

**INSTITUTO TECNOLÓGICO DE MÉRIDA**

# **TESIS**

**“INVESTIGACIÓN Y DESARROLLO DE UNA  
INTERFAZ CEREBRO – COMPUTADORA CON  
ETAPA DE CALIBRACIÓN PARA EL SISTEMA”**

**PARA OPTAR AL GRADO DE:  
MAESTRO EN INGENIERÍA**

**PRESENTA:**

**I.M. RAFAEL ANGEL LEYVA BUSTILLO**

**ASESOR:**

**DR. CARLOS ALBERTO LUJÁN RAMÍREZ**

**MÉRIDA, YUCATÁN, MÉXICO  
JUNIO 2019**





"2019, Año del Caudillo del Sur, Emiliano Zapata"

Mérida, Yucatán, 08/mayo/2019

**ASUNTO:** Solicitud de autorización de impresión.

**M.C. Daniel Arcángel López Sauri**  
Jefe de de la División de Estudios  
de Posgrado e Investigación  
**PRESENTE**

En virtud de que el C. Leyva Bustillo Rafael Angel, ha culminado satisfactoriamente la tesis: "Investigación y desarrollo de una interfaz cerebro-computadora con etapa de calibración para el sistema", participando exitosamente en el Seminario de Investigación de la misma y cubierto los requisitos necesarios para optar al grado de "Maestro en Ingeniería" le solicitamos se le otorgue la Autorización correspondiente para realizar la impresión de su trabajo final, el cual avala como un producto de calidad la Comisión Revisora conformada por los que a continuación firman.

ASESOR	REVISOR
Dr. Carlos Alberto Luján Ramírez	M.C. José Agustín Hernández Benítez
REVISOR	REVISOR
Dr. José Ramón Atoche Enseñat	Dr. Jesús Sandoval Gio

**A T E N T A M E N T E**  
*Excelencia en Educación Tecnológica.*

**M.C. Sylvia María del Rosario Ruiz Casanova**  
Coordinadora de la Maestría en Ingeniería  
C.p. Archivo  
DALS/SRC/fja





# SEP

SECRETARÍA DE  
EDUCACIÓN PÚBLICA



TECNOLÓGICO NACIONAL DE MEXICO

Instituto Tecnológico de Mérida

"2019, Año del Caudillo del Sur, Emiliano Zapata"

**DEPENDENCIA:** DIV. DE EST. DE POSG. E INV.

**No. DE OFICIO:** X-136/19

**ASUNTO:** AUTORIZACIÓN DE IMPRESIÓN

Mérida, Yucatán, **17/mayo/2019**

**C. LEYVA BUSTILLO RAFAEL ANGEL**  
**Pasante de la Maestría en Ingeniería**  
**PRESENTE**

De acuerdo al fallo emitido por su asesor el **Dr. Carlos Alberto Luján Ramírez**, y la comisión revisora integrada por el M.C. José Agustín Hernández Benítez, el Dr. José Ramón Atoche Enseñat y el Dr. Jesús Sandoval Gio, considerando que cubre los requisitos establecidos en el Reglamento de Titulación de los Institutos Tecnológicos le autorizamos la impresión de su trabajo profesional con la TESIS:

