4.2 TÉRMINOS Y CONDICIONES EN EL RECONOCIMIENTO DE CANTOS VOCÁLICOS**

La identificación de una señal de audio depende de distintos factores presentes en el ambiente, y al tratarse de aves los hay de diferentes formas, desde la presencia de otras especies como los son grillos, ranas, factores como el viento, ríos, hasta la "aleatoriedad" o variación con que decida cantar el individuo.

### **4.2.1 Agentes externos**

La presencia de ruido proveniente de agentes externos es muy frecuente en señales grabadas en campo abierto, algunos de estos factores están claramente identificados como lo resulta un río o el viento que producen un ruido de muy baja frecuencia en la grabación, y por tanto para contrarrestar el efecto de estos agentes se hace uso de forros especiales en el micrófono o se utiliza micrófonos direccionales con el fin de atenuar o disminuir en la mayor proporción los efectos sobre la grabación.

Hay otro tipo de agentes como los grillos o las chicharras que generan un sonido constante a lo largo de todo el registro, dejando una marca muy clara en una frecuencia cercana a los 10KHz, en ocasiones se encuentran registros de aves cercanas a esa frecuencia, de tal forma que usando software especializado como Adobe Audition es posible editar la presencia de sonidos no deseados.

### **4.2.2 Variación en la composición del Canto**

Como habíamos descrito en la sección anterior, en ocasiones se encuentran variaciones en la estructura del canto, las cuales siempre y cuando sean en pequeñas proporciones deben poder ser superadas por el sistema, ya que si bien son frecuentes, también es claro que la variación no es muy grande.

Es seguro que las variaciones ocurren constantemente, pero también es cierto que la marca que un canto deja en el espectrograma se caracteriza por ser bien definida, es decir, para un ave en particular se aprecia claramente la frecuencia con que inicia y como varía a lo largo del tiempo en una marca caracterizada por unas "estrías" propias para cada ave.

En muchos grupos animales, la bioacústica puede representar una herramienta muy útil en la caracterización y diferenciación de poblaciones de la misma especie. Dado que, como cualquier carácter etológico, la emisión de sonido presenta un componente anatómico (el órgano productor de sonido) y un componente comportamental (la serie de movimientos y procesos que hacen funcionar el órgano emisor), la variabilidad en la señal emitida puede ser mucho más amplia que la que presentan los caracteres exclusivamente anatómicos. Esto está más acusado en aquellos grupos donde los individuos aprenden el canto por imitación durante el período juvenil<sup>2</sup>.

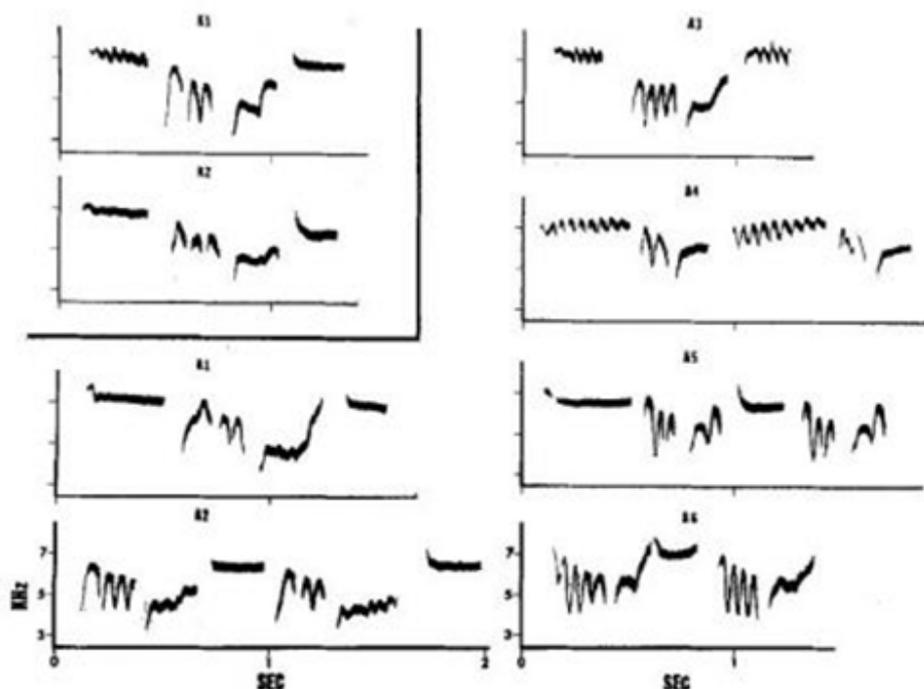


Figura 2. Espectrogramas de diferentes dialectos en el agateador norteño (*Certhia familiaris*). Tomado de Hernández, Jose M. La bioacústica como instrumento en los estudios de biodiversidad y conservación.

Tales diferencias eran de esperarse al tratarse de seres vivos que interactúan con el medio en el que viven, se dice que si el medio en el que se desenvuelven se torna ruidoso ya sea por acción de la cercanía de la ciudad o por la presencia de especies que emiten sonidos muy notorios, el individuo trata de sobresalir acentuando alguna nota o ajustando el timbre del canto.

<sup>2</sup> Ibid.

#### **4.2.3 Número de cantos presentes en una grabación**

Debido a que se realiza grabaciones de 1min de duración promedio, en ese lapso de tiempo el individuo habrá emitido varias repeticiones, y seguramente no sea el único individuo emitiendo trinos, por tanto se hace necesario seleccionar la parte de la grabación en donde está la información de interés, para separarla filtrarla e ingresarla al sistema. Cabe resaltar que la grabación debe ser de una buena calidad, en donde resalte el sonido emitido por el ave, por encima de los demás sonidos presentes.

#### **4.2.4 Delimitaciones de la implementación**

Actualmente se conoce de estudios que pretenden identificar especies a partir de su sonido, y lo que buscan es dar una aproximación a la especie que presente características similares, tanto en ancho de banda del trino como duración y ubicación en el espectro, y es que la búsqueda además de realizarse en una base de datos extensa, (pues se tiene registros por todo el mundo de aves cantoras), presenta variaciones de la frecuencia y notas presentes de acuerdo a la ubicación geográfica, entonces los sistemas que reconocen los distintos cantos lo que hacen es dar un acercamiento a una base de datos reducida que se asemeja a la señal de interés, para así completar la tarea de identificación con el sistema más eficiente en el campo: “el oído humano”, ya que por más confiable que un sistema sea, la última decisión es la que sale de la comparación que hace el oído humano.

### **4.3 METODOS DE RECONOCIMIENTO CANTO CARACTERISTICO**

Como es de esperarse, la solución de un problema siempre se puede abarcar desde distintos puntos de vista, a continuación se hace una descripción de algunos de los métodos usados para la identificación de patrones, se analizan las ventajas, desventajas y las razones por las que luego se escoge el método para el sistema desarrollado.

#### **4.3.1 Dinamyc Time Warping (DTW)**

El reconocimiento de determinado canto del ave presenta ciertas dificultades también presentadas en el reconocimiento de voz o de audio en general; ya obtenidas las características de la señal ingresada, el camino directo al reconocimiento podría ser medir la distancia que separa a las características recién calculadas con las que ya existen. Es claro que podría encontrarse la distancia Euclidiana y encontrar cuan diferentes son las señales.

Sin embargo hay que recalcar que dos cantos que corresponden al mismo tipo de ave no solo presentaran diferencias en la amplitud de las señales, también existen espacios de canto que se prolongan o acortan junto con diferencias en la alineación con el tiempo

Estamos frente a un problema con señales a comparar que internamente varían en longitud y velocidad. El Dynamic Time Warping es un algoritmo que permite medir la similitud entre dos secuencias, las cuales pueden presentar este tipo de alteraciones.

Como resultado del algoritmo tendremos una distancia ajustada a lo más mínimo posible para acercar las dos señales, y claramente que tan bueno haya resultado el ajuste nos dará un estimativo de la cercanía entre las señales tratadas, el canto a analizar y cada uno de la base de datos.

La Figura 3 representa claramente dos palabras localizadas en el eje vertical y horizontal, que son las mismas palabras pero presentan cambios de longitud y velocidad, una simple distancia euclidiana nos mostraría que las palabras son totalmente distintas, pero haciendo una aplicación correcta de DTW se encontrará un buen porcentaje de similitud.

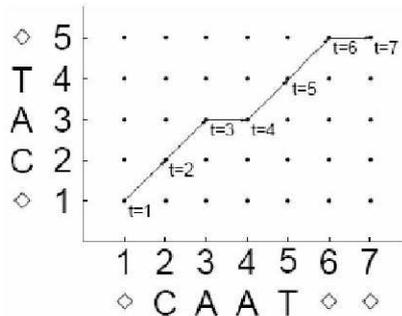


Figura 3. Variación de trama en longitud y velocidad.

La distancia euclidiana entre dos vectores A y B que internamente resultan de la agrupación consecutiva de n vectores se calcula como

$$d(A, B) = \sqrt{\sum_{i=1}^n (A_i - B_i)^2} \quad (1)$$

La función que realiza DTW es minimizar esta distancia moviéndose por tramos en ambas secuencias

El problema básico consiste en encontrar el camino con la distancia más corta o con el menor costo asociado a éste a través de la rejilla que comienza en el nodo origen (0,0) y termina en el nodo terminal (I,J).<sup>3</sup>

$$D(i, j) = \min[D(i-1, j-1), D(i-1, j), D(i, j-1)] + d(i, j) \quad (2)$$

Donde D(i,j) es la distancia global hasta el punto (i,j) y d(i,j) es la distancia local hasta este punto, operando la anterior ecuación a través de toda la palabra o pedazo de canto a analizar, internamente el algoritmo encontrará coeficientes a,b para una distancia global D(a,b) hasta (a,b) y así sucesivamente va encontrando distancias mínimas, de las cuales al final se tendrá un acumulado que dependiendo si pasa de cierto nivel estaremos hablando de si se reconoce o no la señal tratada.

<sup>3</sup> Ripoll Solano, Lácides Antonio. Verificación de hablante basado en Dynamic Time Warping, Barranquilla: Ingeniería y Desarrollo, 1998

El método resulta eficiente cuando se trata de reconocimiento de voz debido a que el individuo que trata de generar un patrón que sea reconocido por el sistema es un humano, y por tal razón sabe que si el sistema está entrenado para reconocer un comando en especial o una palabra determinada, esta debe ser pronunciada lo más claro posible, de forma pausada y completa, mientras que al tratar con aves se encuentra que es un ser totalmente autónomo y emite los sonidos en el orden que “desee”, como el nombre del método lo indica la alineación se hace en el tiempo esperando contar con patrones similares, pero las variaciones existentes en la estructura del canto hacen que el método no se pueda emplear a cantos que pueden variar en su composición, sin embargo es un método que ha resultado muy eficiente en reconocimiento de voz y se decidió examinar si se ajustaba o no al reconocimiento de cantos vocálicos.

#### **4.3.2 Caracterización y reconocimiento mediante Densidad Espectral de Potencial (PSD)**

Como se mencionó anteriormente el canto del ave es una función discreta en el tiempo, se tienen espacios vacíos, oscilaciones en intervalos y cambios bruscos de amplitud; analizar la señal en el dominio del tiempo es una opción que dedicaría una gran complejidad en el análisis de datos y un gran uso de memoria por tratarse de una inmensa cantidad de datos.

El trabajar en el dominio de la frecuencia ahorra bastante trabajo por que caracteriza la señal con menos datos e informa que frecuencias predominan en determinado canto, además, es claro cada ave al oído humano experimentado se diferencia por que predominan sonidos más agudos o graves.

Power spectral density (PSD) es una función proveniente del espectro o transformada de Fourier de una señal, indicando a que frecuencias las variaciones de la señal son más fuertes (energía). En otras palabras, determina como está distribuida la energía de la señal en el espectro.

Claramente el uso de esta función radica en la búsqueda de oscilaciones escondidas que revelan gran cantidad de energía de una señal o la capacidad que se tendría de centrar cierta energía, o si por el contrario solo se tiene ruido que es difícil de encaminar a un propósito, al respecto es importante resaltar que una PSD plana determinará que la señal tratada es únicamente ruido blanco.

La unidad de esta función es de energía por frecuencia, y en general existen dos métodos para calcularla, directamente desde la DTF (transformada discreta de Fourier) o usando el cálculo de la autocorrelación.

Se sabe que la DTF de una señal a analizar resulta de:

$$X_k = \sum_{n=0}^{N-1} x(n) e^{-j2\pi kn/N} \quad (3)$$

Donde  $x(n)$  es el vector de entrada a analizar y  $N$  la longitud de este vector.

Y su densidad espectral de energía DEE:

$$DEE = |X_k|^2 \quad (4)$$

Como estamos interesados en la densidad de potencia también llamado periodograma habrá que dividir esta energía por el tiempo, que resulta ser el número de muestras por el tiempo de muestreo.

Existe un método computacionalmente eficiente que divide todos los  $N$  datos, en  $M$  segmentos de puntos, se realiza un ventaneo a cada segmentos y es a cada segmento que se realiza el periodograma, el resultado esperado de toda la señal corresponderá al promediado de todos los periodogramas parciales.

Los resultados de este método son coherentes con la potencia de cada frecuencia, sin embargo el haber analizado espectralmente la señal por partes puede hacer aparecer lóbulos de frecuencias de baja amplitud que en la señal total pueden no existir, es necesario entonces escoger una función de ventana adecuada.

Como método alternativo se puede calcular la densidad espectral de potencia mediante el teorema de Wiener-Khinchin que determina la ecuación

$$S_{xx} = TF\{R_{xx}(\tau)\} \quad (5)$$

Donde  $R_x$  es la función de autocorrelación de  $x(t)$ . La densidad espectral de potencia resulta ser entonces la transformada de Fourier de la señal de autocorrelación de  $x(t)$ .

Definiéndose la función de autocorrelación como la correlación cruzada de la señal consigo misma; dado que frecuentemente las autocorrelaciones se calculan para señales centradas alrededor del cero, es decir con un valor principal de cero. La definición de la autocorrelación viene dada por:

$$R(j) = \sum_n x_n x_{n-j} \quad (6)$$

y estadísticamente esa función es ampliamente usada para encontrar parámetros repetitivos dentro de una señal.

Ambos métodos planteados son correctos, sin embargo el primer método es el más usado a nivel computacional.

En general la función PSD en aplicación de reconocimiento de patrones a pesar de ser muy parecida a la transformada de Fourier presenta algunas ventajas como en cuanto el tamaño del vector resultante dado que sin importar el vector de entrada, la salida va a tener longitud constante generalmente de 129 datos, aunque se puede usar un vector de mayor o menor longitud, significando en todo caso un barrido en frecuencia desde cero hasta la mitad de la frecuencia de muestreo. Además resulta que los cantos vocálicos que se están tratando presentan ciertos patrones particulares para cada especie, y en comparación a la FFT, la PSD presenta menor susceptibilidad al ruido, es una curva suave de pequeño tamaño, en pocas palabras presenta buen compromiso entre cantidad de información, sensibilidad al ruido y complejidad computacional.

A continuación se presenta la función PSD aplicada a diferentes cantos, la diferenciación entre cada canto es evidente y por tanto más trabajable en un sistema de reconocimiento de patrones.

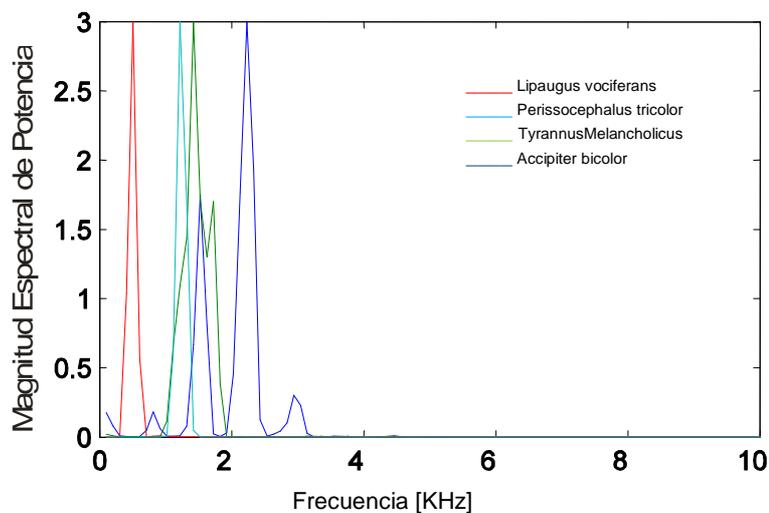


Figura 4. PSD diferentes aves

Así mismo un aspecto a tener en cuenta es como varía el PSD en función de una variación en el trino de una misma especie, es decir si las variaciones en las notas son tan comunes como se menciona en la sección anterior, estas variaciones se deben poder superar en el patrón PSD sin alterar la forma principal. En la figura 5 se aprecian dos ejemplos de cómo varía el patrón PSD para dos cantos emitidos por un mismo individuo en una misma grabación en donde se observa el cambio en algunas de las notas emitidas:

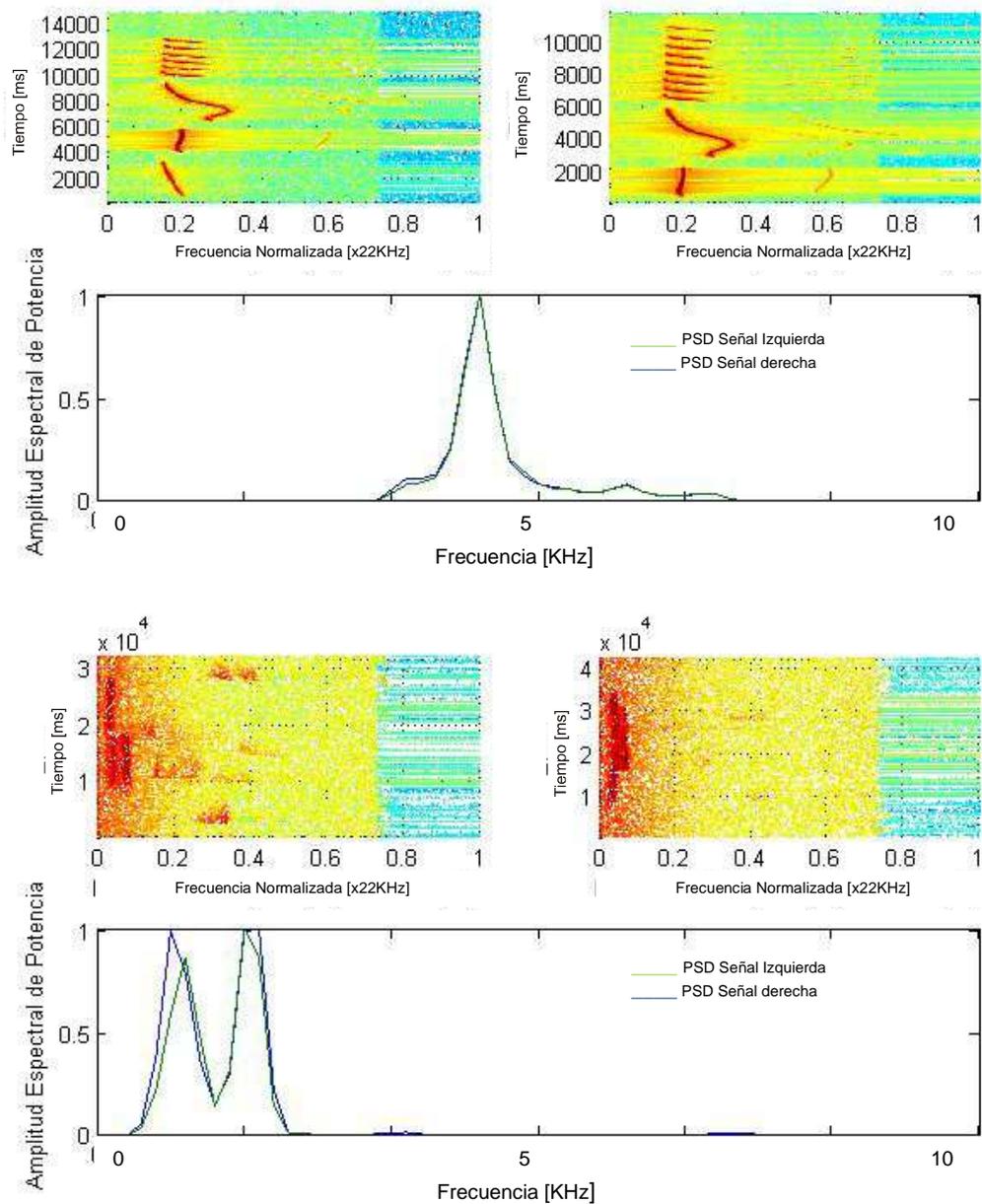


Figura 5. Variación PSD grabaciones mismo individuo.