**"INVESTIGACIÓN Y DESARROLLO DE UNA INTERFAZ CEREBRO-COMPUTADORA CON ETAPA DE CALIBRACIÓN PARA EL SISTEMA"**

**ATENTAMENTE**

*Excelencia en Educación Tecnológica*

**M.C. DANIEL ARCÁNGEL LÓPEZ SAURI**  
**JEFE DE LA DIVISION DE ESTUDIOS DE**  
**POSGRADO E INVESTIGACIÓN**

C.p. Archivo  
DALs/fja



**S. E. P.**  
**INSTITUTO TECNOLÓGICO**  
**DE MERIDA**  
**DIVISION DE ESTUDIOS DE**  
**POSGRADO E INVESTIGACION**



## Tabla de Contenido

<b>Capítulo 1</b> .....	<b>1</b>
<b>1.1 Introducción</b> .....	1
<b>1.2 Antecedentes</b> .....	3
<b>1.3 Descripción de la Problemática</b> .....	7
<b>1.4 Justificación</b> .....	9
<b>1.7 Objetivos</b> .....	11
<b>Capítulo 2</b> .....	<b>12</b>
<b>2.1 Anatomía del cerebro</b> .....	12
<b>2.1.1 División del cerebro</b> .....	13
<b>2.1.2 Comunicación de neuronas</b> .....	14
<b>2.2 Electroencefalograma</b> .....	15
<b>2.2.1 Tipos de señales</b> .....	15
<b>2.2.2 Ondas rítmicas</b> .....	16
<b>2.2.3 Protocolo EEG</b> .....	17
<b>2.2.4 Sensores EEG</b> .....	17
<b>2.3 Procesamiento digital de señales</b> .....	20
<b>2.3.1 Filtros Windowed-Sinc</b> .....	20
<b>2.3.3 Transformada Wavelet</b> .....	21
<b>2.3.3.1 Transformada Continua Wavelet</b> .....	22
<b>2.3.3.2 Transformada Discreta Wavelet</b> .....	23
<b>2.4 Inteligencia Artificial</b> .....	26
<b>2.4.1 Aprendizaje automático</b> .....	26

2.4.1.1 Adaboost .....	27
2.4.1.2 SVM .....	29
2.5 Interfaz gráfica.....	30
2.5.1 Matlab.....	30
2.5.2 VRML.....	30
2.6 BCI.....	31
2.6.1 BCI: Concepto básico.....	31
2.6.2 BCI vs Neuroprótesis.....	32
2.6.3 Aplicaciones.....	32
Capítulo 3 .....	33
3.1 Base de datos.....	33
3.2 Selección de características Adaboost.....	35
3.3 Clasificación de datos Adaboost.....	35
3.4 SVM.....	36
3.5 Interfaz gráfica.....	36
Capítulo 4 .....	37
4.1 Base de datos.....	37
4.2 Selección de características Adaboost.....	41
4.3 Clasificación de datos Adaboost.....	41
4.4 SVM.....	42
4.5 Interfaz gráfica .....	42
Capítulo 5 Resultados .....	44
Capítulo 6 Conclusiones.....	57
Referencias .....	58

## Índice de Figuras

### Índice de Figuras

Figura 1 – Diseño de sistema BCI de Mason.....	Pág. 4
Figura 2 – BRAIN2ROBOT de Robert Mueller.....	Pág. 4
Figura 3 – Exoesqueleto Berkeley Bionics.....	Pág. 5
Figura 4 – Silla de ruedas.....	Pág. 6
Figura 5 – Carro autónomo.....	Pág. 6
Figura 6 – Gráfica de porcentajes de tipos de discapacidades en personas en México.....	Pág. 9
Figura 7 - División del cerebro.....	Pág. 13
Figura 8 – Estructura general de la neurona.....	Pág. 14
Figura 9 - Electroencefalograma.....	Pág. 15
Figura 10 - Diagrama de posición de electrodos.....	Pág. 17
Figura 11 – Resonancia magnética.....	Pág. 18
Figura 12 – Sensor Insight.....	Pág. 19
Figura 13 – Sensor EPOC.....	Pág. 19
Figura 14 – Sensor MindWave.....	Pág. 19
Figura 15 – Tipos de filtros.....	Pág. 21
Figura 16 – Filtro wavelet.....	Pág. 25
Figura 17 – Ejemplo SVM.....	Pág. 29
Figura 18 – Proceso de sistema BCI.....	Pág. 31
Figura 19 – Proceso de neuroprótesis.....	Pág. 32.
Figura 20 – Interfaz gráfica .....	Pág. 43
Figura 21 – Resultados Coiflet 5.....	Pág. 44
Figura 22 – Resultados Symlet 4.....	Pág. 45
Figura 23 – Resultados Symlet 5.....	Pág. 45
Figura 24 – Resultados Symlet 6.....	Pág. 46
Figura 25 – Resultados DB 6.....	Pág. 46

Figura 26 – Resultados DB 40.....	Pág. 47
Figura 27 – Resultados DB 45.....	Pág. 47
Figura 28 – Resultados wavelet Adaboost.....	Pág. 48
Figura 29 – Resultados Coiflet 5 SVM.....	Pág. 48
Figura 30 – Resultados Symlet 4 SVM.....	Pág. 49
Figura 31 – Resultados Symlet 5 SVM.....	Pág. 49
Figura 32 – Resultados Symlet 6 SVM.....	Pág. 50
Figura 33 – Resultados DB 6 SVM.....	Pág. 50
Figura 34 – Resultados DB 40 SVM.....	Pág. 51
Figura 35 – Resultados DB 45 SVM.....	Pág. 51
Figura 36 – Resultados wavelet SVM.....	Pág. 52
Figura 37 – Tesis de referencia .....	Pág. 54
Figura 38 – Resultados sistema local.....	Pág. 55



# Capítulo 1 – Información del Proyecto

## 1.1 Introducción

Desde hace décadas, el hombre ha anhelado poder transformar sus pensamientos en acciones. Sin embargo, el reto tecnológico que actualmente separa al hombre de su meta es bastante elevado. Disciplinas como neurociencia, computación, matemáticas y psicología se ven involucradas al momento de intentar entender la complejidad del cerebro humano.

El trabajo de grupos multidisciplinarios de investigación ha podido concretar proyectos que han acortado la distancia. Un ejemplo es Brain2Robot [15], un proyecto que consiste en el procesamiento de señales EEG provenientes del cerebro, para poder controlar un brazo robótico. Esto ha sido posible gracias a la creación de una nueva disciplina llamada BCI.

BCI (por sus siglas en inglés, Brain Computer – Interface) o ICC (Interfaz cerebro – computadora) es una disciplina que tiene como objetivo permitir al hombre controlar distintas máquinas con la mente.

En este trabajo se presenta el diseño de un sistema BCI con la capacidad de poder auto-calibrarse dependiendo del usuario que lo utilice. El diseño de este sistema está formado por cuatro bloques principales que son adquisición de la señal, auto-calibración, procesamiento de la señal e interfaz de control.

Se centra la investigación y desarrollo en la etapa de auto-calibración, debido a que un sistema BCI que cuente con esta etapa permitiría su uso para casi cualquier persona de forma inmediata pasando por un periodo de calibración automática de algunos minutos.

Toman gran relevancia los métodos utilizados en la etapa de procesamiento de la señal, debido a que esta se sub-divide en cuatro áreas: pre-procesamiento,

extracción, selección y clasificación de características. Utilizar el método adecuado nos podrá otorgar un sistema BCI de alto nivel.

La transformada Wavelet ha sido elegida para extraer información relevante de las señales cerebrales que se encuentran en la base de datos empleada para este trabajo de investigación. La principal razón para elegir Wavelets sobre otras técnicas como la transformada de Fourier es su capacidad de descomponer señales no estacionarias en eventos que se encuentran localizados en el tiempo.

En la etapa de selección de características se usa el método AdaBoost, este ha sido elegido por su alta capacidad como red neuronal y para obtener información de cómo funciona este algoritmo en un sistema BCI.

Como herramienta de clasificación se utiliza un método de redes neuronales debido a su efectividad para la clasificación de patrones ya que permite separar señales cerebrales en clases y dividirlos con un modelo matemático que busca no ubicar datos en la clase incorrecta.

Con este proyecto se busca obtener un análisis confiable sobre los métodos utilizados en cada etapa. Con el fin de aportar a futuros trabajos a realizar con sistemas BCI. De igual forma la etapa de calibración representa un proyecto de estado del arte para esta disciplina.

## 1.2 Antecedentes

BCI (por sus siglas en inglés, Brain Computer – Interface) o ICC (Interfaz cerebro – computadora) es una ciencia que se hace presente durante los años 70's, cuando el hombre por necesidad de mejorar o solucionar los problemas de discapacidad de nuestro cuerpo donde no responde a los impulsos del cerebro. Los estudios iniciales consistieron en pruebas de deletreo basadas en electroencefalografía y en el control de dispositivos de una sola neurona, como la implantación de prótesis neuronales para recuperar la audición, vista o movilidad dañadas.

Uno de los primeros y más importantes trabajos sobre BCI que se realizó fue por el científico J. J. Vidal [1] donde él dio una de las primeras y más importantes definiciones de BCI, donde dijo "The BCI system is geared to use both the spontaneous EEG and the specific evoked responses triggered by time-dependent stimulation under various conditions for the purpose of controlling such external apparatus as e.g. prosthetic devices." En su traducción al español "Los sistemas ICC son orientados a utilizar señales EEG espontáneas y otras estimuladas, ambas en tiempo real con el propósito de controlar aparatos externos o dispositivos de prótesis".

A partir de estas primeras investigaciones el estudio de BCI empezó a tomar fuerza y poco a poco se fue estandarizando el método adecuado para poder diseñar sistemas, pero no fue hasta el año 2003 cuando Steven G. Mason propuso un nuevo modelo funcional para el diseño de sistemas BCI, el diseño se muestra en la figura 1.