#### 4.3.2.1 Principal Component Analysis (PCA)

Se plantea la técnica estadística PCA (Análisis de Componente Principal) como opción al reconocimiento de un patrón, en nuestro caso el canto de un ave en particular, dada una base de datos con múltiples variables.

PCA ha sido ampliamente usada en técnicas como:

- Reducción de factores como ruido y redundancia.
- Búsqueda de interrelaciones entre los datos para hacer un análisis claro del conjunto.
- Creación de nuevas variables que sirven de entrada a otros sistemas que necesitan información sintetizada.
- Reconocimiento y análisis de imágenes.

Mediante esta técnica es posible por ejemplo reducir un problema de tres variables descrito en el espacio a un gráfico 2D en el plano que resulta más fácil de analizar, basándose en relaciones y redundancia de las variables.

En nuestro objeto de estudio, existe una matriz base de datos referente al canto de aves (espectro), que necesita ser sintetizada para encontrar datos claves que determinaran si un nuevo canto existe o no en la base de datos.

Dada una matriz de datos es posible determinar la varianza (desviación estándar al cuadrado) de cada variable como,

$$\sigma^2 = \frac{\sum_{i=1}^n (x_i - \bar{x})^2}{n - 1} \quad (7)$$

$\sigma^2$  determina cuánto varían los valores de x respecto a su media.

Si se trabaja con dos variables también es posible determinar la varianza entre ellas llamada covarianza.

$$\text{cov}(X, Y) = \frac{\sum_{i=1}^n (X_i - \bar{X})(Y_i - \bar{Y})}{n - 1} \quad (8)$$

Valor que determina cuánto varía la dimensión x de su media respecto a la otra variable.

Si se analiza la covarianza entre una dimensión y si misma se obtiene la variancia de sí misma.

Así, si la covarianza entre las variables X y Y es positiva nos da a entender que ambas se incrementan juntas, si es negativo significa que cuando una crece la otra decrece, por último si la covarianza resulta ser cero ambas variables X y Y son independientes una de otra.

Ahora si se trabaja con más variables, por ejemplo tres, tendremos más covarianzas, formando la matriz:

$$C = \begin{pmatrix} cov(x,x) & cov(x,y) & cov(x,z) \\ cov(y,x) & cov(y,y) & cov(y,z) \\ cov(z,x) & cov(z,y) & cov(z,z) \end{pmatrix} \quad (9)$$

Es claro que para J variables se tendrá una matriz J\*J.

Y hasta aquí tenemos una matriz de igual tamaño que la base de datos, pero informándonos como cambia cada variable respecto a la otra, ahora, es necesario a partir de esta matriz C encontrar los Eigenvectores y Eigenvalores, que nos permitirán disminuir el número de variables.

Los eigenvalores resultan importantes en cuanto le dan el peso a cada eigenvector, es decir, si bien cada eigenvector determina una nueva dimensión este es tan importante para considerarlo en nuestro estudio como su respectivo eigenvalor sea.

Podemos tener en cuenta todos los eigenvectores y entonces no habremos reducido el número de variables, y si solo tomamos en cuenta los eigenvectores que tengan un eigenvalor mayor que cierto valor habremos reducido el número de variables.

Haciendo un recuento del procedimiento para reducir el número de variables es necesario antes que nada restarle la media a cada variable, calcular la matriz de covarianza C a esta matriz, calcular los eigenvectores y eigenvalores y desechar los que tengan un eigenvalor por debajo de un cierto nivel (lógicamente entre mas eigenvectores eliminemos más información se perderá pero más sencillo será el estudio).

Para generar la nueva matriz de datos, que debería tener menos columnas. Armamos el vector

$$eigenvectors = (eig1 eig2 eig3 ..... eigp) \quad (11)$$

Donde  $p < n$ , y  $n$  es el numero de variables original.

Nuestra nueva matriz de datos será

$$\text{nuevamatriz} = \text{eigenvectors}^T * \text{datosentrada}^T \quad (12)$$

Donde 'datosentrada' ha resultado de la matriz original de datos después de restarle la media a cada variable (es una matriz centrada de la matriz de datos).

Si se tiene una matriz datos de entrada  $m \times n$ , eigenvectores será  $n \times p$  y la nueva matriz será  $p \times m$ . Aquí  $m$  es el número de individuos y  $p$  el número de nuevas variables ( $< n$ ).

#### 4.3.2.1.1 Aplicación en reconocimiento de patrones

El PCA puede aplicarse en reconocimiento de imágenes o algún patrón, esto haciendo que el número de variables resultantes sea igual al número de imágenes o patrones en la base de datos

Dada una matriz  $A(m \times n)$  que caracteriza a  $n$  individuos con  $m$  características  $m > n$ . ( $m$  puede ser un valor característico para cada pixel en una imagen o el valor del espectro de una señal).

Podemos convertir a  $A$  en una matriz centrada y luego operar así:

$$L = A'A \quad (13)$$

Y  $L$  resultará una matriz  $n \times n$  (Del tamaño del número de individuos)

A esta matriz se le puede encontrar los eigenvalores y eigenvectores.

Eigenvectores será una matriz de  $n \times n$ , aquí no se discriminará variables.

Aplicando la ecuación

$$\text{eigenfaces} = A * \text{eigenvectors} \quad (14)$$

Nuevamente se encontrara una nueva matriz  $m \times n$  pero que ha discriminado correlaciones en la matriz, esto es claro en el tratamiento de imágenes pues si  $A$  contenía toda la información referente a imágenes a reconocer, eigenfaces también representaran imágenes pero a las que se les ha extraído patrones comunes entre ellas y solo se ha dejado lo primordial que las distingue.

Para el reconocimiento de si una nueva imagen o patrón está en la base de datos original se debe armar una serie de vectores llamados proyecciones resultantes de la multiplicación entre eigenfaces y el vector ( $m \times 1$ ) que caracteriza a cada individuo.

Y cuando queramos analizar una imagen o un patrón nuevo, esta también debe multiplicarse por eigenfaces. Y basta con obtener la distancia euclidiana entre este vector y cada uno de las proyecciones obtenidas anteriormente, y ver cual distancia es la menor. La distancia menor corresponderá a la imagen que más se acerca a la imagen o patrón que ingresamos para que se reconozca.

#### 4.3.2.2 Coeficiente de Correlación

Entre los algoritmos más usados para comparar dos tramas de datos esta la correlación, es el más simple pues resulta de la aplicación del principal método estadístico para determinar dispersión de datos.

Las aplicaciones de Correlación son muchas tanto por el hecho de que es un método directo sin la búsqueda de patrones o tramas específicas como por el uso de operaciones sencillas.

Estadísticamente encontrar la correlación entre dos variables A y B significa encontrar cuan relacionadas están las variables en cuanto a que varíen simétricamente.

El coeficiente de correlación más usado es el coeficiente de correlación de Pearson y se encuentra directamente como era de esperarse a partir de la covarianza entre las tramas A y B (introducida en la revisión de PCA).

Matemáticamente esta correlación ( $\rho$ ) se define como la covarianza entre A y B sobre el producto de las desviaciones estándar de cada una.

$$\rho = \frac{\sum_{i=1}^N (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^N (x_i - \bar{x})^2} \sqrt{\sum_{i=1}^N (y_i - \bar{y})^2}} \quad (15)$$

Hay que anotar que a diferencia de la covarianza, la correlación nos genera un valor independiente de la escala de las variables y el resultado será un valor entre -1 y 1.

Análogo al caso de covarianza cuando el factor correlación sea 1, nos denotara una dependencia directa entre la dos variables (poco usual en señales reales, por ruido o desfase como mínimo), mientras que una correlación igual a 0 denota una independencia de las variables, lo que no significa que las señales puedan estar relacionadas de forma no lineal.

### 4.3.2.3 Redes Neuronales Artificiales

Las redes neuronales pueden ser usadas como herramienta en el reconocimiento de patrones, es claro que el canto de un ave presenta patrones que ante un elemento conocedor de los tipos de patrones, será reconocido con cierto grado de certeza.

Las redes neuronales han sido entrenadas para realizar funciones complejas en muchos campos, incluyendo el reconocimiento de patrones, la identificación, la clasificación, la voz, la visión y los sistemas de control.

Dentro de las propiedades más importantes de las ANN se incluyen:

- Procesamiento paralelo: una gran cantidad de neuronas actúan a la vez, ahorrando tiempo y facilitando el trabajo en elementos electrónicos.
- Generalización: las redes almacenan generalizaciones de los datos de ahí que pueden eliminar ruido y su funcionamiento es más flexible.
- Gran cantidad de elementos que vuelven redundante el trabajo.
- Entrenamiento y adaptabilidad.

La estructura de una neurona es muy similar a una neurona biológica y puede representarse así:

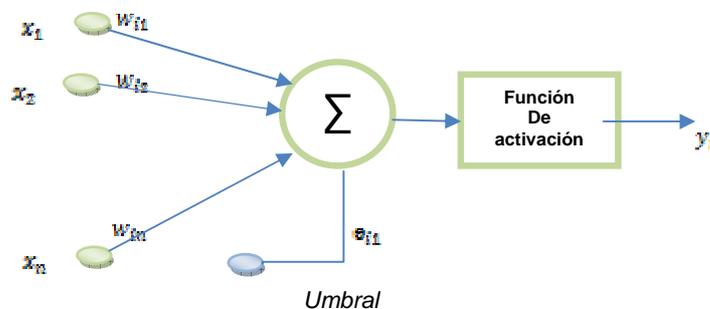


Figura 6. Modelo neurona

Entre las partes fundamentales se pueden diferenciar:

- **Entradas:** Entradas  $x_j$  a la neurona.
- **Pesos (w):** Cada entrada tiene su propio peso relativo el cual proporciona la importancia de la entrada dentro de la función de agregación de la neurona. Los pesos son coeficientes que pueden adaptarse dentro de la red. Ellos son la medida de la fuerza de una conexión de entrada y pueden ser modificadas en respuesta de los ejemplos de entrenamiento

- **Función de propagación** Esta regla permite obtener, a partir de las entradas y los pesos el valor del potencial postsináptico de la neurona: La función más habitual es la suma ponderada de todas las entradas. Podemos agrupar las entradas y pesos en dos vectores ( $x_1, x_2, \dots, x_n$ ) y ( $w_{1j}, w_{2j}, \dots, w_{nj}$ ), así podemos calcular esta suma realizando el producto escalar sobre estos dos vectores.
- **Función de activación o transferencia** El resultado de la función de propagación, es transformada en la salida real de la neurona mediante un proceso algorítmico conocido como función de activación. En la función de activación el valor de la salida de combinación puede ser comparada con algún valor umbral para determinar la salida de la neurona. Si la suma es mayor que el valor umbral, la neurona generará una señal. Si la suma es menor que el valor umbral, ninguna señal será generada. En la siguiente figura se presentan tres tipos de funciones de activación más comunes: lineal, de mantenimiento y sigmoideal, destacando de esta última su amplio uso en reconocimiento de patrones y su versatilidad en el empleo de diferentes tipos de entrenamiento.

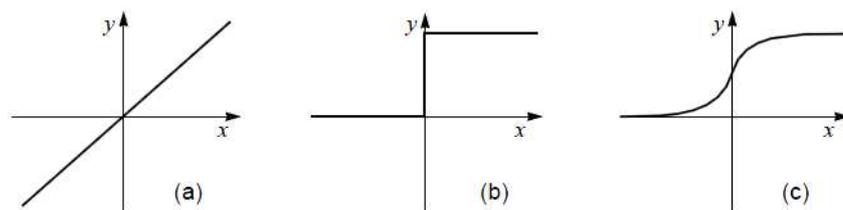


Figura 7. Funciones de activación (a) lineal (b) mantenimiento (c) sigmoideal

#### 4.3.2.4 Entrenamiento y uso de Redes Neuronales

Aquí se parte de una red neuronal con valores iniciales aleatorios en las conexiones, la red neuronal revisará una gran cantidad de información para que ir ajustando continuamente los pesos para que el sistema responda de la forma deseada.

Una de las aplicaciones más importantes de las redes neuronales es el diseño de sistemas que antes de la puesta en marcha tienen determinado claramente la salida deseada del sistema ante cada entrada, así, la red neuronal recibe un par de información entrada salida la vez, y se va ajustando los pesos; La salida obtenida y la salida deseada generan un error a minimizar mediante el ajuste de los pesos.

Además existen aplicaciones donde el entrenamiento no es supervisado, aquí no se conoce la señal que debe dar la red neuronal (señal deseada), por ende no se realizan comparaciones entre las salidas reales y las esperadas. Los pesos se modifican de forma que la red produzca valores 'consistentes' a la salida según la

lectura de la información en la entrada. Es claro que este entrenamiento generara un agrupamiento o clasificación de la información a la entrada.

Ya modificados los pesos internos en la red neuronal se entiende que estos son fijos y ante una entrada la red usara el vector de entrada lo procesara en forma paralela hasta llegar a la salida, internamente las funciones son sencillas y resultan de operaciones matemáticas en cascada, sin embargo hay q resaltar que si se tienen redes neuronales con realimentación, es de esperar que el sistema se torna más dinámico y la estabilidad del sistema debe garantizarse antes de su uso; esto se hace en general revisando la estabilidad en reposos del sistema, si las salidas son acotadas en todo momento y si existen las derivadas en el tiempo de cada función que describa la red.

#### **4.4 ALGORITMO DE CORTE**

Las grabaciones sometidas al proceso de identificación, son grabaciones en las que generalmente se encuentra más de un sonido de fondo, o hay más de una especie interactuando con el medio, podemos imaginar un ambiente natural donde los diferentes sonidos del viento, las aves, ranas, ríos etc. pueden estar presentes y quedar registrados en la grabación, por tanto el problema de la extracción del segmento que contenía la información deseada era un problema que quisimos solucionar estableciendo determinados criterios. Primero se hizo un análisis para tratar de descartar la información por frecuencias, y así dejar únicamente los sonidos que se encuentren dentro del ancho de banda de las aves a ser analizadas, y tras examinar el pequeño grupo de aves se pudo notar que si bien cada ave tenia bien definido su propio ancho de banda, también se observo que los anchos de banda están distribuidos a lo largo del espectro llegando hasta aproximadamente los 10KHz, traslapándose con los sonidos que se quería ignorar, y por eso determinamos descartar este forma de selección.

Con la incertidumbre de no tener una base para determinar el segmento de interés, se intento fijar un umbral de ruido para estimar el inicio y fin del corte, encontrando que los niveles de ruido eran relativos y podían diferir de una grabación a otra, en lo que considerábamos un “silencio” con una proporción de hasta 20dB, debido a que en ocasiones el ruido constante de fondo influye sustancialmente en dicho umbral.

Así que teniendo en cuenta los puntos anteriores y apoyados sobre la premisa de decir que la grabación a identificar debe ser de calidad aceptable en donde haya por lo menos una intervención del individuo a analizar, se procedió a desarrollar un algoritmo el cual segmenta toda la señal en pequeños tramos iguales, calcula la energía de cada tramo y guarda este valor en una posición secuencial de un vector  $V$  en un proceso denominado ventaneo que al final arroja como resultado la información de la envolvente en potencia de la señal.

Durante el ventaneo como tal se toma cada pequeño tramo de la grabación en el dominio del tiempo y se lo multiplica por una forma predefinida o “ventana”, con el fin de minimizar efectos de altas frecuencias en los extremos del tramo producto del corte abrupto que se hace; en tanto a la ventana existen varios modelos (dependiendo de la forma), después de probar con ventanas como Káiser, Hamming y Bartlett, se noto que no había mayor cambio en la envolvente y se decidió dejar por defecto la ventana de Hamming:

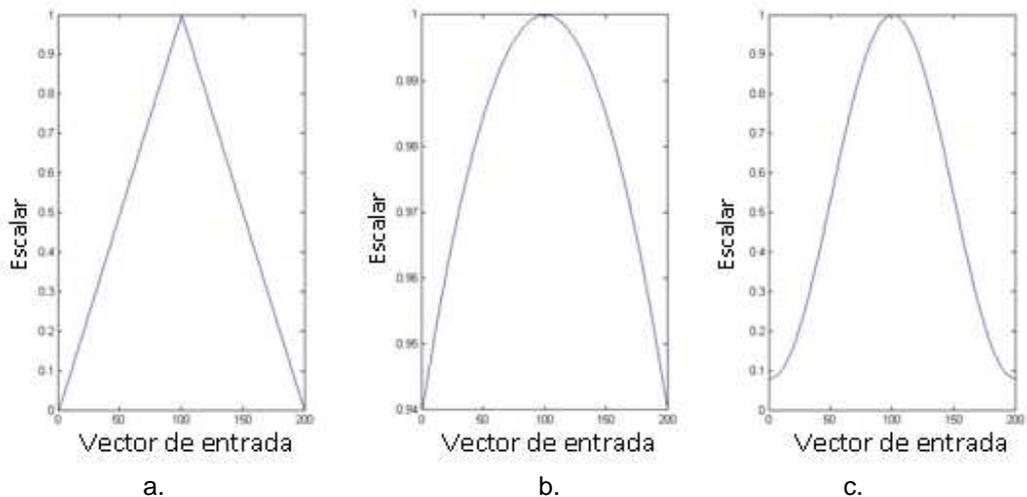


Figura 8. Funciones ventana. a) Barlett, b) Káiser, c) Hamming

Otro de los parámetros que hay que tener en cuenta a la hora del ventaneo es el grado de traslapamiento o superposición entre una muestra y la siguiente, con el fin de aprovechar al máximo la información de la señal y no perder los bordes del corte, mas sin embargo para el cálculo de la envolvente no se utilizó superposición debido a que la variación en la envolvente resultante era despreciable. Finalmente cada muestra o ventana se definió de 50ms porque presento mejores resultados en la resolución del ventaneo.

Se implemento el recorrido por toda la señal de audio mediante programación dinámica y se obtuvo envolventes similares a la de la siguiente grafica:

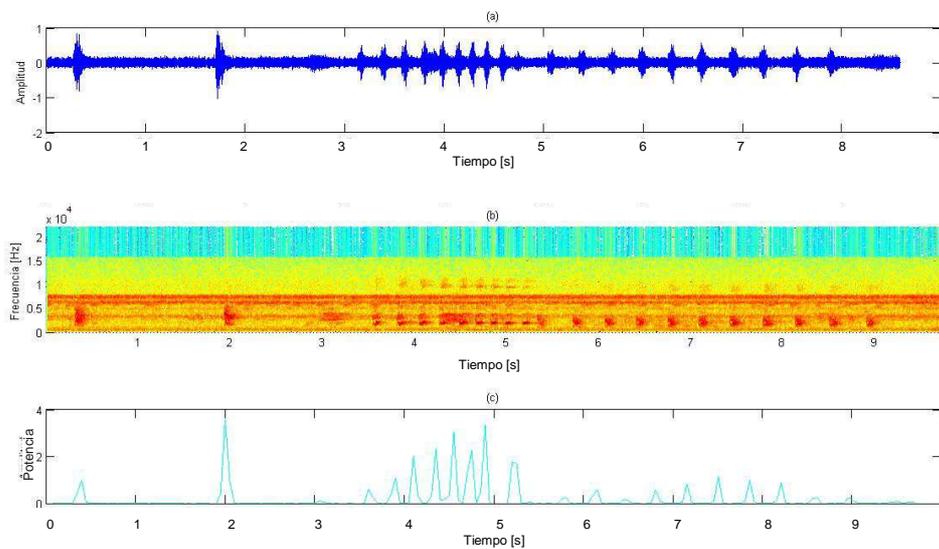


Figura 9. Obtención de envoltura de la señal (a), cuyo espectrograma (b) revela la presencia de ruido de grillos cerca de los 8kHz. (c) Envoltura  $V$  de potencia de la señal

De las gráficas se puede apreciar que en la grabación no únicamente hay ruido de fondo muy marcado, sino que inclusive se aprecia la marca constante que dejan los grillos en el registro cerca de los 8 KHz, mas sin embargo el procesamiento con PSD resulta prácticamente “inmune” a tales efectos y tras realizar el ventaneo constante se obtiene una envoltura que facilitará el corte del segmento de interés, debido a que apoyados en la premisa de decir que la grabación sobresale de alguna forma la especie que se desea identificar, se procede a encontrar el segmento a partir del máximo de la envoltura, del cual podemos decir encierra el sonido con mayor energía de la grabación, en el ejemplo se encuentra en la ventana número 40 con una magnitud de alrededor de 3.57 la cual es irrelevante siempre que sea el máximo de toda la envoltura.

A partir de este máximo se procede a descender por la gráfica tanto a la derecha como a la izquierda hasta encontrar el mínimo relativo y así identificar por la izquierda el inicio del segmento y por la derecha el fin del mismo.

La determinación para establecer que se ha encontrado el mínimo relativo se basa en dos criterios a los cuales se llegó después de encontrar algunos casos particulares:

- El primer criterio para decidir que se había encontrado un mínimo fue establecer que si el valor de este era al menos de la decima parte del máximo es decir si la señal se había atenuado 1dB, caso en el cual se consideraba que se había encontrado un límite del segmento.

- El segundo criterio fue apoyado sobre contemplar la idea de que el primer criterio no se cumpla debido a la posible presencia de ruido que impida que el sonido se llegue a atenuar 1dB, y por tanto se determinó establecer cual era el umbral mínimo ( $\Omega$ ) de la señal, para lo cual se extrae el mínimo de la envolvente en potencia (que debe ser diferente de cero), y así se busca que el mínimo local sea por lo menos inferior que el doble del mínimo  $\Omega$  de la envolvente.

Con estos criterios se fijaron umbrales “dinámicos”, que dependían de la calidad de ruido presente en el entorno y así entonces si el ruido crece, el umbral también crece.

De acuerdo con los criterios anteriores, para el ejemplo se obtuvo un corte del segundo 1.9 hasta 2.1, que para este caso resulta ser un segmento con una muy buena aserción en la elección del corte.

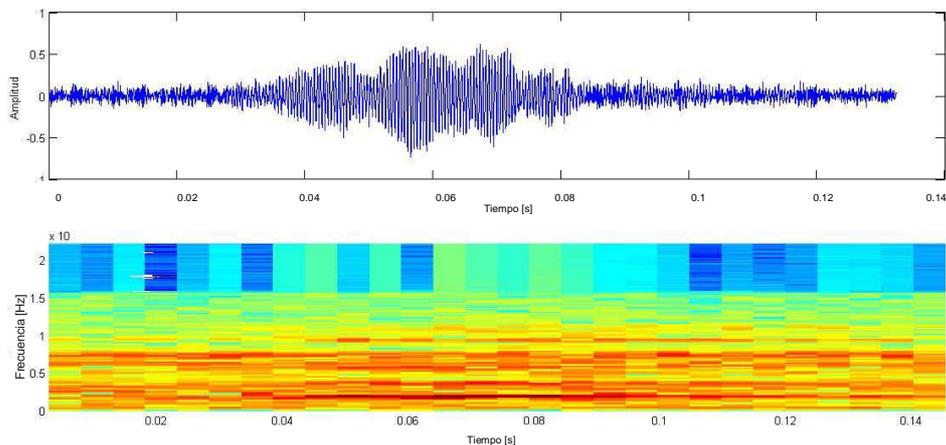


Figura 10. Aplicación algoritmo de corte.

El algoritmo funciona siempre y cuando la grabación tenga una calidad aceptable, en algunos casos se encontraron grabaciones en donde el algoritmo identificaba el ruido característico que se presenta al golpear el micrófono, que por realizarse de tan cerca es lógico que contenga mucha energía de un solo golpe, este tipo de ruidos accidentales deben ser eliminados de la grabación para no entorpecer el corte del segmento de interés.

También se encontró que en ocasiones tras extraer el segmento, se presenta un pequeño silencio tanto al principio como al final, estos silencios se deben a que la resolución que se utilizó para el ventaneo es de 50ms, y por tanto los silencios no superan ese intervalo, se podrían eliminar aumentando la resolución mas sin embargo se encontró que a intervalos menores, se producían cortes a mitad de una nota, con lo cual se perdía mucha información por tratarse de una nota incompleta y por esa razón se optó por dejar la resolución en 50ms.

## **5. DESARROLLO DE LA APLICACIÓN**

En los capítulos anteriores se han estudiado los diferentes métodos para el análisis de las características propias de cada especie, se han comparado resultados de algunos procesos, se ha hecho la revisión sobre redes neuronales, en el presente capítulo se explica cómo se desarrolla la aplicación que quedaría en el grupo de investigación GAICA, se tuvo en cuenta algunos aspectos importantes como lo fue que el sistema sea flexible a nuevas especies, sea capaz de leer archivos MP3, tenga la posibilidad de manejar diferentes bases de datos todo con el fin de facilitar el manejo a los investigadores del grupo.

Se decidió hacer una aplicación en MATLAB, debido a la facilidad que este representa en cuanto al manejo de redes neuronales, operación de matrices despliegue de datos, y manejo de las funciones matemáticas entre otras.

### **5.1 SISTEMA RECONOCIMIENTO MATLAB**

En el siguiente diagrama se hace una representación de la estructura que soporta el sistema, se parte de una lectura del archivo a analizar, se ejecuta el algoritmo de corte si el usuario lo ha seleccionado y posteriormente se calcula la función PSD del canto para finalmente poderlo analizar mediante los códigos de Correlación, Redes Neuronales o Análisis de Componentes Principales

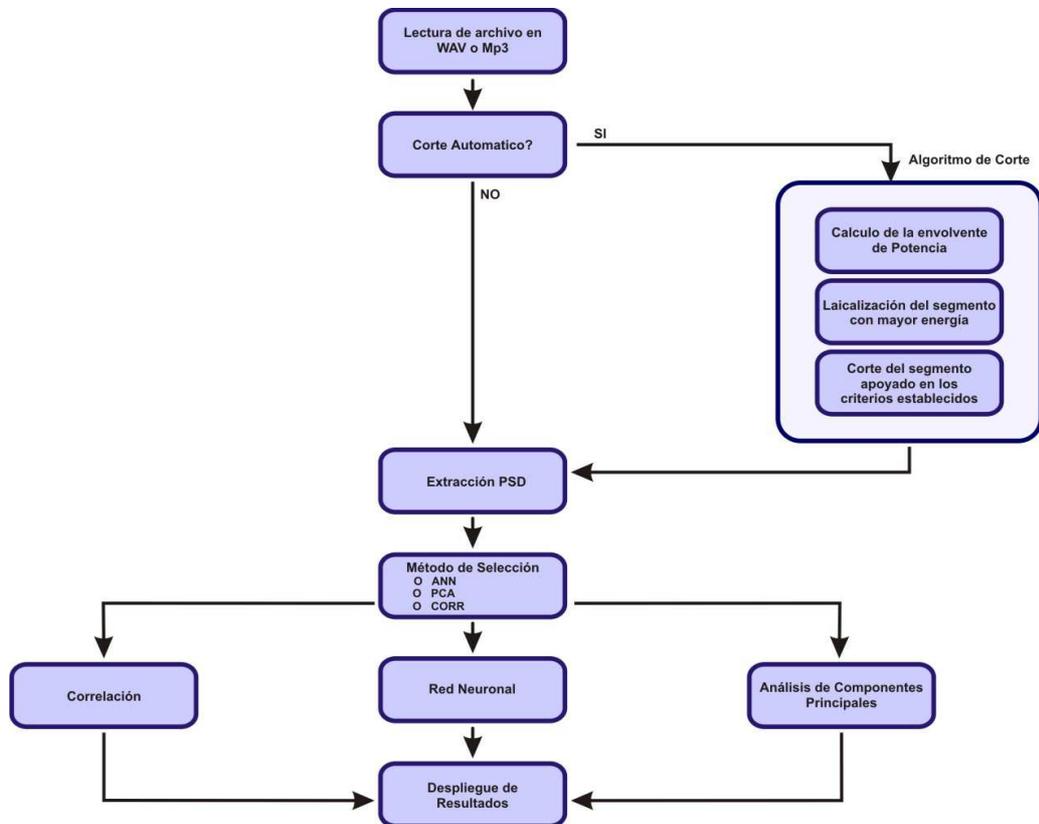


Figura 11. Diagrama de flujo aplicación en MATLAB.

### 5.1.1 Lectura de archivos WAV o MP3

Es la primera etapa en el procesamiento de las señales, y consiste en importar los archivos de sonido con las grabaciones de los cantos a identificar, para llevar a cabo la lectura de archivos WAV se contó con la facilidad de las herramientas de MATLAB, quien cuenta con el comando *wavread* para la importación de tales archivos, mientras que para la lectura de los archivos MP3 se acudió a la función *mp3read* desarrollada para MATLAB por Dan Elis quien amablemente ha dispuesto de ella en el siguiente portal<sup>4</sup>.

### 5.1.2 Algoritmo de corte

Como se puede observar del esquema desarrollado en el sistema, la decisión de usar el algoritmo de corte es tomada directamente por el usuario, se quiso

<sup>4</sup> Ellis, Dan. Mp3read and mp3write for Matlab. [artículo de internet]. <http://labrosa.ee.columbia.edu/MATLAB/mp3read.html> [Consulta 15 abril 2011]

desarrollar así ya que como se vio en el capítulo sobre el algoritmo de corte, este funciona sobre señales que presenten buenos niveles de calidad en la grabación con lo que se automatiza y agiliza la tarea de extraer el segmento de interés, mas sin embargo se puede presentar la situación en que la grabación no tenga dicha calidad o simplemente no se quiera realizar tal proceso de manera automática como lo manifestó el equipo de biología, por estas razones se decidió dejar esta etapa como opcional.

Si la opción de corte esta activa este realiza los procesos que se describieron en el capítulo de algoritmo de corte donde se extrae la potencia de la señal por segmentos en un proceso denominado ventaneo el cual arroja como resultado la curva envolvente de potencia de toda la señal, se localiza el máximo de esta curva y se hallan los segmentos de inicio y fin a partir de este máximo apoyado sobre los criterios de atenuación de la señal (Ver capitulo de algoritmo de corte).

### **5.1.3 Método de Comparación**

En esta etapa con el segmento seleccionado y habiendo extraído los patrones madre, nos encontramos con la decisión acerca de que método utilizar para la comparación de dichos patrones, basados en los resultados de los métodos estudiados y conociendo la efectividad que representa el uso de la red neuronal sería lógico que sea ese el método por defecto y sin lugar a decisiones, pero si bien es el método con mejor desempeño también tiene la dificultad de el entrenamiento previo que necesita, el cual solamente se puede realizar en MATLAB, pues el Toolbox de entrenamiento es soportado únicamente en la plataforma, no en aplicaciones creadas por MATLAB, por esta razón se decidió de todas formas dejar el algoritmo de comparación por PCA en la aplicación, ya que si bien no es el más exacto, cuenta con la ventaja de que no necesita entrenamiento, simplemente se crea la base de datos y queda listo para comparar. En tanto al método de comparación por correlación es el menos exacto pero se lo quiso conservar debido a que tampoco necesita entrenamiento y porque es uno de los métodos que utilizan los biólogos tradicionalmente para la comparación de semejanzas entre determinadas especies.

### **5.1.4 Despliegue de Resultados**

De acuerdo al resultado arrojado por el método escogido, el sistema organiza los “puntajes” para cada especie y así despliega en pantalla las fotografías de los probables individuos en orden del más al menos semejante, con una respectiva muestra de su canto.

## **5.2 CREACIÓN DE BASE DE DATOS**

Como se ha descrito durante el desarrollo se tuvo en cuenta que el sistema sea flexible a nuevas especies para poder organizar nuevas bases de datos, y para lograr tal objetivo se implemento un sistema guiado para la creación de la base de

datos con nuevos patrones de muestras escogidas por el usuario, así entonces para organizar una nueva base de datos se debe seleccionar los cantos que servirán de apoyo (filtrados y editados) y organizarlos en determinada jerarquía como muestra la siguiente figura:

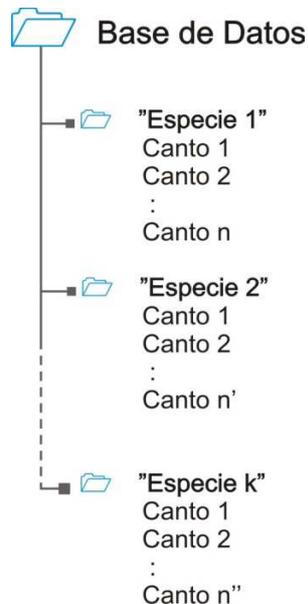


Figura 12. Organización de carpetas con especies y respectivos cantos

Es importante que el nombre de la especie sea igual al nombre de cada carpeta, pues es de ahí de donde el sistema toma como referencia las especies que contendrá dicha base de datos, en tanto al nombre de cada grabación en las carpetas puede ser cualquiera, pues lo único importante en dichos archivos es que estén en formato WAV (PCM) o MP3. Una vez se tiene organizada la base de datos por carpetas se procede a la extracción automática de los patrones por parte del sistema, que en este caso solicita la ubicación de la carpeta que contiene a las especies, realizando un recorrido por todos los archivos y extrayendo de cada uno el patrón característico, es importante resaltar que para este proceso el sistema también tiene en cuenta la opción de corte automático, pues dicho algoritmo también se puede usar para el corte en esta etapa de creación de la base de datos.

Mientras el sistema recorre cada uno de los archivos y los procesa, se puede ir apreciando en pantalla cada uno de los patrones extraídos, con su respectivo espectrograma, hasta que acaba con todas grabaciones solicita al usuario un destino para guardar los resultados del proceso, el archivo generado es de extensión .mat y contiene los nombres de las especies, una matriz de comparación "input" y una matriz de referencia "target".

Adicional a la carpeta de la base de datos, se debe crear una carpeta “Muestrario” que contenga un ejemplo del canto de cada animal acompañado de una fotografía del mismo, cada uno con el mismo nombre que figura en las carpetas (con la única diferencia en la extensión del archivo), preferiblemente la fotografía debe estar en formato JPG, mas sin embargo el sistema es capaz de procesar archivos BMP, GIF, ICO, JPEG, PGM, PNG y PBM, mientras que la grabación de muestra puede estar en formato WAV o MP3, esta carpeta es usada por la aplicación al momento del despliegue de resultados, en donde nos muestra la foto y nos permite escuchar el sonido que figuren con el nombre de la especie en la carpeta “Muestrario”.

### 5.3 INTERFAZ GRÁFICA DEL SISTEMA

En el capítulo anterior se definieron los procesos y métodos a implementar en el sistema, por tanto en el presente capitulo nos concentraremos en el desarrollo de una interfaz amable para el usuario.

La ventana principal luce como se muestra en la siguiente imagen:

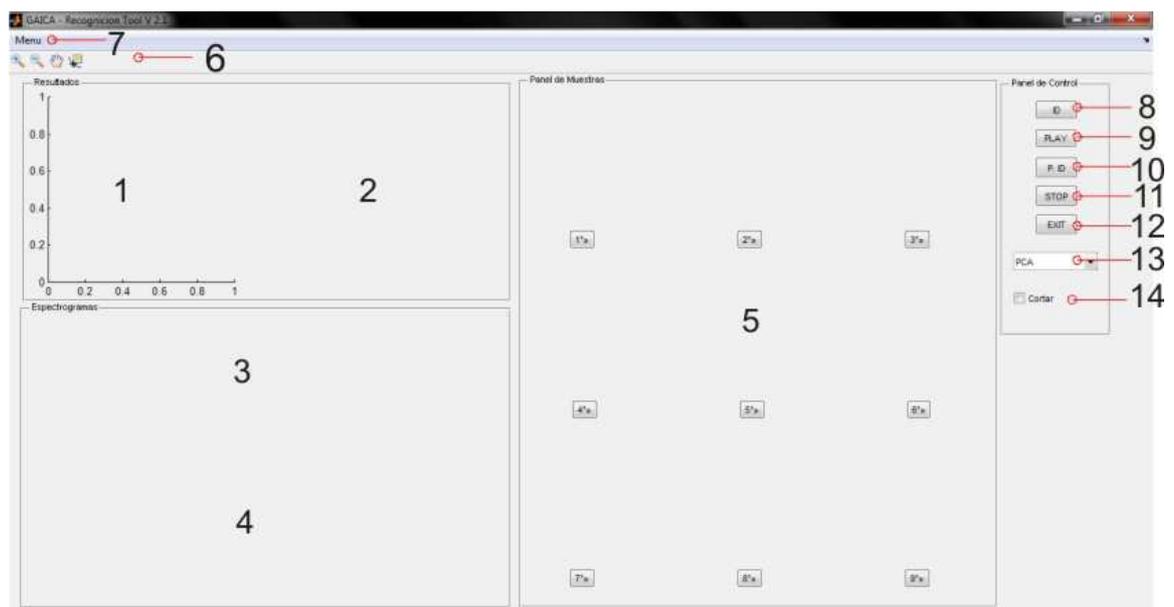


Figura 13. Interfaz de usuario principal.

- 1: Diagrama de barras del porcentaje de semejanza para cada especie.
- 2: Gráficos de los patrones PSD de la grabación y del “Muestrario”.
- 3: Espectrograma de la grabación analizada.
- 4: Espectrograma de la grabación del “Muestrario”.
- 5: Área de Muestras.
- 6: Barra de herramientas.
- 7: Menú.

- 8: Botón de identificación.
- 9: Play.
- 10: Reproducir segmento identificado.
- 11: Stop.
- 12: Salir.
- 13: Selección de método.
- 14: Opción de corte automático.



Figura 14. Opciones del menú.

- 15: Cargar base de Datos.
- 16: Crear Base de Datos.
- 17: Especies en la Base de Datos.
- 18: Seleccionar carpeta del Muestrario.

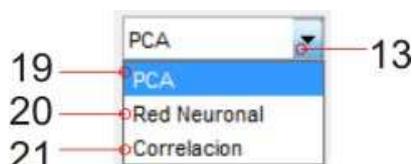


Figura 15. Menú escoger método de comparación.

- 19: Comparación por componentes principales.
- 20: Selección por Red Neuronal.
- 21: Comparación por Correlación.

### 5.3.1 Identificación de una Especie.

Las especies con las que se entreno una de las redes neuronales y con las que se hicieron pruebas son las del siguiente listado:

- [1] *Accipiter bicolor*
- [2] *Ammodramus aurifrons*
- [3] *Currucutú O. choliba*
- [4] *Zonotrichia capensis*
- [5] *Perissocephalus tricolor*
- [6] *Lipaugus vociferans*
- [7] *Pitangus lictor*
- [8] *Anisognathus somptuosus*

[9] *Tyrannus melancholicus*

[10] *Eremophila alpestris*

Para hacer funcionar el sistema lo primero será cargar la base de datos, para hacerlo nos dirigimos a Menú>> Cargar B.D.



Figura 16. Opción cargar base de datos.

E inmediatamente nos pide seleccionar la base de datos que deseamos cargar:

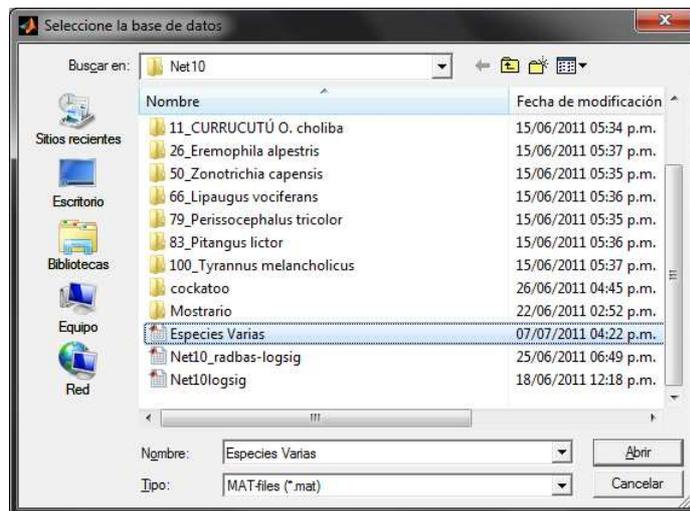


Figura 17. Directorio seleccionar base de datos.

Una vez seleccionada la B.D. nos despliega una ventana informativa con las especies que contiene el archivo:

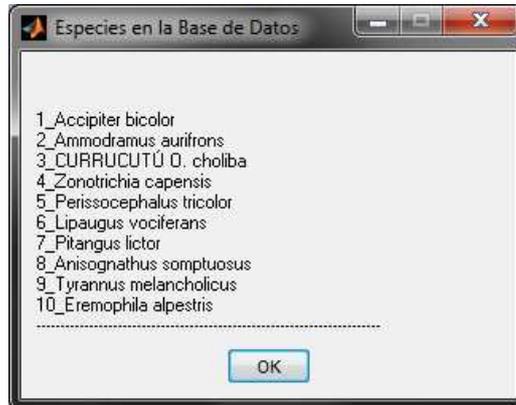


Figura 18. Especies en base de datos.

En tanto pulsamos OK el sistema carga dicha base de datos y asume que en la misma locación se encuentra la carpeta Muestrario de dicha base de datos, recordemos que tal muestrario debe contener un ejemplo del canto con el que se está comparando la señal y una foto del ejemplar. Si dicha carpeta tiene un nombre distinto o se encuentra en algún otro lugar del ordenador, entonces después de pulsar ok nos dirigimos a Menú>> Seleccionar Muestrario de donde emerge una ventana solicitando dicha carpeta:

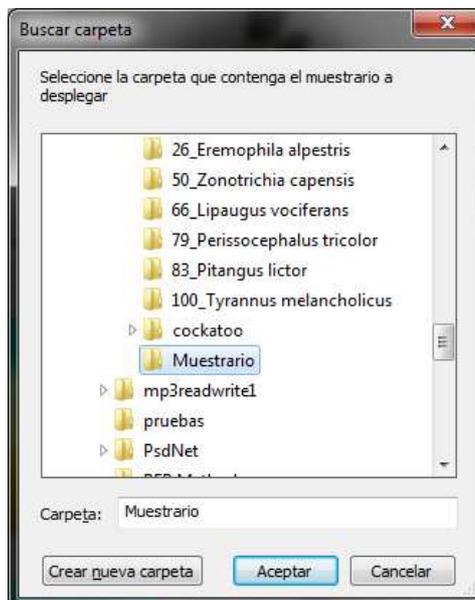


Figura 19. Seleccionar directorio muestrario.