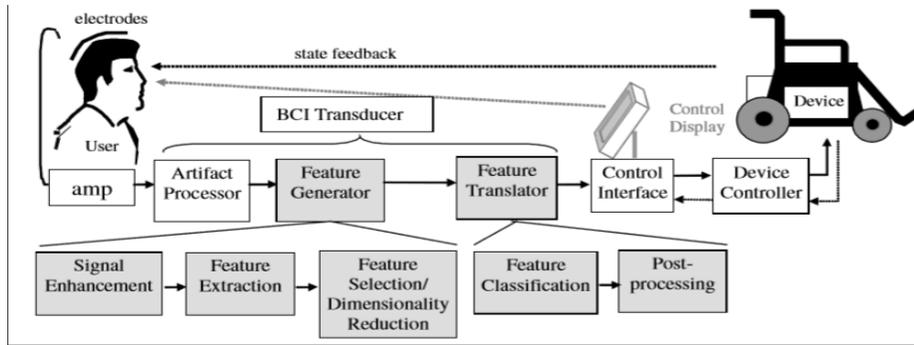


Figura 1. Diseño de sistema BCI de Mason.

A principios del siglo XXI surgieron las primeras Interfaces Cerebro-Computadora no invasivas, tecnología que ha dado un giro hacia la industria de los videojuegos y a la industria médica. Posterior a esto grandes empresas, centros de investigaciones médicas y por supuesto países primermundistas han decidido dedicar grandes cantidades de dinero y recursos en el desarrollo de nuevas tecnologías basadas en BCI.

Klause Robert Mueller [15] como jefe de la investigación, realizó con su equipo de investigadores el proyecto BRAIN2ROBOT, el objetivo de este proyecto fue unir la tecnología BCI con *eye tracker* (esta última consiste en evaluar el punto donde se fija la mirada, a partir de cámaras é infrarrojos), para poder mover un brazo robótico. Esta fue la primera gran muestra del poder de BCI al realizar el movimiento de un brazo robótico como un medio de apoyo y con un financiamiento de 12 millones de dólares, la figura 2 muestra el ejemplo del experimento.



Figura 2. BRAIN2ROBOT de Robert Mueller.

Las uniones de universidades europeas empezaron a trabajar hasta 2012 en el proyecto llamado HIVE [12] (por sus siglas en inglés, Hyper interaction viability experiments), que consiste en generar una nueva generación de tecnología que no solo comunique el cerebro con la computadora, sino que además sirva para mandar estímulos al cerebro a través de métodos no invasivos. Este gran proyecto en conjunto de muchas universidades ha logrado de manera precisa el estudio del comportamiento del cerebro en tecnologías BCI con ayuda de estimulaciones para mejorar las señales recibidas.

Berkeley Bionics [8] desarrolla un exoesqueleto que busca ayudar a personas con problemas en la médula espinal a que pueda caminar de nuevo con ayuda de tecnología BCI. Este sistemas es de lo más poderoso que se puede controlar con BCI debido a que también cuenta con un gran sistema de inteligencia artificial que ayuda a poder realizar movimientos complejos. Este gran avance muestra que para poder emular movimientos humanos con gran eficiencia, se debe de unir los sistemas BCI con inteligencia artificial. La figura 3 nos muestra el exoesqueleto.



Figura 3. Exoesqueleto Berkeley Bionics.

Raúl Rojas [13] científico de nacionalidad mexicana que se encuentra trabajando en la universidad de Berlín, el actualmente investiga sobre el uso de carros autónomos con y sin la tecnología BCI, el realizó dos experimentos en los que manejaba un vehículo totalmente con BCI y un segundo donde usaba un vehículo semiautónomo con apoyo de tecnología BCI. El con estos dos experimentos pudo demostrar la factibilidad de BCI para realizar tareas cotidianas y aún más que al complementar esta tecnología con otras puede funcionar con alta eficiencia no solo para personas con discapacidades sino para todo mundo en general. El experimento se realizó de manera igual con una silla de ruedas y con un automóvil comercial, ambos modificados para su uso con estas tecnologías. En la figura 4 se muestra la silla de ruedas y en la figura 5 el automóvil utilizado.



Figura 4. Silla de ruedas.



Figura 5. Carro autónomo.