Seleccionamos el muestrario y se oprime Aceptar. Ahora el sistema está listo para empezar a identificar los diferentes sonidos, para hacerlo seleccionamos el método a emplear y nos dirigimos al botón "ID" donde escogemos el track que contiene la grabación a identificar:

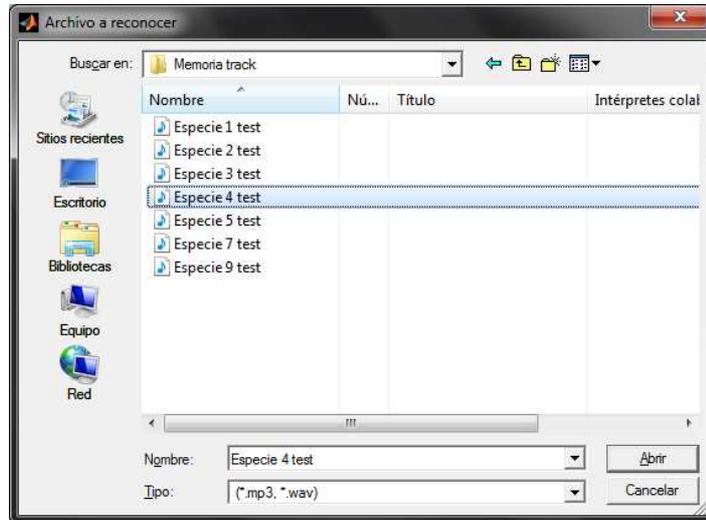


Figura 20. Ventana escoger track a reconocer.

Una vez seleccionada la grabación el sistema empieza a procesar la información y en instantes arroja sus resultados:

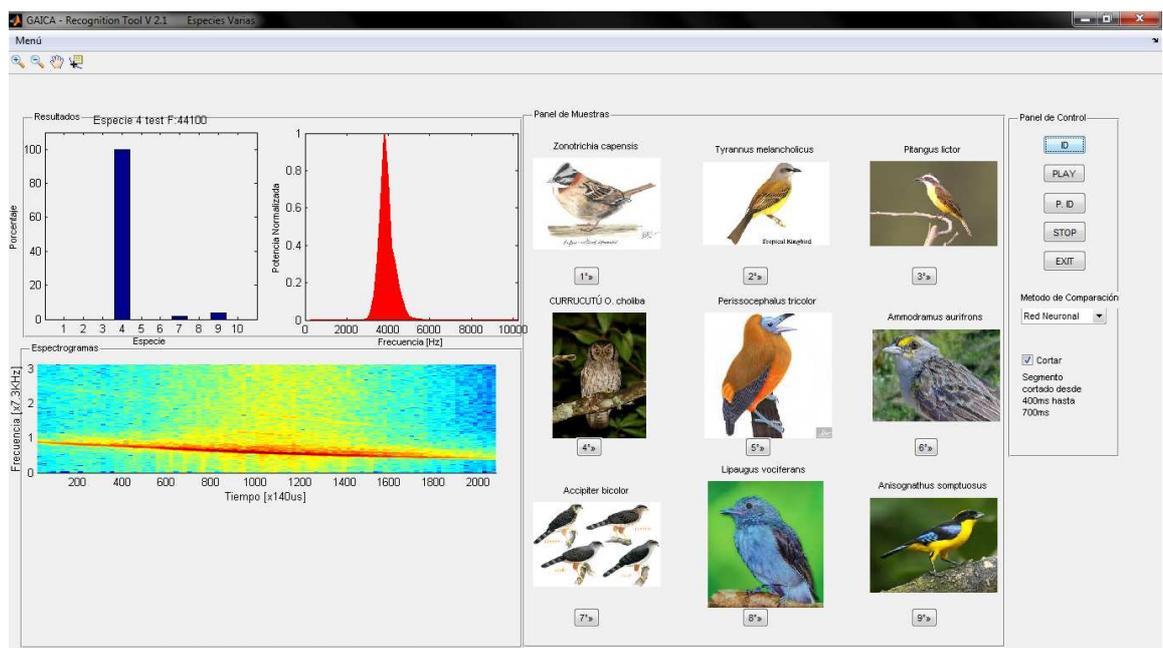


Figura 21. Resultados procesamiento track

Como se puede observar la aplicación organiza los resultados y los despliega en el panel de muestras, además se puede apreciar el patrón de la grabación en una grafica sombreada de color rojo y en el panel de espectrogramas vemos la sección de la grabación que decidió cortar, para el ejemplo el sistema recorto una sola nota del trino y nos informa que corresponde al segmento de 400ms hasta 700ms.

En el panel de muestras se puede apreciar que el candidato más probable es el “*Zonotrichia capensis*” y al pulsar sobre el botón ubicado bajo su fotografía el sistema nos reproduce un ejemplo de su canto (el cual está definido en la carpeta de “Muestrario”), nos grafica el patrón característico en color cian sobre el patrón rojo de la grabación identificada y su respectivo espectrograma en el panel de espectrogramas para ayudar en la comparación e identificación:

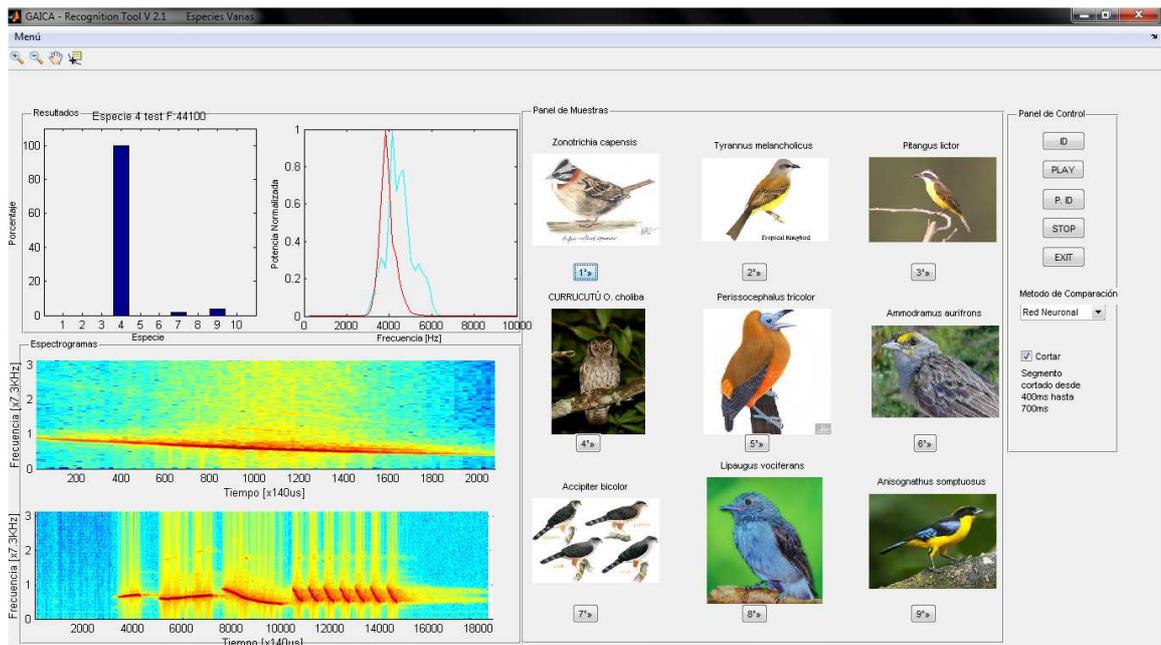


Figura 22. Resultados del proceso de identificación y reproducción del canto.

### 5.3.2 Creación de una nueva Base de Datos.

Como se vio en el ejemplo el sistema es capaz de identificar las especies siempre y cuando exista una base de datos, como ya se ha explicado la base de datos con la que se trabajo fue creada enteramente en MATLAB, y se quiso implementar una forma de crear nuevas bases de datos asistidas, para poder realizarlo partimos de la organización de los archivos en la forma jerárquica que se describió en un capítulo posterior y una vez este organizada y editada nos dirigimos al botón Menú>>Crear Nueva B.D., en donde nos solicitara la carpeta que contiene las carpetas con las grabaciones, hay que tener en cuenta en esta etapa del transcurso que la opción de corte es tenida en cuenta para el proceso de las muestras, es decir la decisión de cortar las muestras también se tiene en cuenta para las muestras de las carpetas:

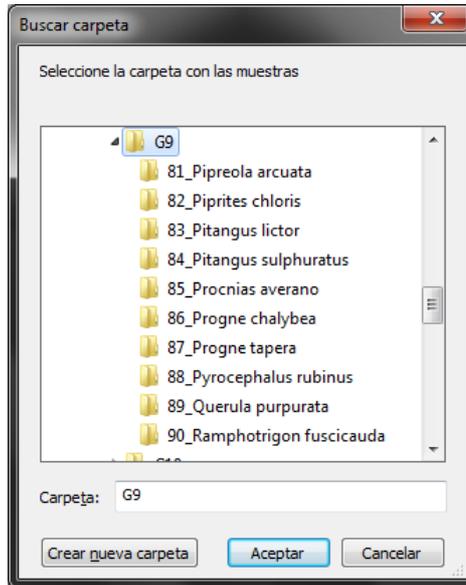


Figura 23. Directorio muestras para nueva base de datos.

El número de especies en dicho directorio puede ser cualquiera, pues el sistema está en capacidad de manejar cualquier número, mas sin embargo la efectividad que se obtuvo en el presente trabajo fue en pruebas realizadas con 10 especies diferentes, por tanto si se desea manejar volúmenes más grandes los más seguro es que se vea reflejado en una disminución en la efectividad de reconocimiento. El sistema empezará el análisis de los archivos e ira desplegando en pantalla tanto el espectrograma como el patrón PSD de cada archivo hasta acabar con todos los archivos, y en ese momento solicitara el nombre y destino donde se guardara dicha Base de Datos:

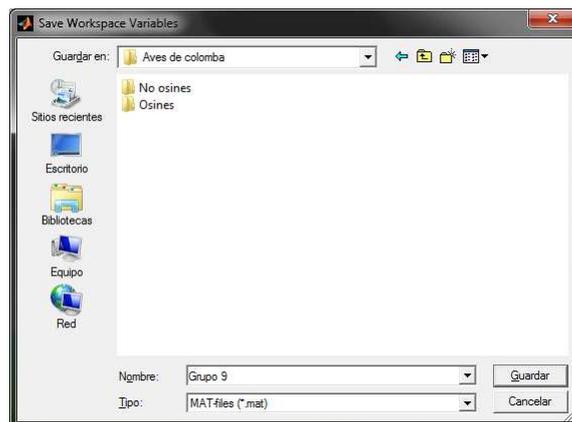


Figura 24. Directorio destino base de datos.

El archivo generado posee una extensión .mat y es el archivo que se debe cargar en la aplicación.

Para cargar la nueva Base de Datos nos dirigimos a Menú>>Cargar B.D. y seleccionamos el archivo que se acabo de generar:

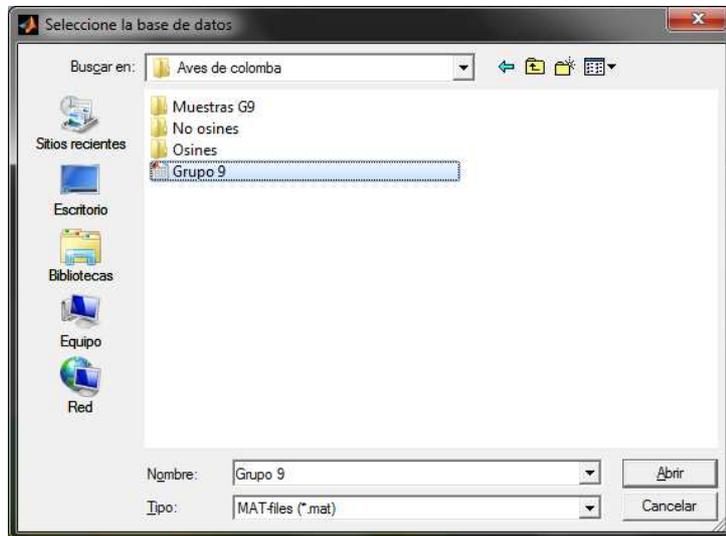


Figura 25. Directorio cargar base de datos.

Como lo vimos en un ejemplo pasado el sistema nos informa de las especies que se encuentran en dicho archivo:



Figura 26. Especies encontradas en base de datos.

Y al pulsar OK el sistema asume que la carpeta “Muestrario” que contiene las imágenes y cantos de ejemplo a desplegar en el panel de Muestras se encuentra en la misma ubicación donde está la BD, es importante resaltar que cada imagen y grabacion en esta carpeta debe tener el mismo nombre con el que figura en la carpeta de la base de datos (a excepción de la extensión por supuesto). Si la carpeta muestrario está ubicada en un lugar diferente, le informamos al sistema la ubicación de esta mediante Menú>> Seleccionar Muestrario:

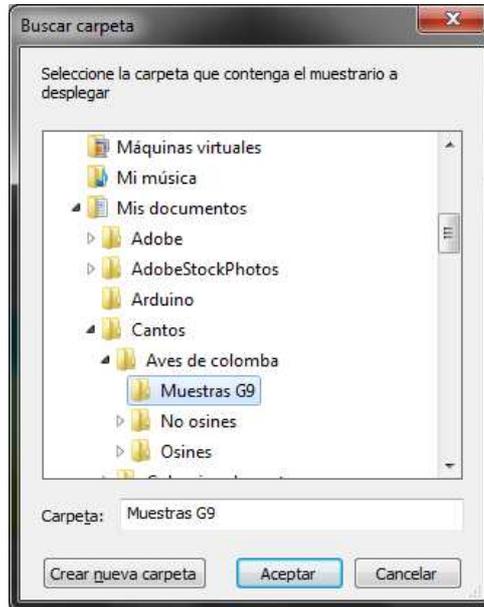


Figura 27. Directorio seleccionar muestrario.

Y el sistema está listo para empezar a identificar patrones; debido a que MATLAB no soporta el toolbox de entrenamiento de Redes Neuronales en Aplicaciones Stand Alone, la base de datos creada no posee una Red neuronal y por tanto las identificaciones que se realicen se podrán hacer únicamente por los procesos de PCA y Correlación. Para el ejemplo creado se obtuvo este tipo de resultado por identificación basado en PCA:

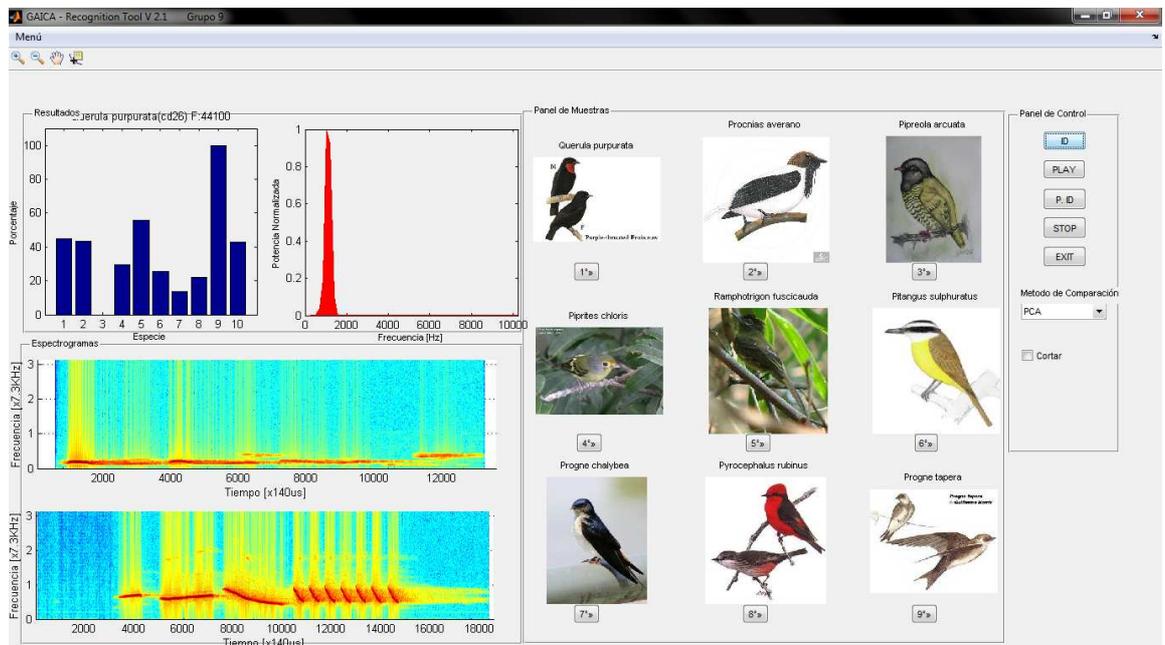


Figura 28. Resultados con nueva base de datos.

## 6. IMPLEMENTACION DEL SISTEMA EN MEDIOS EMBEBIDOS

Uno de los objetivos del proyecto es la implementación del sistema en un medio embebido, el cual se había contemplado a razón de la facilidad que tendría el transporte de este dispositivo en las salidas al campo donde el acarrear con portátiles y equipo robusto resulta en ocasiones poco conveniente por las diferentes dificultades que se encuentran en el medio ambiente.

Para la implementación del sistema se partió del hecho de que el grupo cuenta con el equipo de muestreo para la grabación y registro de los distintos sonidos, el equipo de grabación cuenta con micrófonos direccionales de alta definición, forros especiales, y un mando central Marantz PMD-660, capaz de almacenar las grabaciones en una tarjeta Flash Card con capacidad de 4Gb, las grabaciones son registradas en formato WAV con una frecuencia de muestreo de 44KHz.

Partiendo de lo anterior se decidió que se aprovecharía la etapa de muestreo y filtrado hecha por la grabadora, y el dispositivo se centraría en saber leer e interpretar dichos archivos dejados en la memoria. Para empezar se decidió hacer uso de la tarjeta de desarrollo Domino FEZ la cual

- Cuenta con USBizi Chipset, que es un procesador capaz de ejecutar código compilado en .Net
- Cuenta con el socket para la conexión con tarjeta micro SD
- Procesa operaciones en 32bit
- Posee entradas y salidas tanto análogas como digitales.
- Debugger
- Usb Host y device entre otras.

La programación de esta tarjeta se realiza en Microframework, el cual contiene un lenguaje reducido de C# que facilita el procesamiento de la información.

Debido a las limitantes que presenta en tanto a memoria, todo el proceso se realiza de manera iterativa, tomando pequeños bloques de información, procesándola y guardando el resultado parcial hasta acabar con el total del archivo que por la naturaleza y calidad de grabación no son archivos pequeños.

Un esquema simplificado de la conexión realizada en la tarjeta se muestra en la siguiente figura:

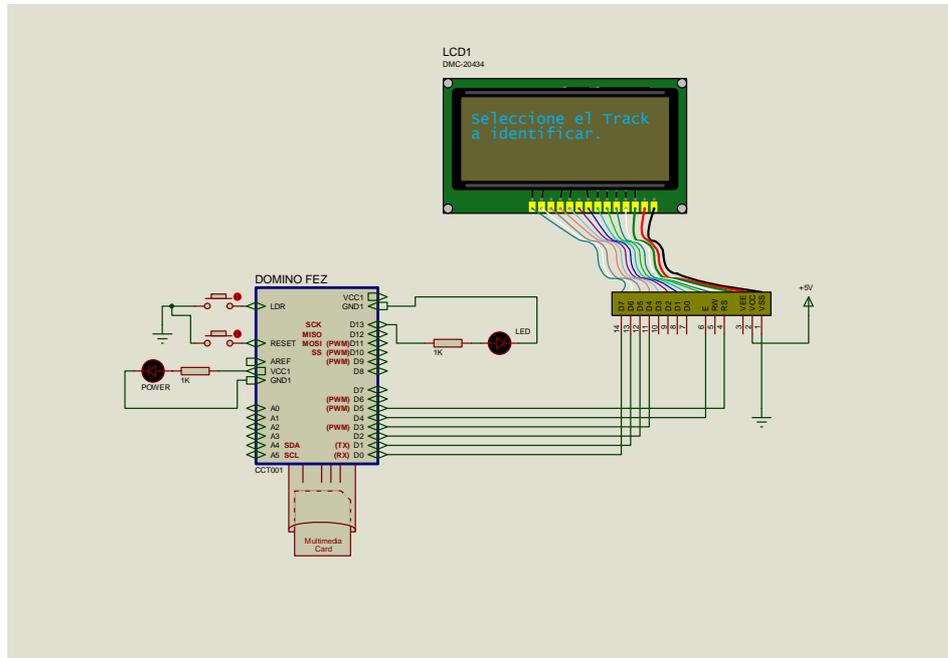


Figura 29. Conexión de Pantalla y Memoria al sistema emputrado

### 6.1 SISTEMA RECONOCIMIENTO MEDIO ENBEDIDO

La figura 30. presenta un diagrama de bloques con los diferentes procesos que realiza el sistema para procesar una grabación:

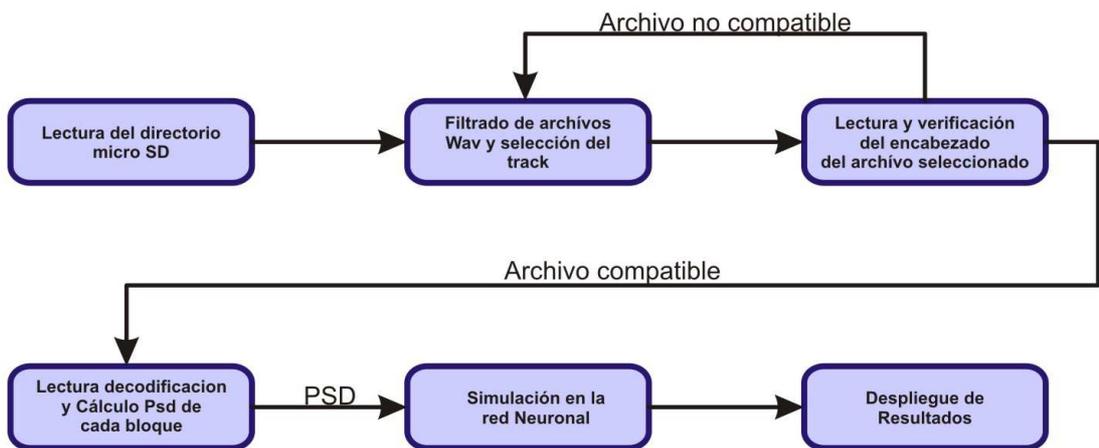


Figura 30. Diagrama de flujo embebido.

### 6.1.1 Lectura del directorio en la memoria micro SD

Para empezar con el proceso la tarjeta hace un montaje del sistema de archivos, es importante resaltar que la memoria debe estar en formato FAT16/FAT32, de lo contrario el dispositivo no puede trabajar, una vez verificado que el formato sea el correcto el sistema procede a la lectura del directorio raíz en donde se supone se encuentran las grabaciones de interés.

### 6.1.2 Filtrado de los archivos WAV y selección de la grabación

Una vez conocidos los archivos que están en la raíz, se prosigue con la selección de aquellos que están en formato WAV, previniendo abrir otras extensiones y evitar errores, acto seguido se procede al despliegue del nombre de las grabaciones disponibles por medio de la salida en la LCD con el fin de que el usuario seleccione la grabación que desea identificar mediante el botón LDR que posee la tarjeta, si los archivos se pasan el software volverá a repetirlos en forma cíclica hasta que el usuario seleccione uno.

### 6.1.3 Lectura y verificación del encabezado del archivo seleccionado

De antemano se sabe que se va a trabajar con archivos de sonido en formato WAV de 16 bits, sin embargo dentro de cada archivo WAV hay varias consideraciones a tener en cuenta, tales consideraciones vienen descritas por el encabezado que brinda información acerca de cómo viene compuesto el archivo en una estructura RIFF (Resource Interchange File Format - formato de fichero para intercambio de recursos). Que puede definirse como una estructura de archivo para ficheros de recursos multimedia.

Tabla 1. Formato archivo de audio.

Posición	Ejemplo	Descripción
1-4	"RIFF"	Marca el archivo como un archivo riff. Caracteres son cada uno de 1 byte de largo.
5-6	Tamaño	Tamaño del archivo general - 8 bytes, en bytes (entero de 32 bits)
7-12	"WAVE"	Tipo de archivo de encabezado del tipo de archivo - debe decir WAVE
13-16	"fmt "	Marcador de formato "Chunk"- incluye nulo al final
17-20	16	Longitud de los datos
21-22	1	Tipo de formato (PCM=1)
23-24	2	Número de canales
25-28	44100	Frecuencia de muestreo
29-32	176400	Bytes por muestreo
33-34	4	Bloque de alineación
35-36	16	Bits por muestra
37-40	"data"	Marca el principio de la sección de datos
41-44	tamaño datos	Tamaño de la sección de datos
Muestras dadas en bloques de 16 bits		

Cabecera RIFF: El fichero RIFF lleva siempre una cabecera de 8 bytes, que identifica al fichero y especifica la longitud de los datos a partir de la cabecera (esto es, la longitud total menos 8).

Esta cabecera se compone de 4 bytes con el contenido "RIFF" y los otros 4 indican la longitud. Después de la cabecera RIFF siempre hay 4 bytes que identifican el tipo de los datos que contiene; para el caso .WAV estos 4 bytes contienen "WAVE"<sup>5</sup>.

Longitud de los datos: Tamaño en que viene cada muestra, las grabaciones manejadas por el equipo son de 16bit.

Categorías del formato WAVE: La categoría se especifica a través del campo **Tipo de formato**. Los campos específicos del segmento de formato y la representación de los datos en el segmento de datos dependen de este valor.

Actualmente se han definido los siguientes formatos no propietarios:<sup>6</sup>

Formato	Valor	Descripción
WAVE_FORMAT_PCM	0x0001	Microsoft Pulse Code Modulation (PCM)
FORMAT_MULAW	0x0101	IBM mu-law format
IBM_FORMAT_ALAW	0x0102	IBM a-law format
IBM_FORMAT_ADPCM	0x0103	IBM AVC Adaptative Differential PCM format

Para el sistema embebido trabajaremos con archivos PCM (sin compresión), ya que es el formato que trabaja el equipo de grabación del grupo.

Número de canales: Para grabaciones monofónicas es igual a 1, estéreo 2 etc. el sistema es capaz de leer grabaciones con múltiples canales pero trabaja con un solo canal.

Frecuencia de muestreo: Frecuencia con la que se grabó la señal de audio, el grupo maneja grabaciones a 44Khz, y el sistema se calibra para trabajar a esa frecuencia.

Como se puede apreciar algunas variaciones en el formato son soportadas por el sistema, mas sin embargo hay algunas que son específicas como lo es por ejemplo la compresión utilizada, en donde solo se maneja tipo PCM sin compresión, si existe alguna discrepancia en la verificación de la cabecera, el sistema simplemente lanza el mensaje de error en pantalla y pide se seleccione una grabación compatible, por ejemplo si al leer el encabezado se encuentra que

---

<sup>5</sup> ARRIBAS, Juan Ignacio. Formato de audio digital. [artículo de internet]. [http://www.lpi.tel.uva.es/~nacho/docencia/ing\\_ond\\_1/trabajos\\_01\\_02/formatos\\_audio\\_digital/html/wavformat.htm](http://www.lpi.tel.uva.es/~nacho/docencia/ing_ond_1/trabajos_01_02/formatos_audio_digital/html/wavformat.htm) [Consulta 1 junio 2011]

<sup>6</sup> Ibid.

el formato es FORMAT\_MULAW el mensaje que se verá en pantalla es: "Debe ser PCM".

#### 6.1.4 Lectura decodificación y cálculo del PSD de cada bloque

Una vez seleccionado el archivo y habiendo comprobado la compatibilidad con el encabezado del mismo se procede a la lectura de este, debido a las limitaciones por memoria del sistema, la lectura se hace por bloques de manera similar al ventaneo, por tanto se cargan los primeros 512 bytes del archivo en el buffer de lectura de la tarjeta, se convierte los datos a 16 bits con signo tomando parejas de bytes del buffer siendo las posiciones impares las menos significativas (LSB) y las pares las más significativas (MSB), así por ejemplo para convertir las primeras dos posiciones a un numero de 16 bit con signo se operaria:

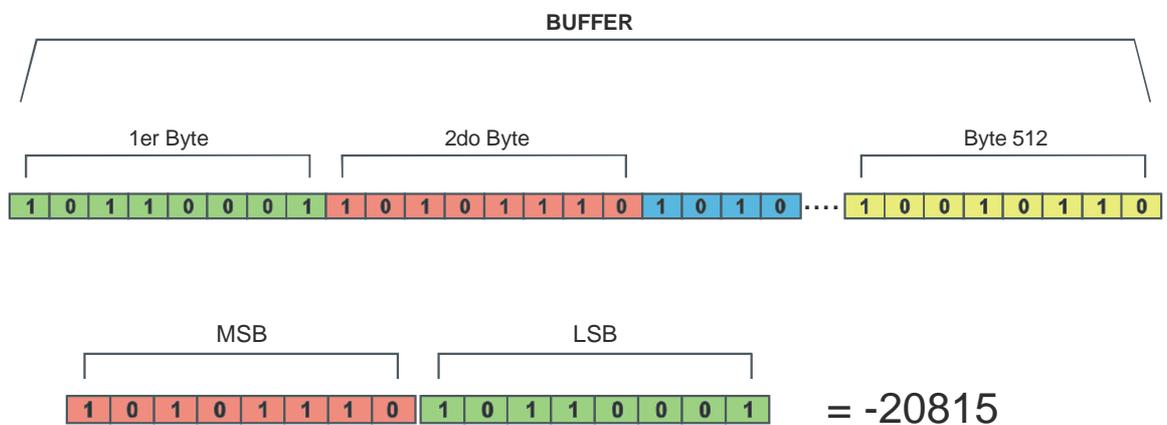


Figura 31. Conversión a 16 bits lectura SD.

Si la grabación es de un solo canal a 16 bits se llenarían los 256 datos necesarios para la transformada de Fourier en una cargada al búfer, mas sin embargo si se tratase de una grabación estéreo se haría necesario dos cargas al búfer ya que los primeros 2 bytes serian correspondientes al canal izquierdo (L) y los dos siguientes al canal derecho (R) y el sistema solo toma el primer canal, si la grabación tiene 4 canales se harían necesarias 4 cargadas al buffer y así sucesivamente hasta completar los 256 datos, una vez se completen las muestras se procede a multiplicarlas por la ventana tipo Bartlett previamente definida en un vector de longitud 256 que contiene los valores entre 0 y 1 con la forma de dicha ventana, estos valores fueron obtenidos de MATLAB y copiados a un arreglo o array en Microframework, para realizar dicha multiplicación simplemente se recorren las 256 posiciones del vector tomando el elemento n del arreglo de las muestras y multiplicándolo por el elemento n del arreglo ventana.

Cuando se complete dicha multiplicación se procede a realizar el cálculo del PSD de esa muestra, como se vio en el capítulo anterior, la función PSD se puede definir como la sumatoria de las trasformadas de Fourier al cuadrado, entonces la

forma más eficiente que se encontró para dicho cálculo fue usar el algoritmo para el cómputo de la transformada de Fourier del libro Numerical Recipes, el cual contiene un compilado de diversas funciones matemáticas implementadas en lenguaje C, y una vez obtenida la transformada se eleva al cuadrado tanto la componente real y la imaginaria de las primeras 60 posiciones (mas adelante veremos porque solo 60) para finalmente sumarlas y acumular su resultado.

El algoritmo de cálculo de la transformada de Fourier como ya se menciono esta implementado en lenguaje C y por tanto se hizo necesario la migración de este código al lenguaje C# soportado por la tarjeta.

Como era de esperarse, el algoritmo para el computo de la transformada de Fourier al ser un poco robusto por la serie de senos y cosenos que usa, ocupa un tiempo considerable en llevar a cabo su tarea, sin embargo fue el más eficiente en tiempo y recursos, ya que al hacer ensayos con otros algoritmos se encontraba que tardaban 5 y 6 veces más que el descrito en Numerical Recipes. Existen varios algoritmos que se deducen de la definición fundamental de Fourier en donde se puede notar el uso tanto de las funciones seno, coseno, ln y log, lo que conlleva a su notorio consumo en tiempo, mientras que el algoritmo descrito en Numerical Recipes usa una recurrencia trigonométrica que mejora notoriamente el tiempo empleado en el cálculo de la FFT.

Un redescubrimiento de FFT realizado por Danielson y Lanczos en 1942, proporciona una de las derivaciones más claras del algoritmo. Danielson y Lanczos demostraron que en una transformada discreta de Fourier de la longitud N se puede reescribir como la suma de dos transformadas discretas de Fourier, cada una de longitud N/2. Una de las dos se forma de los puntos pares de longitud N original, la otra con puntos de los números impares.

Lo maravilloso sobre el lema de Danielson y Lanczos es que dicha transformada puede ser utilizada recurrentemente. Reduciendo el problema de computar  $F_k$  (transformada de Fourier) a aquel de computar  $F_k^e$  (los componentes impares de la transformada) y  $F_k^o$  (los componentes pares) podemos hacer la misma reducción de  $F_k$  al problema de computar la transformada de los datos de entrada N/4 números pares y de los datos con números impares de N/4. Es decir podemos definir  $F_k^{ee}$  y  $F_k^{eo}$  para que sea la transformada discreta de Fourier de los puntos que son respectivamente pares-pares y pares-impares en las subdivisiones sucesivas de los datos.

En tanto se acaba con la transformada de dicho bloque se continua con la lectura del siguiente bloque en el buffer y se repite el proceso, el resultado se lo acumula en la sumatoria del array que contiene el PSD y al acabar de recorrer el archivo se obtiene el valor final del vector.

Como se menciono antes solamente se hace el cálculo para las primeras 60 posiciones ya que durante las diferentes observaciones se noto que las aves

presentaban registros hasta aproximadamente los 10Khz, lo que traducido en términos del vector constante del PSD arrojado en MATLAB de 129 datos, significaba que los valores a partir de la posición 58 aproximadamente hasta la posición 129 eran prácticamente igual a cero y por tanto se decidió despreciarlos con el fin de optimizar la el uso de los recursos.

Así que con el patrón PSD calculado solo restaba normalizarlo para que sea ingresado a la red neuronal, este proceso se implemento con un simple ciclo para encontrar el máximo y dividir todo el array entre dicho máximo y al final obtener un patrón con una amplitud máxima igual a 1.

Tras implementar dicho proceso se obtuvieron patrones como los de las siguientes figuras en donde se hace una comparación entre lo que arroja MATLAB y lo arrojado por la tarjeta:

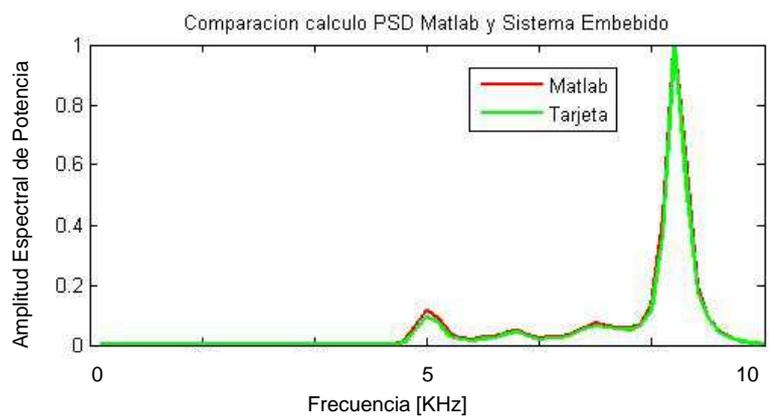
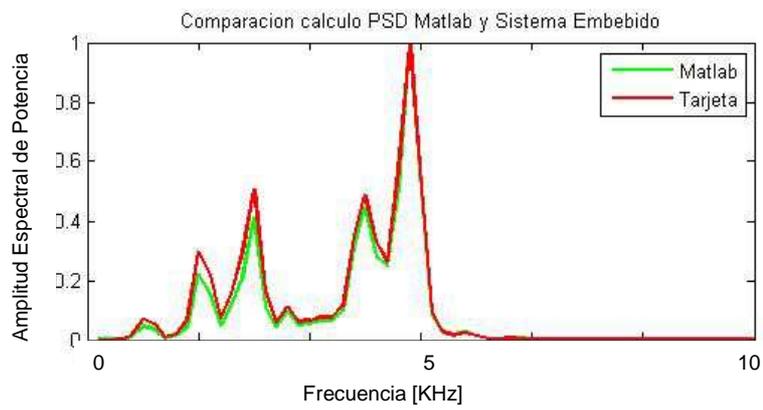
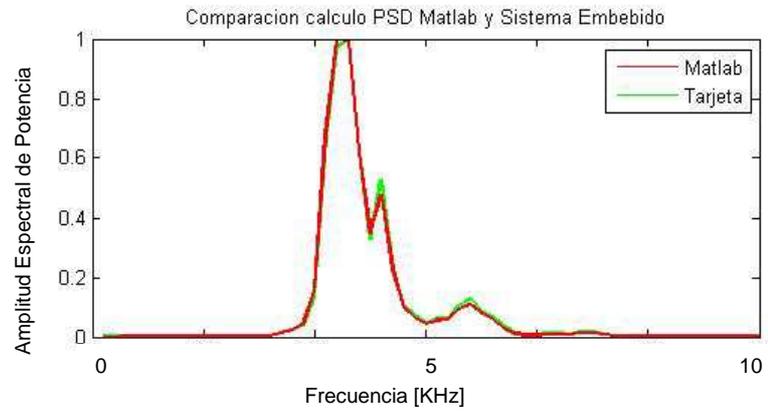


Figura 32. Comparación calculo PSD MATLAB y Sistema Embebido

Gráficamente se puede apreciar que hay una pequeña discrepancia en algunos de los picos que seguramente son producto de las diferencias entre los algoritmos usados para el cálculo de los patrones, no obstante se puede decir que el cálculo del PSD que realiza la tarjeta es del todo aceptable y en buena parte su semejanza con los resultados arrojados por MATLAB se debe a la precisión que maneja en sus operaciones, pues estas se realizan en su totalidad en 32bits lo que en contraparte se ve reflejado en la velocidad en que las ejecuta.

## 6.2 SIMULACIÓN EN LA RED NEURONAL

A este punto ya se había extraído el patrón de la densidad espectral de potencia y se hacía necesario el análisis de este, como se había observado en las practicas realizadas con MATLAB el sistema que mejor examinaba tales patrones fue sin dudas la red neuronal que se entreno con la base de datos, por tanto se decidió implementar dicha red ya entrenada en el sistema embebido,

Las operaciones que se llevan a cabo en una neurona se pueden representar en la siguiente figura:

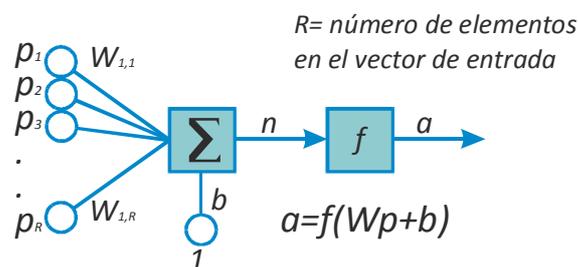


Figura 33. Esquema neurona.

Donde  $\mathbf{p}$  es el vector de entrada que contiene el patrón a identificar, para el caso de los cantos  $\mathbf{p}$  es el vector PSD extraído de la grabación y la longitud de este sería  $R = 60$ , cada elemento del vector es multiplicado por cada peso  $W$  a la entrada del sistema y se suman todos los resultados, a toda esta sumatoria se adiciona un elemento más que no tiene que ver con el vector de entrada, es un elemento independiente llamado bias ( $b$ ) y es similar a tener un peso  $W = 1$ .

De toda esta sumatoria se obtiene como resultado un número  $X$  del cual se obtendrá el valor  $f(X)$  para determinada función de activación que en este caso es la función tangente sigmoidea descrita por la siguiente ecuación:

$$a = f(x) = \frac{2}{1 + e^{-2x}} - 1 \quad (17)$$

Al unir varias neuronas se obtiene una capa y a la salida de cada capa se tendrá un vector conformado por las salidas de cada neurona  $a_1 a_2 a_3 \dots a_s$  donde  $s$  es el número de neuronas en la siguiente capa, en la siguiente figura se ilustra cómo está conformada una capa:

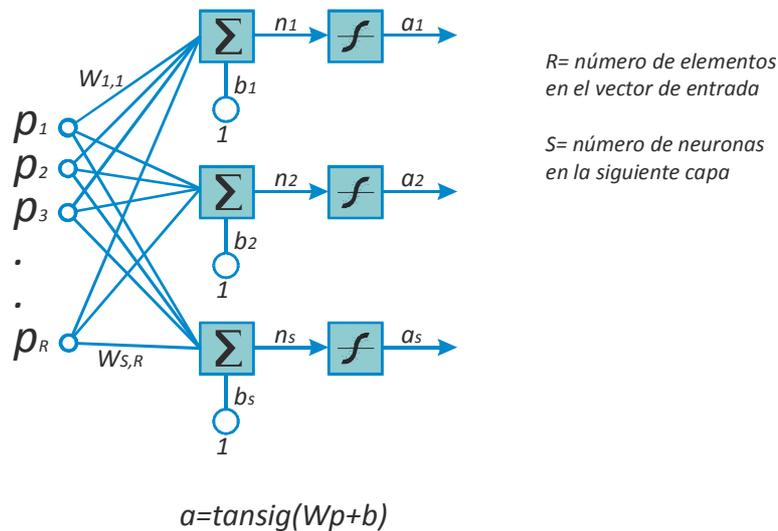


Figura 34. Representación de una capa de la red neuronal.