### 1.3 Descripción de la Problemática

*Según la Clasificación Internacional del Funcionamiento, de la Discapacidad y de la Salud [9], las personas con discapacidad “**son aquellas que tienen una o más deficiencias físicas, mentales, intelectuales o sensoriales y que al interactuar con distintos ambientes del entorno social pueden impedir su participación plena y efectiva en igualdad de condiciones a las demás**”.*

Actualmente el 10% de la población mundial presenta algún tipo de discapacidad permanente (motriz, sensorial, mental, entre otras), es decir, cerca de 600 millones de personas.

También existen personas con discapacidades transitorias o temporales, cerca de un 15% de la población mundial se encuentra en este rubro. A estos datos hay que agregar a las personas de la 3ra edad, las cuales tienen algún impedimento o discapacidad temporal o permanente, situando a estos en el 12% de la población mundial. De esta manera podemos saber que aproximadamente el 37% de la población mundial sufre algún problema de discapacidad.

Tan solo en México, de acuerdo a un estudio realizado por el INEGI [10], existen más de 5 millones de personas con algún tipo de discapacidad, lo que representa el 5.1% de nuestra población hasta ese año.

Este dato preocupa debido a la gran cantidad de personas que no pueden tener un nivel alto de calidad de vida, por ello, durante los últimos años se ha tratado de enfocar un gran número de ramas de la ciencia a solucionar este problema. Una de las principales áreas que ha logrado un avance significativo es BCI (por sus siglas en inglés, Brain Computer Interface), encontrando métodos no invasivos para ayudar a las personas con discapacidades a poder realizar tareas cotidianas con apoyo de prótesis robóticas o vehículos de apoyo.

BCI ha sido una rama emergente durante los últimos años, aunque ha aportado grandes avances todavía no hemos podido explotar todo lo que se puede realizar.

Igual que en cualquier área presenta diversas problemáticas que actualmente están sometidas a investigaciones para poder resolverlas, una de ellas es mejorar la precisión de reconocimiento de patrones y características, debido a que nuestro cerebro no funciona con cierta lógica y puede presentar cambios muy repentinos, por ello se propone investigar y agregar nuevos métodos para realizar esto.

De igual manera en BCI y en otras áreas con el procesamiento digital de señales durante los últimos años se han usado diferentes técnicas para la extracción, selección y clasificación de datos y/o patrones, sin embargo, a pesar de existir ya desde hace mucho tiempo, algunas técnicas han sido poco utilizadas, tal vez por su complejidad de entendimiento o nivel matemático más avanzado.

Tal es el caso de la transformada de Wavelet, una herramienta matemática muy poderosa para utilizar especialmente con señales no-estacionarias, entendiendo por no-estacionaria a una señal que varía constantemente en el tiempo. Este tipo de señales pueden ser de las más recurrentes en nuestro alrededor, un ejemplo muy claro de este tipo de señales se pueden encontrar en nuestro cuerpo, como las señales que produce el cerebro.

Al igual que Wavelet existen otras técnicas muy útiles para selección como AdaBoost, Genetic Algorithm, o para clasificación como SVM. Un paso importante será encontrar la mejor combinación de estas herramientas para poder realizar un procesamiento más completo de señales no-estacionarias.

Los sistemas BCI tienen, de manera similar, el problema para usarse en más de un individuo. Esto se debe a las variaciones en las señales cerebrales utilizadas en el proceso. Para ello se hace una calibración al sistema a partir de métodos muy lentos o difíciles de ejecutar.

## 1.4 Justificación

De una investigación hecha por la OMS en 2012, se sabe que aproximadamente el 37% de la población mundial sufre de algún problema de discapacidad temporal o permanente. Este resultado también se puede ver como que una da cada tres personas se ve afectada por alguna discapacidad. Por lo tanto, cualquier aporte que resulte en la mejora de la calidad de vida de estas personas tendrá un impacto social significativo.

Otro estudio realizado por parte del Instituto Nacional de Estadística y Geografía (INEG) [9], deja en manifiesto que la discapacidad que más predomina en México, con un 58.3%, se debe a daños en el sistema motriz (Figura 6). A este tipo de discapacidad le siguen la discapacidad por problemas visuales (27.2%), auditivos (12.1%), mentales (8.5%), entre otras.

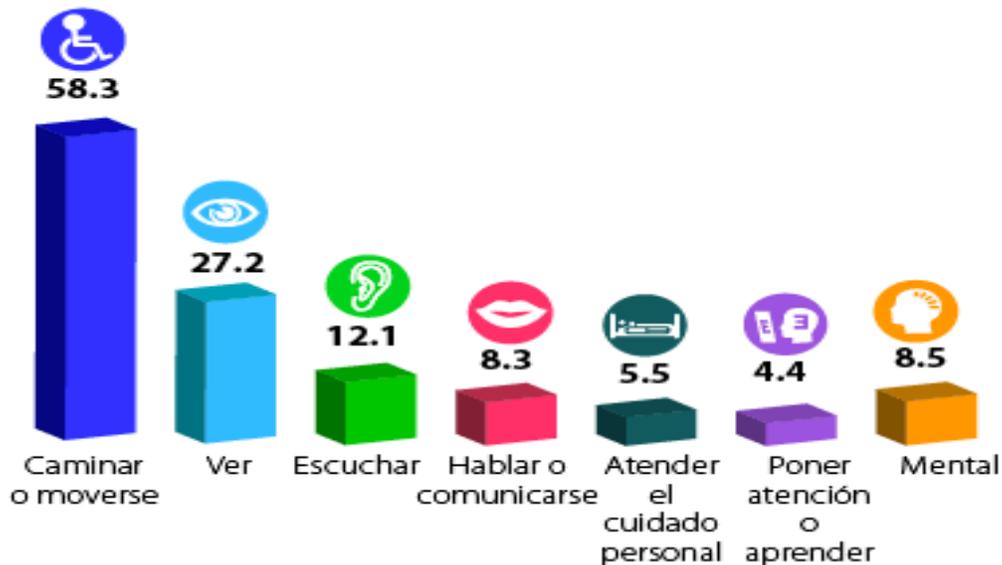


Figura 6. Gráfica de porcentajes de tipos de discapacidades en personas en México.

Es una problemática que genera gran preocupación en México, debido al gran número de personas que se encuentran en estos grupos. Sin embargo, las herramientas usadas actualmente para apoyo y mejora de vida para estas personas son un tanto ineficientes. Las sillas de ruedas y prótesis de extremidades, aunque funcionan, no permiten poder realizar los movimientos complejos necesarios para el día a día.

La tecnología basada en BCI representa una herramienta muy poderosa para ayudar a personas con problemas de discapacidad, principalmente aquellas con discapacidades motrices. Durante los años de investigación en este tipo de tecnología se ha logrado un gran avance en la rama médica, robótica y mecánica. Algunos ejemplos de estos avances son proyectos encargados de controlar prótesis robóticas, sillas de ruedas, carros semiautónomos y hasta los primeros exoesqueletos, todos controlados con señales EEG.

En esta investigación se diseña un sistema BCI el cual contará con una etapa de auto-calibración y así poder hacer pruebas de funcionamiento a partir de diferentes individuos. Este trabajo estudia técnicas de extracción y selección de características provenientes de señales EEG. Adicionalmente, dos técnicas de clasificación de patrones son exploradas a lo largo de este documento.

Los algoritmos de aprendizaje de máquina y redes neuronales apoyaran en la etapa de auto-calibración generando un entrenamiento para el sistema a partir de la captura de señales del individuo y repitiendo la prueba.

El principal aporte de este trabajo consiste en analizar el desempeño de diferentes técnicas del estado del arte para el desarrollo de sistemas BCI.

## 1.5 Objetivos

### 1.5.1 Objetivos generales

Desarrollar una interfaz cerebro – computadora con etapa de auto-calibración bajo el análisis de este y de diferentes técnicas de reconocimiento de patrones.

### 1.5.2 Objetivos específicos

- Crear una base de datos con señales EEG correspondientes a varios procesos mentales.
- Adquirir las señales eléctricas alfa, beta, delta, gama y teta a partir de las señales EEG en la base de datos.
- Comparar el desempeño de *AdaBoost* y de un tipo de red neuronal para la selección de características.
- Investigar los métodos *AdaBoost* y de un tipo de red neuronal para la clasificación de señales EEG.
- Analizar el desempeño de la etapa de auto-calibración utilizando el sistema con 5 individuos.

## Capítulo 2 – Marco teórico

### 2.1 Anatomía del cerebro

El cerebro es un órgano del cuerpo humano que nos permite imaginar, pensar, aprender, movernos, etc. Situado en la cabeza, este órgano se encarga de centralizar y controlar todo el sistema nervioso periférico y central.

#### 2.1.1 División del cerebro

El cerebro se puede dividir en diferentes partes, que son el telencéfalo, el cerebelo, el diencéfalo y el tronco encefálico.