Es importante notar que los números  $S$  y  $R$  no necesariamente deben ser iguales, ya que  $R$  representa la longitud del vector columna de entrada y  $S$  el número de elementos o neuronas en la capa oculta, se podría decir que el vector de  $p$  entra a la red a través de la matriz de pesos  $W$ .

$$W = \begin{bmatrix} w_{1,1} & w_{1,2} & \dots & w_{1,R} \\ w_{2,1} & w_{2,2} & \dots & w_{2,R} \\ \vdots & \vdots & \ddots & \vdots \\ w_{S,1} & w_{S,2} & \dots & w_{S,R} \end{bmatrix} \quad (18)$$

Donde todos los pesos  $w_{i,j}$  representan la conexión existente entre el patrón de entrada  $p_j$  y la neurona  $i$ . De esta forma el vector  $n$  de longitud  $S$  se puede calcular como el producto matricial:

$$\mathbf{n} = W\mathbf{p} + \mathbf{b} \quad (19)$$

Donde  $b$  es el vector que contiene los *bias* que se conectan a cada una de las neuronas de la capa. Por lo tanto el vector de salida  $a$  se puede calcular como el producto matricial:

$$\mathbf{a} = f(W\mathbf{p} + \mathbf{b}) \quad (20)$$

La red neuronal que se entreno contiene una capa oculta con 20 neuronas, por lo tanto las operaciones en la siguiente capa son similares a las de la primera con una diferencia en que ahora el vector de entrada es el resultado obtenido en  $\mathbf{a}$  de la primera capa y la salida de esta capa serán los resultados de la red.

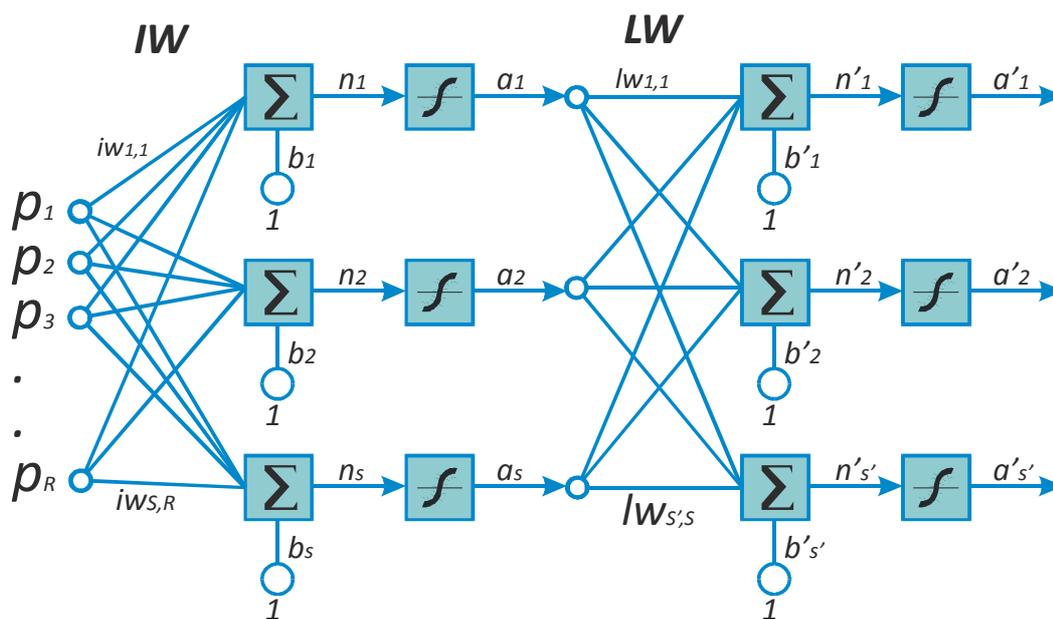


Figura 35. Representación de una red neuronal de dos capas.

Para nuestro caso particular las salidas  $a'_1, a'_2, \dots, a'_{s'}$  representan a cada una de las especies en donde  $s' = 10$  y la magnitud del valor a la salida representara la semejanza del patrón ingresado en el vector  $\mathbf{p}$  con cada una de las especies. Y el producto matricial total de las salidas se podría escribir:

$$\mathbf{a}' = f'(LWf(IW\mathbf{p} + \mathbf{b}) + \mathbf{b}') \quad (21)$$

Donde en este caso  $f$  es igual a  $f'$  que corresponde a la función de activación tangente sigmoidea de cada capa y  $\mathbf{a}'$  es el vector resultado arrojado por la red neuronal.

Por lo tanto para la implementación de la red neuronal en la tarjeta de desarrollo era necesario tener las matrices de pesos  $IW$  y  $LW$ , los vectores *bias*  $\mathbf{b}$  y  $\mathbf{b}'$  y con la correspondiente función tangente sigmoidea para así recrear la función (21), donde  $\mathbf{p}$  sería el vector PSD que ya se tiene calculado; para empezar mediante un script hecho en MATLAB se copiaron los pesos de cada capa de la red en un

arreglo vectorial ya que el lenguaje de la tarjeta no soporta estructuras multidimensionales o matriciales.

Una vez copiados los diferentes pesos de la red se implemento la simulación de cada capa que se resume en una multiplicación dinámica recorriendo cada fila y cada columna tanto de los pesos como de las entradas en un ciclo de 60 iteraciones para la primera capa y de 20 iteraciones para la segunda capa, estas iteraciones vienen dadas por el numero de neuronas en cada etapa seguidas de una sumatoria y finalmente una función de activación descrita en (21), pudiendo así simplificar la operatoria en tareas simplificadas.

Al comparar los resultados con los obtenidos en MATLAB satisfactoriamente los valores correspondían y no se observo variaciones de ningún tipo para las pruebas hechas a patrones idénticos, donde se ingreso vectores PSD tanto en el sistema embebido como a la red neuronal de MATLAB, el producto de este proceso fue analizado y descrito en el capítulo de resultados.

### **6.3 DESPLIEGE DE RESULTADOS**

Con el resultado arrojado por la red neuronal se procede a desplegar los valores en términos de porcentajes con 2 cifras significativas en el visualizador LCD, mostrando de forma secuencial cada especie y su porcentaje. Una vez acabado el despliegue de los resultados el sistema queda listo para el análisis de una nueva grabación repitiendo los pasos descritos.

## 7. RESULTADOS

En las secciones anteriores hemos hecho una revisión acerca de los diferentes métodos y conceptos involucrados en el desarrollo del sistema para el reconocimiento de aves, hemos encontrado ciertos patrones de semejanza en los cantos de cada especie y se decidió comparar patrones PSD de cada grabación por su bajo ruido y por no presentar desfase en los puntos de comparación, en esta sección se evalúan los resultados arrojados por los métodos empleados y se cualifica el sistema que mejor se desempeña haciendo un análisis de dichos resultados.

### 7.1 RECONOCIMIENTO DE 10 ESPECIES DIFERENTES

Para evaluar el desempeño de los algoritmos de identificación de los patrones se decidió crear dos bases de datos, cada una de 10 especies y con cada especie se realizaron 20 pruebas con cada algoritmo, para un total de 600 pruebas en cada base de datos.

Para la creación de la primera base de datos se seleccionaron las siguientes especies que conformarían el Grupo 1:

- *Accipiter bicolor*
- *Ammodramus aurifrons*
- *Otus choliba*
- *Zonotrichia capensis*
- *Perissocephalus tricolor*
- *Lipaugus vociferans*
- *Pitangus lictor*
- *Anisognathus somptuosus*
- *Tyrannus melancholicus*
- *Eremophila alpestris*

Cabe recordar que las grabaciones que se utilizaron para conformar la base de datos fueron todas previamente editados en corte y filtrado en el programa Adobe Audition 3.0, pues se requería de una señal en las mejores características para servir de referencia a nuevas grabaciones, un aspecto importante a la hora de editar las grabaciones de la base de datos es tener en cuenta las marcas notorias del medio como por ejemplo los grillos que pueden generar un pico notorio en el patrón si no son previamente editados.

Para las respectivas pruebas de los algoritmos se hacía necesario contar con grabaciones debidamente identificadas, y el banco de datos del grupo GAICA no

cuenta con un soporte para dicha labor, por lo tanto se acudió a librerías virtuales que prestan este servicio de referencia en internet en donde la comunidad de ornitólogos que cuentan con grabaciones identificadas suben sus archivos a la red para que las personas de todo el mundo puedan consultar estas grabaciones y tenerlas de referencia en tareas de identificación entre otras, las paginas que se uso de referencia fueron Macaulay Library y Xeno-Canto de donde se descargaron 20 grabaciones de cada especie y se editaron ya que las grabaciones están marcadas con frases que describen el ambiente en que se grabo e información del archivo; y a pesar de que el sistema filtra algunas componentes de bajas frecuencias es mejor recortar la voz humana para no entorpecer la identificación con sonidos que no corresponden a los de interés.

Debido a que si bien en algunos casos no se llegaba a la identificación exacta del individuo se quería por lo menos dar un acercamiento entre los primeros lugares, y por tanto se midieron dos aspectos importantes, el primero el grado de similitud que el algoritmo encuentre entre dos patrones y el segundo la posición que ocupa entre la lista de individuos, de estas dos medidas surgieron las siguientes tablas para cada algoritmo:

Tabla 2. Porcentaje de reconocimiento red neuronal

PORCENTAJE DE RECONOCIMIENTO RED NEURONAL																				PROMEDIO	
AVE #	I	II	III	IV	V	VI	VII	VIII	IX	X	XI	XII	XIII	XIV	XV	XVI	XVII	XVIII	XIX	XX	[%]
1	100	100	100	100	74,06	100	100	100	100	2,3	100	99,9	100	99,9	100	100	100	100	100	99,9	93,80
2	99,9	99,9	99,9	99,9	99,9	99,9	97	99,9	99,9	99,9	99,9	99,9	99,9	99,9	99,9	99,9	99,9	99,9	99,9	99,9	99,976
3	100	99,9	0	100	100	99,9	97,9	100	100	99,9	100	100	100	100	100	100	99,9	10,3	100	99,9	90,39
4	77,9	99	99,9	78,5	99,7	99,9	99,9	89,1	99,9	99,9	99,9	97,6	99,9	99,9	99,9	99,9	99,9	100	100	100	97,04
5	99,9	99,9	99,9	99,9	99,9	99,9	99,9	99,9	99,9	99,9	99,9	99,9	99,9	99,9	99,9	99,9	99,9	99,9	99,9	99,9	99,990
6	99,9	99,9	99,9	99,9	99,9	99,9	99,9	99,9	99,9	99,9	99,9	99,9	99,9	99,9	99,9	99,9	99,9	99,9	99,9	99,9	99,990
7	99,9	100	99,9	100	99,9	25,4	0,1	99,9	99,9	99,9	99,9	99,9	99,9	4,3	99,9	99,9	99,9	99,9	99,9	99,9	86,42
8	99,9	100	99,9	99,9	99,9	99,9	100	100	100	100	100	100	100	100	100	100	100	100	100	100	99,98
9	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	1,3	100	100	99,9	95,06
10	99,9	99,9	99,9	99,9	99,9	99,9	99,9	99,9	99,9	99,9	99,9	99,9	99,9	99,9	100	99,9	99,9	99,9	99,9	99,9	99,991
<b>TOTAL</b>																				<b>96,21</b>	
LUGAR OCUPADO EN LA RED NEURONAL																				ACIERTO [%]	
AVE #	I	II	III	IV	V	VI	VII	VIII	IX	X	XI	XII	XIII	XIV	XV	XVI	XVII	XVIII	XIX	XX	
1	1	1	1	1	2	1	1	1	1	3	1	1	1	1	1	1	1	1	1	1	94,17
2	1	1	1	1	4	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	96,25
3	1	1	9	1	1	1	2	1	1	1	1	1	1	1	1	1	10	1	1	1	88,56
4	1	1	1	1	1	1	1	2	1	1	1	1	1	1	1	1	1	1	1	1	97,50
5	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	100
6	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	100
7	1	1	1	1	1	2	7	1	1	1	1	1	1	2	1	1	1	1	1	1	90,71
8	1	1	1	1	1	1	1	1	1	1	1	1	1	1	2	1	1	1	1	1	97,50
9	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	10	1	1	1	1	95,50
10	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	100

<b>TOTAL</b>	<b>96,02</b>
--------------	--------------

Promediando las dos pruebas hechas se tiene un resultado de 96.115% de efectividad para la red neuronal en el primer grupo. Las mismas pruebas fueron llevadas a cabo para el algoritmo de PCA y se obtuvieron los siguientes resultados:

Tabla 3. Porcentaje de reconocimiento PCA

PORCENTAJE DE RECONOCIMIENTO PCA																					ACIERTO
AVE #	I	II	III	IV	V	VI	VII	VIII	IX	X	XI	XII	XIII	XIV	XV	XVI	XVII	XVIII	XIX	XX	[%]
1	90,5	96,2	100	76,9	85,6	92,5	90,1	86,5	89,9	13,9	85,6	100	100	96,6	100	94,8	91,8	100	94,6	56,3	87,09
2	90,8	91,5	89,5	76,5	85,6	84,9	85,4	94,4	96	94,7	94,6	98,6	93,4	97,1	97,6	78	99,1	84,6	85,2	73,5	89,55
3	99,3	92	71,4	86,1	98,5	82,2	79,2	75,3	95,6	100	86,4	100	100	86,3	100	100	79,5	98,2	98	100	91,40
4	78,5	86,3	81,8	100	99,6	99	79,5	99,6	78,9	100	100	40,1	86,4	95,6	100	95,6	100	100	95,6	100	90,83
5	65,3	100	5,7	94,5	75,8	86,1	100	94,7	96,8	94,1	99,6	95,6	100	86,4	80,7	70,7	100	95,6	100	100	87,08
6	100	79,5	100	79,5	100	86,3	100	68,3	100	86,3	99,3	100	86,4	96,6	78,9	100	79,5	100	78,9	95,6	90,76
7	87,6	100	100	97,4	90,5	95,9	75,8	100	86,4	100	86,4	100	100	90,2	100	100	100	78,9	100	100	94,46
8	100	76,3	90,3	98,5	89,3	75,6	100	86,4	100	92,1	91,2	86,4	100	78	65,3	97,8	85,3	78,4	95,6	95,6	89,11
9	100	86,3	100	78,7	86,3	100	100	100	74,1	78,9	100	100	79,5	100	100	100	86,7	100	100	100	93,53
10	99,7	100	81,4	92,7	86,4	79,5	100	97,3	89,3	99,7	86,4	95,6	78,9	100	100	79,5	100	78,9	95,6	100	92,05
<b>TOTAL</b>																					<b>90,58</b>
LUGAR OCUPADO EN LA PCA																					ACIERTO
AVE #	I	II	III	IV	V	VI	VII	VIII	IX	X	XI	XII	XIII	XIV	XV	XVI	XVII	XVIII	XIX	XX	[%]
1	1	1	1	3	3	1	2	2	1	6	1	1	1	1	1	1	1	1	1	4	80,42
2	1	2	3	1	4	1	2	1	1	1	3	1	2	1	1	5	1	1	1	6	73,92
3	1	1	3	2	1	3	2	2	1	1	1	1	1	1	1	1	1	1	1	1	85,83
4	1	1	2	1	1	1	1	1	1	1	1	2	1	1	1	1	1	1	1	1	95,00
5	1	1	8	1	1	1	1	1	1	1	1	1	1	1	1	2	1	1	1	1	93,13
6	1	1	1	1	1	1	1	2	1	1	1	1	1	1	1	1	1	1	1	1	97,50
7	2	1	1	1	1	1	2	1	1	1	1	1	1	1	1	1	1	1	1	1	95,00
8	1	3	1	1	2	1	1	1	1	1	1	1	1	1	3	1	1	1	1	1	90,83
9	1	1	1	2	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	97,50
10	1	1	3	1	1	1	1	1	2	1	1	1	1	1	1	1	1	1	1	1	94,17
<b>TOTAL</b>																					<b>90,33</b>

AL promediar las dos pruebas obtenemos un 90.455% de efectividad para este algoritmo en el grupo 1, y además se puede observar que logro clasificar algunos de los patrones que no logro la red.

Las pruebas de este grupo también se clasificaron mediante correlación y los resultados se pueden observar en las siguientes tablas:

Tabla 4. Porcentaje de reconocimiento Correlación

PORCENTAJE DE RECONOCIMIENTO CORRELACION																					ACIERTO
AVE #	I	II	III	IV	V	VI	VII	VIII	IX	X	XI	XII	XIII	XIV	XV	XVI	XVII	XVIII	XIX	XX	[%]
1	99,9	95,4	71,2	71,7	61,9	96,7	94,9	95	85,9	46,7	94	98,1	94,5	80,4	99,7	92,1	92	96	92,8	86	87,25
2	95	90,6	90,8	80,6	92	87,5	0,5	94,8	80,9	89	95,6	97	92,8	88,1	87,4	85,6	91,6	88,9	92,7	98,7	86,01
3	97,5	68,7	34,2	94,5	98,6	24,8	30,8	96,7	85,17	51,2	100	100	100	100	100	100	75	63,4	94,3	0,9	75,79
4	84,6	95,5	95,6	93,4	82,7	83,2	92	94,4	92,9	92,7	90,4	31,8	98,3	97,6	90	85,3	94,6	87,3	84,3	92,8	87,97
5	97,2	59,9	16,4	81,5	21,9	75,4	10,3	92,1	69,2	73,1	82,5	90,5	7,3	91	86	15,1	95,6	93,2	87,3	90	66,78
6	60	97,8	94,7	96,3	67,1	46,9	75,1	64,5	74,5	77,1	89,9	74,1	65,6	89,9	96	90	85	97,2	90,6	90,8	81,16
7	66	73,9	85,8	83,2	81,6	63	67,5	64,7	64,7	39	68,3	73,6	75,7	62,6	69,3	33,8	65,7	90,3	90,7	89,1	70,43
8	98	98,2	99,4	95,2	99,6	89,3	95,5	92,6	98,5	98	95	92	98	82,3	80,2	96,1	95,5	84,8	98,1	98	94,22
9	95,8	96,5	91,4	86,9	88,5	94,5	90,1	94,4	95,4	94	93,1	91,8	91,3	89,8	79	92,9	80,8	81,9	15,3	83	86,32
10	79,5	89,4	75,1	83,4	88,5	78,5	72,6	63,4	75,1	79,8	84,9	64,4	83,8	70,3	95,6	90,5	92,8	86,7	95,2	97	82,33
<b>TOTAL</b>																					<b>81,82</b>
LUGAR OCUPADO EN CORRELACION																					ACIERTO
AVE #	I	II	III	IV	V	VI	VII	VIII	IX	X	XI	XII	XIII	XIV	XV	XVI	XVII	XVIII	XIX	XX	[%]
1	1	1	1	1	1	1	1	1	1	5	1	1	1	1	1	1	1	1	1	1	96,00
2	1	1	1	1	1	1	8	1	1	1	1	1	1	1	1	1	1	1	1	1	95,63
3	1	1	3	1	1	4	3	1	1	2	1	1	1	1	1	1	1	1	1	9	82,64
4	4	1	1	3	2	2	2	2	1	2	1	6	1	1	1	1	1	1	1	1	76,25
5	1	2	7	1	4	2	5	1	1	1	1	1	4	1	1	5	1	1	2	1	72,71
6	4	1	1	1	3	4	2	3	2	1	1	3	3	2	1	1	1	1	1	1	71,67
7	2	3	1	1	2	3	2	2	2	4	2	2	2	2	2	6	2	1	1	1	57,92
8	1	1	1	1	1	1	1	1	1	1	1	1	1	2	2	1	1	2	1	1	92,50
9	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	2	7	4	89,46
10	4	2	4	2	4	2	4	4	2	2	1	4	1	2	1	1	1	1	1	1	62,50
<b>TOTAL</b>																					<b>79,73</b>

En este caso al promediar el resultado de las dos pruebas obtenemos una efectividad del 80.775% la cual resulta bastante baja para este grupo en comparación con los algoritmos anteriores.

Para la creación de la segunda base de datos se decidió escoger las especies del siguiente listado que conformarían el Grupo 2:

- *Pipreola arcuata*
- *Piprites chloris*
- *Pitangus sulphuratus*
- *Procnias averano*
- *Progne chalybea*
- *Progne tapera*
- *Pyrocephalus rubinus*
- *Querula purpurata*

- *Ramphotrigon fuscicauda*
- *Riparia riparia*

En este caso se tomaron 10 muestras para cada especie, pues en el grupo 1 el número de muestras era variable por especie y debido a que se observó que las especies que tenían menor número de muestras presentaron un reconocimiento inferior a las de mayor número. De igual manera al grupo 1 todos los track fueron debidamente editados con el fin de mejorar la calidad de las muestras de referencia que se conservaran en la base de datos.

Los cantos usados en las pruebas fueron encontrados en Macaulay Library y Xeno-Canto, se editó las partes que contenían voz humana y se corrigió la frecuencia de muestreo cuando fue necesario, pues cabe recordar que las grabaciones deben tener una frecuencia de muestreo de 44000 Khz para que las procese el sistema. Los resultados del proceso para la red neuronal del Grupo 2 fueron los siguientes:

Tabla 5. Porcentaje reconocimiento Red Neuronal 2

PORCENTAJE DE RECONOCIMIENTO RED NEURONAL																					ACIERTO
AVE #	I	II	III	IV	V	VI	VII	VIII	IX	X	XI	XII	XIII	XIV	XV	XVI	XVII	XVIII	XIX	XX	[%]
1	96,8	97,7	97,1	96,6	98,0	96,5	98,3	96,1	95,2	96,0	96,7	96,6	95,7	96,9	96,3	96,2	97,2	98,6	99,4	95,3	96,85
2	98,7	99,3	99,1	98,7	99,1	98,4	99,3	100,0	98,4	98,7	98,2	98,4	98,6	98,8	98,0	98,2	98,3	98,4	98,3	96,3	98,57
3	98,4	98,1	99,3	96,9	99,1	99,8	98,1	98,9	99,6	98,3	98,2	99,1	98,8	99,7	99,6	98,6	98,0	99,3	98,6	99,6	98,81
4	99,1	98,5	99,2	98,5	99,0	99,0	99,7	98,1	99,8	99,4	98,5	99,8	99,8	99,4	99,3	99,5	98,4	99,5	98,3	98,5	99,06
5	100,0	99,9	98,1	98,2	99,4	98,9	98,6	99,1	99,1	99,4	99,2	98,9	99,8	99,9	98,2	99,8	99,7	98,1	99,5	98,2	99,09
6	95,3	99,9	95,3	95,9	97,2	98,0	95,4	95,3	96,4	97,0	95,6	95,6	99,2	99,2	96,1	95,3	97,9	95,5	99,9	98,5	96,91
7	93,7	99,7	96,3	95,4	95,6	99,9	95,3	92,1	96,4	98,8	98,5	94,7	95,0	93,1	98,9	95,6	93,5	95,1	96,4	92,6	95,83
8	96,2	99,0	98,3	97,1	91,0	97,2	96,4	98,0	94,3	99,2	99,6	94,8	93,7	91,0	93,8	90,1	95,7	92,8	94,8	98,3	95,56
9	98,2	99,5	97,0	98,6	94,5	99,6	95,0	95,8	98,7	95,8	96,2	94,1	93,7	96,1	94,5	95,2	96,6	97,3	95,6	93,7	96,29
10	99,3	99,9	98,3	95,7	96,6	97,6	97,9	96,1	98,5	97,8	95,8	96,9	98,4	95,8	98,6	96,2	95,0	98,5	95,8	99,3	97,40
TOTAL																					97,44
LUGAR OCUPADO EN LA RED NEURONAL																					ACIERTO
AVE #	I	II	III	IV	V	VI	VII	VIII	IX	X	XI	XII	XIII	XIV	XV	XVI	XVII	XVIII	XIX	XX	[%]
1	1	1	2	1	1	1	1	2	1	1	1	1	1	1	1	1	1	1	1	1	95,00
2	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	2	1	1	1	1	97,50
3	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	100,00
4	1	1	2	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	97,50
5	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	100,00
6	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	100,00
7	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	100,00
8	1	1	1	1	1	1	3	1	1	1	1	1	1	1	1	1	1	1	1	1	96,67
9	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	100,00
10	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	100,00
TOTAL																					98,67

Al promediar las dos pruebas obtenemos un 98.055% de efectividad y además al observar la tabla de posiciones nos damos cuenta que la posición más alejada en la que se clasifico un canto es en la 3, lo cual en el grupo 1 a pesar de no ser frecuente algunos se clasificaron en el último puesto, casualmente eran los mismos con el menor número de muestras en la base de datos.

Las pruebas se aplicaron al algoritmo de PCA y se obtuvieron los siguientes resultados:

Tabla 6. Porcentaje reconocimiento PCA 2

PORCENTAJE DE RECONOCIMIENTO PCA																					ACIERTO
AVE #	I	II	III	IV	V	VI	VII	VIII	IX	X	XI	XII	XIII	XIV	XV	XVI	XVII	XVIII	XIX	XX	[%]
1	93,5	95,4	94,2	94,6	96,0	93,0	96,5	92,6	91,8	92,1	94,9	94,7	91,5	93,9	92,6	92,3	94,5	98,5	98,8	90,5	94,09
2	92,5	90,7	97,7	94,9	96,0	91,3	98,3	97,1	98,1	91,4	91,2	96,7	95,4	95,8	98,2	91,5	92,4	91,1	93,3	91,8	94,27
3	81,6	95,0	82,5	89,7	89,8	80,2	80,3	99,6	94,6	99,5	86,4	89,7	85,4	92,0	93,9	80,2	95,0	92,9	95,7	97,6	90,07
4	95,3	86,7	90,8	82,8	85,1	87,0	83,5	84,9	81,8	86,0	98,3	98,4	84,2	92,7	92,6	81,5	82,8	89,3	81,7	96,6	88,10
5	93,1	82,2	82,7	94,5	85,6	92,7	82,3	83,0	83,7	98,4	90,1	85,8	83,3	92,4	87,8	83,0	90,0	87,3	88,7	89,1	87,79
6	99,9	94,5	99,9	92,5	90,3	95,4	96,4	96,8	96,0	95,1	98,4	98,4	91,8	97,5	99,3	91,0	98,8	90,8	99,2	96,8	95,94
7	86,6	89,6	97,4	88,0	97,2	94,5	99,9	99,1	96,6	87,6	88,3	91,1	90,5	89,4	97,4	94,4	98,6	97,0	93,8	99,1	93,81
8	82,6	99,5	94,0	83,8	88,4	82,5	83,2	94,7	93,7	98,1	82,1	83,2	85,5	99,4	83,7	96,8	81,5	84,4	99,5	85,2	89,09
9	98,5	99,5	96,0	91,1	89,6	89,6	86,4	88,7	98,4	98,5	87,7	94,4	94,6	86,0	94,8	95,0	87,8	88,5	91,6	98,1	92,75
10	94,6	90,5	95,0	95,2	97,1	93,4	94,7	89,7	89,8	95,1	91,2	96,3	98,0	95,1	94,9	99,1	96,7	98,7	99,2	94,0	94,91
<b>TOTAL</b>																					<b>92,08</b>
LUGAR OCUPADO EN LA PCA																					ACIERTO
AVE #	I	II	III	IV	V	VI	VII	VIII	IX	X	XI	XII	XIII	XIV	XV	XVI	XVII	XVIII	XIX	XX	[%]
1	1	2	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	3	94,17
2	1	1	2	1	1	2	1	2	1	1	1	1	1	1	1	1	1	1	1	1	92,50
3	1	3	1	3	1	2	1	1	1	1	1	1	1	1	2	1	1	1	1	1	88,33
4	1	2	1	1	1	1	1	1	2	1	1	1	1	1	1	1	1	1	1	1	95,00
5	1	1	1	1	1	1	2	2	1	1	1	1	3	1	1	3	1	1	1	1	88,33
6	2	1	2	1	1	1	1	1	1	1	4	1	1	1	1	2	1	1	1	1	88,75
7	1	3	1	1	1	2	1	1	1	1	1	1	1	2	1	1	1	1	2	1	89,17
8	1	1	1	2	1	1	1	4	1	1	2	1	2	1	1	1	1	1	1	1	88,75
9	1	1	1	1	1	1	1	1	2	1	1	3	1	1	1	3	1	1	1	3	87,50
10	1	2	1	1	1	1	1	1	3	1	1	1	1	1	1	1	1	1	1	1	94,17
<b>TOTAL</b>																					<b>90,67</b>

Al promediar las dos pruebas obtenemos un 91.374% de efectividad el cual mejoró en reconocimiento, sin embargo al mirar la tabla de posiciones notamos muchas de las pruebas fueron clasificadas hasta el 4º lugar, pues si bien el PCA es efectivo produce cierto tipo de dispersión en los resultados.

Finalmente las pruebas fueron aplicadas al algoritmo de correlación y se obtuvieron los siguientes resultados:

Tabla 7. Porcentaje reconocimiento Correlación 2

PORCENTAJE DE RECONOCIMIENTO CORRELACION																					ACIERTO
AVE #	I	II	III	IV	V	VI	VII	VIII	IX	X	XI	XII	XIII	XIV	XV	XVI	XVII	XVIII	XIX	XX	[%]
1	93,9	94,2	93,2	95,9	91,4	98,2	92,3	95,1	85,0	87,6	92,4	89,2	94,6	98,4	90,0	99,7	97,9	92,4	85,9	87,0	92,71
2	95,5	93,2	88,8	96,5	92,8	93,6	97,9	88,7	91,8	92,7	87,1	88,6	93,7	96,9	89,3	92,9	91,7	85,5	91,9	98,0	92,35
3	89,4	78,7	98,4	82,7	85,1	70,5	87,1	76,2	96,7	89,6	73,3	98,1	72,6	95,8	74,7	78,8	96,8	76,3	90,2	92,0	85,15
4	96,6	86,5	73,5	93,6	82,0	97,6	87,0	71,9	95,0	82,8	89,5	94,6	93,2	71,4	80,2	70,1	94,9	86,6	97,6	75,0	85,97
5	71,5	97,3	86,6	83,5	94,9	73,4	88,7	81,0	73,3	91,6	85,0	88,3	83,4	95,4	80,4	96,9	86,5	75,4	83,0	77,0	84,65
6	95,3	94,2	91,1	91,6	85,7	97,6	87,2	92,4	87,1	96,0	89,6	85,9	91,5	86,2	88,2	90,0	90,9	88,2	90,1	90,4	90,48
7	91,4	95,6	92,1	90,3	97,2	92,1	98,6	100,0	99,9	98,7	94,6	91,5	90,5	97,0	98,4	90,7	95,8	99,7	96,7	98,1	95,44
8	85,5	94,0	83,2	83,3	89,6	87,1	91,6	82,6	96,5	97,4	88,5	88,5	86,1	94,8	97,1	93,4	85,1	97,1	83,7	94,0	89,95
9	97,0	93,8	89,5	89,0	98,8	97,7	89,0	95,3	95,9	97,3	88,8	98,3	91,0	90,6	94,7	91,8	93,0	99,6	93,7	89,2	93,69
10	85,1	86,9	95,6	92,0	94,9	86,6	94,1	97,2	96,6	85,1	87,9	87,1	97,4	96,4	93,5	97,9	90,0	95,5	87,1	97,4	92,21
<b>TOTAL</b>																					<b>90,26</b>
LUGAR OCUPADO EN CORRELACION																					ACIERTO
AVE #	I	II	III	IV	V	VI	VII	VIII	IX	X	XI	XII	XIII	XIV	XV	XVI	XVII	XVIII	XIX	XX	[%]
1	1	2	1	1	1	1	2	1	1	3	1	1	2	1	1	1	1	1	1	1	89,17
2	1	1	4	1	1	1	1	1	1	1	1	3	1	1	1	2	1	1	3	1	87,08
3	1	2	5	1	3	3	1	1	1	1	1	1	3	1	1	1	4	1	1	1	79,75
4	1	1	1	4	1	1	1	2	1	1	2	1	1	1	2	1	1	1	2	1	86,25
5	1	3	1	1	3	1	2	1	1	1	1	2	1	1	1	1	1	1	1	1	88,33
6	1	2	2	1	1	1	1	1	1	2	1	1	1	1	2	1	1	1	1	3	86,67
7	1	1	1	1	3	1	1	1	1	1	1	2	3	1	1	1	2	1	1	1	88,33
8	1	2	1	1	1	1	3	1	1	1	1	1	1	2	1	1	1	1	3	1	88,33
9	1	1	1	1	1	1	1	1	1	1	1	3	1	1	3	1	1	1	1	1	93,33
10	1	1	1	2	1	1	1	1	2	1	1	1	2	1	1	1	2	1	2	1	87,50
<b>TOTAL</b>																					<b>87,48</b>

Al promediar las dos pruebas obtenemos un 88.868% de efectividad en el reconocimiento de los patrones de cada track, que comparado con los algoritmos de ANN y PCA resulta inferior, pero no es un resultado del todo despreciable si se tiene en cuenta la sencillez de la comparación y el costo computacional tan reducido que conlleva este tipo de selección.

Al promediar los resultados del primer y segundo grupo obtenemos una relación de efectividad entre los 3 algoritmos de selección que se puede apreciar mejor en el siguiente grafico:

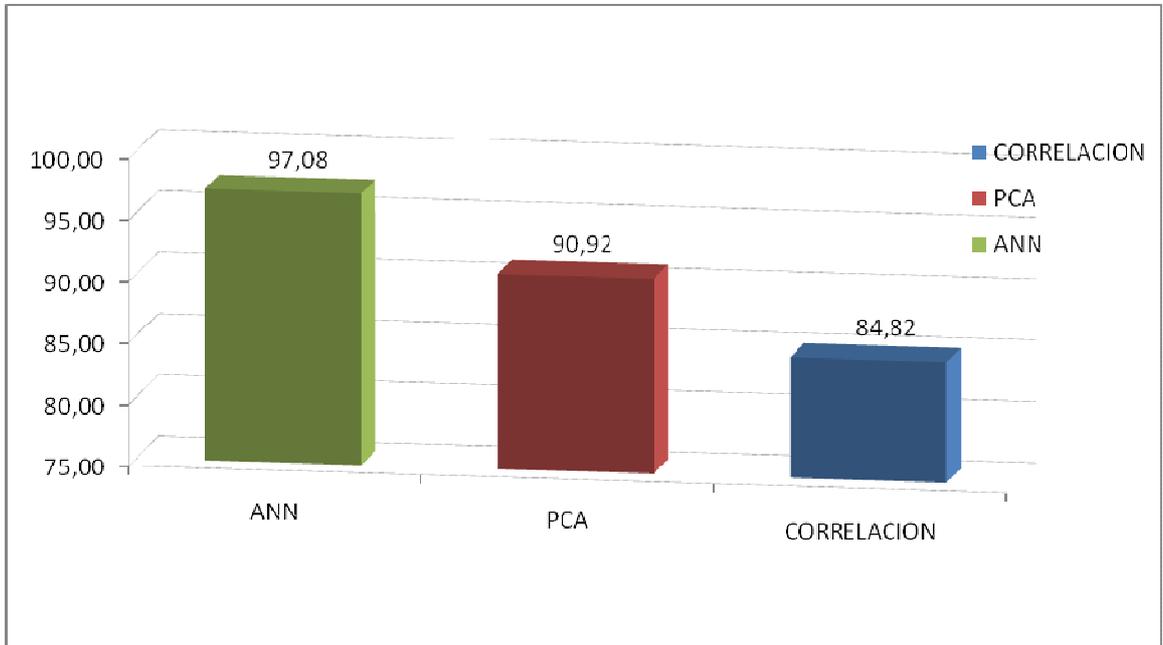


Figura 36. Comparación reconocimiento ANN, PCA y Correlación.

Se puede decir que los porcentajes de reconocimiento dejan muy en claro que la red neuronal es el sistema con el mejor desempeño reconociendo los patrones, y lo más importante es que los resultados arrojados poseen menor dispersión. En la siguiente figura se ilustra el resultado de los tres algoritmos para las 20 pruebas de la especie 9 del grupo 2:

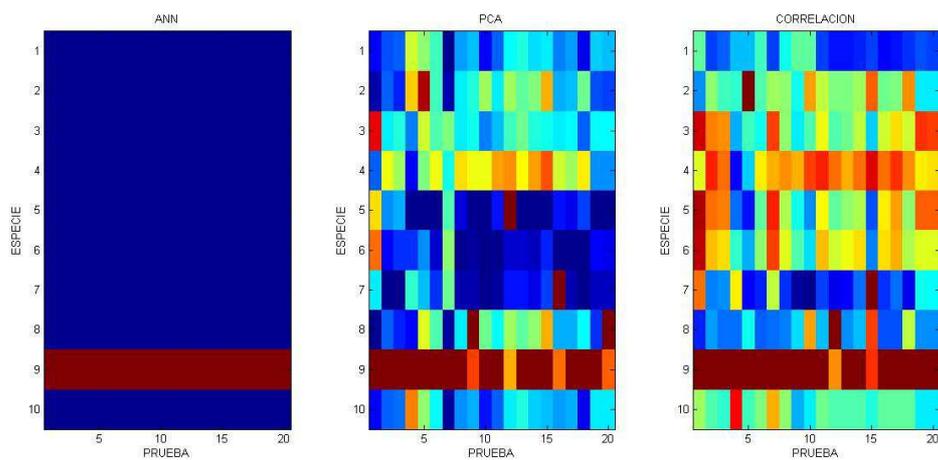


Figura 37. Comparación reconocimiento ANN, PCA y Correlación 20 especies.

Mientras el valor del resultado es más cercano a 1 el color se torna rojo oscuro y mientras el valor es cercano a 0 el color se torna más azul, así podemos observar que los resultados para la red neuronal son totalmente diferenciables pasan del rojo al azul, mientras que en los resultados de PCA y correlación se aprecia cierto *ruido* entre los resultados de las especies que no son las de la grabación, a pesar de tener un buen porcentaje de reconocimiento quedan unas brechas muy pequeñas entre la especie reconocida y la no reconocida.

## 7.2 IMPLEMENTACIÓN DE LA FUNCIÓN PSD EN EL SISTEMA EMBEBIDO

Al hacer pruebas con patrones PSD arrojados por MATLAB en la red neuronal de la tarjeta los resultados eran los mismos, y al hacer pruebas con los patrones calculados por la tarjeta se evidenciaban ciertas pequeñas variaciones decimales producto de las pequeñas deformaciones que se producían en el cálculo del PSD ya que al introducir tales patrones calculados por la tarjeta en las redes de MATLAB se obtenían los resultados que la tarjeta desplegaba y por tanto se determino que la red funcionaba exactamente igual a la red de MATLAB.

Para estimar la distorsión  $\zeta$  entre los patrones calculados por MATLAB ( $P_m$ ) y los calculados por la tarjeta ( $P_t$ ) se hicieron una serie de pruebas tanto en MATLAB como en la tarjeta y se estimó una distorsión dada por la siguiente función:

$$\zeta = \frac{|P_m - P_t|}{P_m} \times 100\% \quad (22)$$

Que describe el porcentaje promedio de las diferencias absolutas entre cada punto del vector, y se obtuvieron los resultados de la siguiente tabla:

Tabla 8. Distorsión MATLAB y Embebido

DISTRORCION					
Pruebas	$\zeta$ [%]		$\zeta$ [%]		$\zeta$ [%]
1	1,1337	21	0,9449	41	0,6548
2	0,4888	22	0,9631	42	0,6981
3	0,2721	23	0,7716	43	0,9314
4	0,4131	24	0,2215	44	0,5302
5	0,7047	25	0,5431	45	0,2774
6	1,1131	26	0,3675	46	0,4706
7	0,2028	27	0,7088	47	0,3837
8	0,5641	28	0,8605	48	0,7351
9	0,9391	29	1,0451	49	0,6737
10	0,4429	30	0,7973	50	0,6955
11	0,3679	31	0,6589	51	0,8415
12	0,5647	32	0,4073	52	0,7844
13	0,5735	33	0,5903	53	0,7357
14	0,3827	34	0,4602	54	0,9274
15	0,2834	35	0,8636	55	0,8292
16	0,9793	36	0,9408	56	0,815
17	0,7359	37	0,5288	57	0,9619
18	0,8571	38	0,3543	58	0,7094
19	0,8394	39	0,4899	59	0,8979
20	0,9197	40	0,3334	60	0,8966

$\zeta$ PROMEDIO	0,668 %
------------------	---------

Dejando como resultado final una distorsión  $\zeta$  promedio de aproximadamente 0.66% para 60 pruebas realizadas.