El telencéfalo se divide en hemisferios, unidas por el cuerpo caloso. La mayoría de los centros de funciones están en cada hemisferio, pero algunos centros son únicos, como el centro del lenguaje, y están localizados individualmente en el hemisferio derecho o izquierdo.

Estos hemisferios se pueden dividir en lóbulos, donde cada uno de estos tiene una función en específico (ver figura 7).

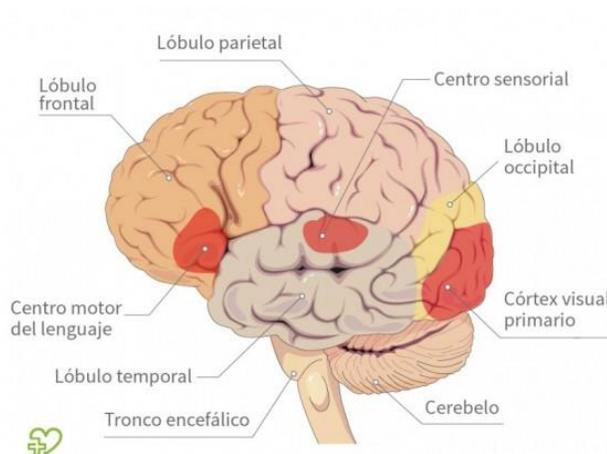


Figura 7. División del cerebro.

- **Lóbulo frontal:** en el área frontal del cerebro; movimiento, comportamiento.
- **Lóbulo parietal:** parte trasera superior; sensaciones.
- **Lóbulo temporal:** situado detrás de cada sien; reconocimiento de personas, oído.
- **Lóbulo occipital:** en la parte posterior de la cabeza; visión.

### 2.1.2 Comunicación de neuronas

Nuestro cerebro está conformado por miles de millones de células cerebrales llamadas neuronas, que utilizan electricidad para comunicarse entre sí (ver figura 8).

La combinación de millones de neuronas enviando señales a la vez produce una enorme cantidad de actividad eléctrica en el cerebro.

La actividad producida por nuestro cerebro genera diferentes tipos de patrones cerebrales. Esto se debe a que son de naturaleza cíclica y ondulada.

Los patrones cerebrales pueden agruparse gracias a su frecuencia, actividad y posición del cuero cabelludo, donde que pueden ser detectados mediante equipos médicos sensibles, que mide los niveles de electricidad en el área del cuero cabelludo. Este tipo de estudio se conoce como electroencefalograma.

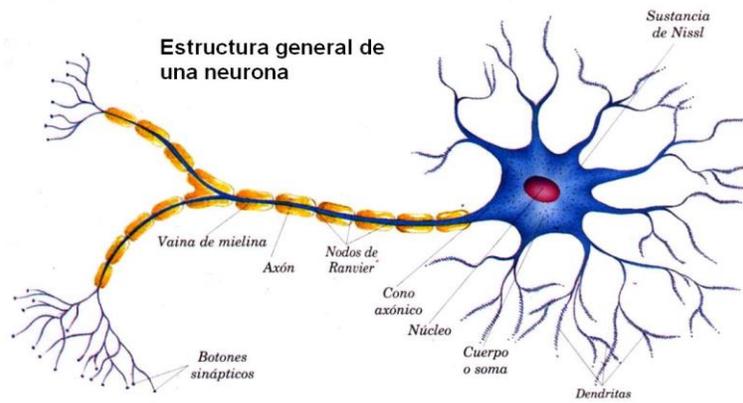


Figura 8. Estructura general de la neurona.

## 2.2 Electroencefalograma

Un electroencefalograma es un estudio que sirve para medir la actividad eléctrica del cerebro. Se realiza mediante un equipo especial llamado electroencefalógrafo.

Esto consiste en captar las señales o patrones eléctricos producidos por el cerebro, por medio de electrodos situados en el cuero cabelludo (ver figura 9).