De forma similar a las pruebas realizadas en el software de escritorio para los diferentes algoritmos, se decidió hacer pruebas para los dos grupos con la red neuronal implementada en la tarjeta y los resultados obtenidos para el primer grupo se reflejan en la siguiente tabla:

Tabla 9. Porcentaje reconocimiento Red Neuronal Tarjeta 1

PORCENTAJE DE RECONOCIMIENTO RED NEURONAL																					ACIERTO
AVE#	I	II	III	IV	V	VI	VII	VIII	IX	X	XI	XII	XIII	XIV	XV	XVI	XVII	XVIII	XIX	XX	[%]
1	99,6	99,4	99,5	99,2	1,4	98,9	99,2	99,4	99,1	1,5	99,3	99,4	99,0	99,1	99,2	99,2	99,1	99,0	99,2	99,1	89,44
2	99,4	99,4	99,9	99,3	-0,3	99,3	99,1	99,7	100,0	99,6	99,0	99,6	99,6	99,0	99,3	99,2	99,9	99,7	99,9	99,2	94,49
3	99,0	99,9	15,1	99,0	99,5	99,4	99,4	99,9	99,5	29,3	99,1	99,9	99,4	99,4	99,3	99,3	99,3	10,3	99,0	99,6	87,24
4	99,4	99,5	99,8	99,8	99,9	99,2	99,1	99,3	100,0	99,9	99,1	99,8	99,8	99,8	99,1	99,5	99,4	99,8	100,0	100,0	99,61
5	99,6	99,0	99,8	99,2	99,5	99,4	99,8	99,8	100,0	99,7	99,3	99,0	99,2	99,1	99,6	99,8	99,5	99,4	100,0	99,6	99,52
6	99,9	99,7	99,4	99,6	99,3	99,7	99,1	99,3	99,2	99,2	99,6	99,8	99,1	99,9	99,0	99,8	99,7	99,7	100,0	99,7	99,52
7	99,8	99,6	99,8	99,3	21,6	-0,8	1,2	99,9	99,5	99,8	99,7	99,2	99,7	11,3	99,9	99,1	99,4	99,4	99,4	99,9	81,33
8	99,7	99,6	99,6	99,9	99,5	99,4	99,9	99,7	99,9	99,3	99,8	99,1	99,1	99,1	99,4	99,4	99,4	99,8	99,2	99,8	99,54
9	99,4	99,0	99,5	99,0	99,3	99,3	99,1	99,2	99,9	99,1	99,3	99,6	99,3	99,8	99,2	99,3	0,6	99,6	99,9	99,0	94,42
10	99,5	99,9	99,9	99,9	99,4	99,8	99,4	99,1	99,8	99,8	99,7	99,3	99,3	99,2	99,4	99,1	100,0	99,2	99,8	99,6	99,55
<b>TOTAL</b>																					<b>94,47</b>
LUGAR OCUPADO EN LA RED NEURONAL																					ACIERTO
AVE#	I	II	III	IV	V	VI	VII	VIII	IX	X	XI	XII	XIII	XIV	XV	XVI	XVII	XVIII	XIX	XX	[%]
1	1	1	1	1	9	1	1	1	1	8	1	1	1	1	1	1	1	1	1	1	91,18
2	1	1	1	1	3	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	96,67
3	1	1	5	1	1	1	1	1	1	3	1	1	1	1	1	1	6	1	1	1	88,50
4	1	1	1	1	1	1	1	1	1	1	1	2	1	1	1	1	1	1	1	1	97,50
5	1	1	1	1	1	1	1	1	1	1	1	1	1	1	2	1	1	1	1	1	97,50
6	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	100,00
7	1	1	1	1	3	8	6	1	1	1	1	1	1	3	1	1	1	1	1	1	84,79
8	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	100,00
9	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	8	1	1	1	1	95,63
10	1	1	1	1	1	1	1	1	2	1	1	1	1	1	1	1	1	1	1	1	97,50
<b>TOTAL</b>																					<b>94,93</b>

De donde al promediar las dos pruebas obtenemos un 94.7% de efectividad. Para el segundo grupo obtuvimos los resultados de la siguiente tabla:

Tabla 10. Porcentaje reconocimiento Red Neuronal Tarjeta 2

PORCENTAJE DE RECONOCIMIENTO RED NEURONAL																					ACIERTO
AVE#	I	II	III	IV	V	VI	VII	VIII	IX	X	XI	XII	XIII	XIV	XV	XVI	XVII	XVIII	XIX	XX	[%]
1	99,0	99,9	99,7	98,8	99,6	99,5	99,8	99,9	98,4	99,9	99,1	99,0	99,5	99,8	99,5	99,9	99,2	99,0	99,8	99,9	99,46
2	99,2	99,1	99,9	99,4	99,6	99,9	99,2	99,2	99,9	99,8	99,9	99,1	99,6	99,9	99,2	99,2	99,1	99,1	99,1	1,6	94,56
3	51,7	99,6	99,9	99,2	99,5	-93,3	99,1	99,2	99,6	99,8	99,3	99,5	99,4	99,0	99,5	-69,5	99,1	99,1	99,1	99,1	78,90
4	99,5	99,9	99,1	99,6	99,8	99,1	99,1	99,6	99,4	99,8	99,0	99,3	99,0	99,6	99,3	99,8	99,1	99,8	99,1	99,5	99,43
5	99,2	99,9	98,7	99,1	99,6	99,1	17,6	99,1	98,2	99,8	99,4	99,6	99,0	100,0	98,7	99,0	99,4	99,2	99,5	99,1	95,17
6	99,5	99,4	99,7	99,8	68,9	99,3	99,2	99,2	16,2	99,1	86,6	99,0	99,6	99,1	99,1	99,3	99,2	99,2	99,5	99,3	93,01
7	99,2	99,8	99,1	99,9	99,7	99,2	99,4	99,5	99,3	99,8	12,7	99,9	99,6	99,3	99,9	99,0	99,7	99,1	99,5	99,5	95,16
8	99,1	99,7	99,6	99,5	99,5	99,8	99,4	100,0	99,3	99,2	99,0	99,5	99,9	99,3	99,7	99,4	99,5	99,8	99,8	99,9	99,54
9	99,4	99,0	99,8	99,0	99,1	99,8	99,2	99,2	99,6	99,3	99,5	99,5	99,2	99,7	99,1	99,5	99,0	99,4	99,3	99,2	99,35
10	99,0	99,9	99,7	99,4	99,7	99,5	99,6	99,1	2,2	99,2	99,8	99,7	99,6	99,6	99,0	99,8	99,2	99,5	99,4	99,8	94,65
<b>TOTAL</b>																					<b>94,92</b>
LUGAR OCUPADO EN LA RED NEURONAL																					ACIERTO
AVE#	I	II	III	IV	V	VI	VII	VIII	IX	X	XI	XII	XIII	XIV	XV	XVI	XVII	XVIII	XIX	XX	[%]
1	1	1	1	1	2	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	97,50
2	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	5	96,00
3	3	1	1	1	3	1	1	1	1	1	1	1	1	1	2	1	1	1	1	1	90,83
4	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	100,00
5	1	1	1	1	1	5	1	1	1	1	1	1	1	1	1	1	1	1	1	1	96,00
6	1	1	1	1	2	1	1	1	4	1	3	1	1	1	1	1	1	1	1	1	90,42
7	1	1	1	1	1	1	1	1	1	1	6	1	1	1	1	1	1	1	2	1	93,33
8	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	100,00
9	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	100,00
10	1	1	1	1	1	1	1	1	5	1	1	2	1	1	1	1	1	1	1	1	93,50
<b>TOTAL</b>																					<b>95,76</b>

En promedio de las dos pruebas se refleja un 95.34% de efectividad.

En total para los dos grupos tenemos un 95.05% el cual es bastante cercano al obtenido para el sistema de escritorio que recordemos tenía una efectividad de 97.08%, lo cual es bastante aceptable y se puede decir que el sistema embebido se comporta de manera similar al de escritorio.

Es importante notar que la red que contiene la tarjeta se entreno con patrones calculados en MATLAB y conociendo que existe cierta distorsión, se podría decir que se igualarían los resultados o se mejorarían en una pequeña proporción si se entrena la red de la tarjeta con los patrones que se extrae y calcula la misma.

## 8. CONCLUSIONES

### 8.1 CONCLUSIONES Y APORTES

El uso de redes neuronales artificiales resulta el método más eficaz en el reconocimiento bioacústico, si bien existen otros métodos como PCA o correlación (ampliamente usado por biólogos) que resultan más simples y con la gran ventaja de no necesitar entrenamiento no sobrepasan los altos porcentajes de reconocimiento con ANN.

Las redes neuronales por el hecho de tener internamente una muy alta cantidad de pesos alterados en un amplio aprendizaje de varios cantos presentan una mayor flexibilidad en cuanto al reconocimiento de un patrón nuevo que presente ruido o algunas alteraciones mas.

Trabajar en el dominio de la frecuencia es el primer paso al caracterizar un canto de ave, pues como en los seres humanos la anatomía marca cambios de frecuencia en la voz, y si bien existen factores a considerar como prolongaciones o pausas en el habla o en el canto, el espectro en frecuencia simplifica el trabajo reduciendo la cantidad de datos característicos.

La transformada discreta de Fourier determina un espectro marcado que diferencia un canto de otro en el dominio de la frecuencia, sin embargo, la cantidad de datos es muy amplia y la comparación no resulta sencilla; la función densidad espectral de potencia PSD disminuye la cantidad de datos y presenta una marcación más clara del espectro dado que para cada canto la energía es depositada principalmente en dos o tres frecuencias.

Algoritmos de reconocimiento como PCA o Correlación son más flexibles en la aplicación en cuanto a cambio de bases de datos a reconocer se refiere, únicamente basta con cambiar la matriz con la que se va a operar, las redes neuronales en cambio requieren de un método de entrenamiento que requiere tiempo y validación.

Los resultados fueron satisfactorios dado que se usaron tanto cantos de ave de la región como archivos de audio de diferentes reservas del mundo con porcentajes de efectividad superiores a 97% con redes neuronales.

Actualmente la elaboración de sistemas embebidos se facilita con el uso de potentes sistemas como Domino Fez por el trabajo con Microframework que permite usar poderosas herramientas de uso de hardware de forma eficaz para el desarrollo de software con algoritmos complejos.

En el sistema embebido se ha trabajado de forma limpia en el manejo de hardware, es virtual para el usuario controlar cada recurso, lo que genera una disminución de errores y tiempo destinado a la programación, en general es más importante preocuparse por no saturar las CPU con instrucciones para agilizar el trabajo, y no almacenar una excesiva cantidad de información.

El hecho de usar C# en el lenguaje de programación ha facilitado la generación del algoritmo en el sistema embebido, como por ejemplo en la creación de la función PSD y manejo de librerías.

Los tres códigos de reconocimiento ANN, PCA y correlación son factibles de implementar en el sistema embebido, sin embargo aquí resulta más complejo aun el trabajo con otras bases de datos en el caso de redes neuronales, dado que no se posee un algoritmo de entrenamiento como el incluido en MATLAB.

Los resultados obtenidos en el sistema embebido con redes neuronales fueron satisfactorios con una baja disminución de efectividad comparada con la obtenida en un PC, 1%, se observa una mayor diferencia en el tiempo de procesamiento, con el uso de la función PSD y lectura de la memoria SD.

## **8.2 DESARROLLOS FUTUROS**

La aplicación que se generó incluye reconocimiento del canto de aves por medio de Redes Neuronales, código PCA y Correlación, esto basándose en una base de datos de 10 aves diferentes, aunque para PCA y Correlación se pueden crear bases de datos nuevas. Si se quiere trabajar con una nueva base de datos a reconocer con Redes Neuronales, es necesario editar internamente el código de los pesos, por tanto podría usarse una plataforma auxiliar y virtual para el usuario de entrenamiento de redes neuronales que genere los nuevos pesos de forma automática y los valide, sin incurrir en grandes errores por tratarse de un entrenamiento más sencillo.

El sistema embebido que se ha creado responde de forma muy cercana a los resultados encontrados en la plataforma por PC, para esto se ha trabajado de forma conjunta con los archivos de audio grabados por la grabadora de sonido de alta fidelidad; el tratamiento de señal que la grabadora hace internamente entrega una señal limpia y digitalizada con una amplia resolución, podría plantearse con el ánimo de hacer más compacto el sistema, realizar una tarjeta de tratamiento de audio que remplace la digitalización avanzada de la grabadora de sonido con una amplia resolución y filtrado adecuado.

## 9. BIBLIOGRAFÍA

BAPTISTA, L.F. & GAUNT, S.L.L., 1997. Bioacoustics as a tool in conservation studies. In: *Behavioral Approaches to Conservation in the Wild*: 212-242 (J. R. Clemmons and R. Buchholz, Eds.). Cambridge: Cambridge University Press.

BECKER, P. H., 1982. The coding of species specific characteristics in bird sounds. In: *Acoustic communication in birds*: 213-252 (D. E. Kroodsma & E. H. Miller, Eds.). Vol. 1. New York: Academic Press.

BIN-AMIN, Talal y MAHMOOD, Iftekhar. Speech Recognition Using Dynamic Time Warping. Islamabad: 2nd International Conference on Advances in Space Technologies, 2008.

Dan Ellis, Function Mp3 read and write MATLAB [En línea] Disponible en internet: <http://labrosa.ee.columbia.edu/MATLAB/mp3read.html>

Dan. Ellis. Dynamic Time Warp (DTW) in MATLAB. [En línea] Disponible en internet: <http://www.ee.columbia.edu/~dpwe/resources/MATLAB/dtw/>.

FALLS, J.B., 1981. Mapping territories with playback: An accurate census method for songbirds. In: *Estimating numbers of terrestrial birds*: 86-91. (C. J. Ralph & J. M. Scott, Eds.) Stud. in Avian Biol., 6. Kansas: Allen Press.

GASTON, K. J. 1996. Species richness: measure and measurement, pp. 77-113, en: Biodiversity. Blackwell Science.

HALFFTER, G., MORENO, C. & PINEDA, E. 2001. *Manual para la evaluación de la biodiversidad*. En: *Reservas de la Biosfera*. Manuales y Tesis Sociedad Entomológica Aragonesa. Volumen 2. Zaragoza, España. 80 pp.

ICONTEC, Norma Técnica Colombiana 1486 (sexta actualización). Documentación. Presentación de tesis, trabajos de grado y otros trabajos de investigación.

Ingeniería de las Ondas I, [En línea] Disponible en internet: [http://www.lpi.tel.uva.es/~nacho/docencia/ing\\_ond\\_1/trabajos\\_01\\_02/formatos\\_audio\\_digital/html/wavformat.htm](http://www.lpi.tel.uva.es/~nacho/docencia/ing_ond_1/trabajos_01_02/formatos_audio_digital/html/wavformat.htm)

KROODSMA, D. E. 1996. Ecology of passerine song development. In: Ecology and evolution of acoustic communication in birds. (D. E. Kroodsma, and E. H. Miller, eds.), pp. 3–19. Cornell University Press, Ithaca, NY.

KUMAR, Kishore y KUMAR, Lalithl. Speech Recognition Using Neural Networks. Singapore: International Conference on Signal Processing Systems, 2009.

LOMONT, Chris. The Fast Fourier Transform. Chris Lomont's homepage[online]. Enero de 2010. Available from [www.lomont.org](http://www.lomont.org)

Mathworks, Neural Network Toolbox™ 7 User's Guide. [En línea] Disponible en internet: [http://fita.hua.edu.vn/ntthuy/courses/nnet\\_MATLAB.pdf](http://fita.hua.edu.vn/ntthuy/courses/nnet_MATLAB.pdf)

MAALY, Iman y EL-OBAID, Manal. Speech Recognition using Artificial Neural Networks. Damascus: Information and Communications Technologies, 2006.

OROZCO, Mauricio. ACOSTA, Carolina. OCAMPO Tobón, Sergio. Representaciones de disimilitudes en bioacústica computacional: una metodología para el monitoreo automatizado de aves [http://www.manizales.unal.edu.co/gta/signal/morozcoa/ornitologia2010\\_diapositivas.pdf](http://www.manizales.unal.edu.co/gta/signal/morozcoa/ornitologia2010_diapositivas.pdf)

PAYNE, R.B., 1986. Bird songs and avian systematics. In: *Current Ornithology*: 87-126 (R. J. Johnston, Ed.). Vol. 3. New York: Plenum Publishing Corporation.

RALPH, C.J., GEOFFREY R.G., PYLE, P., MARTIN, T.E, DESANTE, D.F & MILÁ, B. 1996. Manual de métodos de campo para el monitoreo de aves terrestres. Gen. Tech. Rep. Albany,CA: Pacific Southwest Research Station, Forest Service, U.S. Department of Agriculture.

STILES, F.G. 1983. Birds. Introduction, p. 502-530. In D.H. Janzen (ed.). Costa Rican natural history. University of Chicago, Chicago, Illinois, USA.

STOTZ, D.F., PARKER, T.A., FITZPATRICK, J.W. & MOSKOVITS, D.K. 1996. *Neotropical birds: ecology and conservation*. Chicago: University Press. 481 pp.

TUBARO, P.L. 1999. Bioacústica aplicada a la sistemática, conservación y manejo de poblaciones naturales de aves. *Etología*, 7:19-32.

VILLARREAL H., M. ÁLVAREZ, S. CÓRDOBA, F. ESCOBAR, G. FAGUA, F. GAST, H. MENDOZA, M. OSPINA y A.M. UMAÑA. 2006. Manual de métodos para el desarrollo de inventarios de biodiversidad. Segunda Edición. Programa de Inventarios de Biodiversidad. Instituto de Investigación de Recursos Biológicos Alexander von Humboldt. Bogotá, Colombia. 236 p.

*William H. Press, Saul A. Teukolsky, William T. Vetterling, Brian P. Flannery*  
Numerical Recipes 2007

Xeno-Canto, Sharing birds sound from around the world [En línea] Disponible en internet: <http://www.xeno-canto.org>

## 10. ANEXOS

### ANEXO A. Eigenvectores y eigenvalores

Un eigenvector de una matriz  $C$  (como por ejemplo la matriz covarianza) es un vector  $v$  que cumple con la ecuación

$$Cv = \lambda v \quad (10)$$

Donde  $\lambda$  es el eigenvalor del eigenvector.

Es claro que hay múltiples eigenvalores y eigenvectores para una misma matriz. Y que la ecuación representa un cambio de la magnitud del eigenvector cuando es multiplicado por la matriz  $C$

Así dada una matriz  $C$   $n \times n$ , se tendrán  $n$  eigenvectores cada uno con su respectivo eigenvalor. Además todos los eigenvectores de una matriz serán perpendiculares, lo que resulta interesante pues así podemos representar los datos en este nuevo sistema de coordenadas, no hay que olvidar que los eigenvectores se deben representar en longitud uno.

### ANEXO B. Redes neuronales

Las redes neuronales artificiales están inspiradas en la organización del complejo sistema neuronal del cerebro humano. El sistema nervioso humano constituido por células llamadas neuronas presenta una estructura muy compleja. El número estimado de neuronas es de  $10^{11}$  y las interconexiones entre ellas son del orden de  $10^{15}$ .

Cada neurona comparte muchas características con otras células del cuerpo humano pero tiene propiedades particulares y especiales para recibir, procesar y transmitir señales electroquímicas a través de todas las interconexiones del sistema de comunicación del cerebro.

Del cuerpo de la neurona se extienden las dendritas hacia otras neuronas donde reciben las señales transmitidas por otras neuronas. El punto de contacto o de conexión se llama sinapsis y estas entradas son dirigidas al núcleo donde se suman. Algunas de las entradas tienden a excitar a la célula y otras sin embargo

tienden a inhibir la célula. Como resultado si la excitación acumulada supera un valor umbral, las neuronas envían una señal a través del axón a otras neuronas. En general este es el modelo de las ANN aun cuando el comportamiento real de una célula nerviosa tiene muchas complejidades y excepciones.

Usualmente, las redes neuronales se ajustan, o entrenan, para que una entrada particular conduzca a una salida u objetivo específico. Normalmente se requieren muchos pares de entrada-salida para entrenar una red neuronal.

El conexionismo o estudio de las redes neuronales artificiales, estaba inicialmente inspirado por la neurobiología, pero ahora se ha vuelto un campo interdisciplinario incluyendo la ciencia de computación, ingeniería eléctrica y electrónica, matemáticas, física, sicología, lingüística entre otras. Algunos investigadores aún se encuentran estudiando la neurofisiología del cerebro humano, pero se está prestando mucha atención a las propiedades de la neuro computación, usando modelos neuronales simplificados.

El trabajo con gran cantidad de neuronas para diferentes fines determina ciertas clasificaciones.

**Según el número de capas:** existen las *redes neuronales monocapas y multicapas*, según si la entrada se propaga a la salida mediante un grupo de neuronas o si hay varios grupos conectados en serie hasta la salida

**Según el tipo de conexiones:** existen las *redes neuronales recurrentes y no recurrentes*; según exista o no lazos de realimentación o memoria de la salida hacia la entrada.

**Según el grado de conexión:** donde se encuentran las *redes neuronales totalmente o parcialmente conectadas*; según si las conexiones entre capas son totales o no.