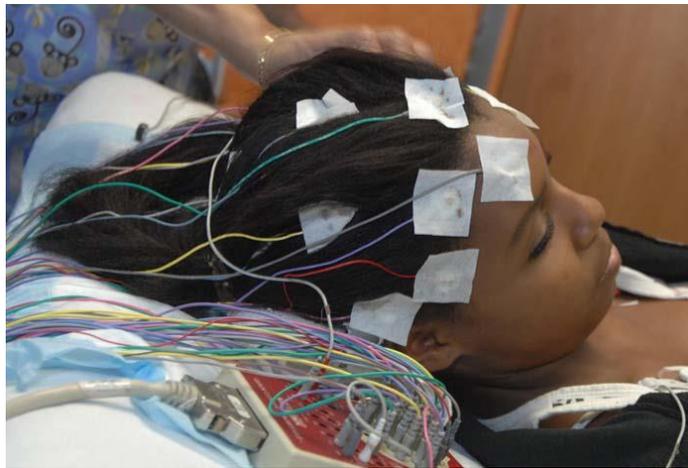


Figura 9. Electroencefalograma.

### 2.2.1 Tipos de señales

De acuerdo a estudios realizados para aplicaciones terapéuticas de EEG (Neidermeyer.,1999) existen diferentes tipos de señales electrofisiológicas, como son rítmicas, potenciales, entre otras.

Nuestro cerebro emite estos diferentes tipos de señales a consecuencia de la actividad neuronal. Estos tipos de señales son las rítmicas y evocadas.

Se considera que las señales rítmicas o espontáneas son las generadas por voluntad, porque no necesitan ningún tipo de estímulo.

Las evocadas de manera contraria, son las producidas por la estimulación de los sentidos con ayuda de imágenes, música, etc.

### **2.2.2 Ondas rítmicas**

Son conocidos como ritmos, producidos por los impulsos eléctricos producidos por la comunicación de las neuronas. Estos transportan información y se pueden observar en un EEG. Se pueden dividir en sub-señales alfa, beta, delta, teta y gamma.

- 1. Onda Alfa: [Rango de frecuencia 7.5 – 12 Hz; low alfa 7.5 – 9.25 Hz; high alfa 10 – 11.75 Hz]** Están asociadas con la relajación. Una manera de llegar a niveles altos de estas ondas es cerrar los ojos y pensar en algo tranquilo. La localización de electrodos más precisa para estas ondas es en la parte trasera de la cabeza (o1 y o2) y el lóbulo frontal.
- 2. Onda Beta: [Rango de frecuencia 12 – 30 Hz; low beta 13 – 16.75; high beta 18 – 29.75]** Están asociadas con la concentración. Las mejores formas de llegar a niveles altos de estas ondas es cuando se resiste o suprime el movimiento, resolviendo ecuaciones matemáticas.
- 3. Onda Teta: [Rango de frecuencia 3.5 – 7.5 Hz]** Están asociadas con la ineficiencia y el sueño profundo.
- 4. Onda Delta: [Rango de frecuencia 0.5 – 3.5 Hz]** Están asociadas con sueño ligero.
- 5. Onda Gamma: [Rango de frecuencia 31 – 50 Hz]** Se cree que refleja el mecanismo de la conciencia. Beta y gamma juntas han estado asociadas con

la atención, percepción y cognición. Se puede estimular meditando mientras se concentra en un objeto en específico.

### 2.2.3 Protocolo EEG

El procedimiento consiste en colocar electrodos en diferentes punto de la corteza cerebral (ver figura 10) de acuerdo a la parte del cerebro que se quiere analizar. Posteriormente se captan las señales eléctricas provenientes de la actividad cerebral, amplifican y analizan por medio de un software.

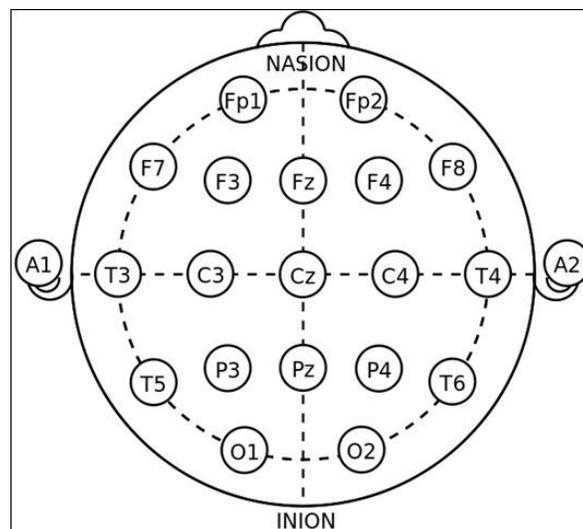


Figura 10. Diagrama de posición de electrodos.

### 2.2.4 Sensores EEG

Los sensores EEG son capaces de representar en tiempo real la actividad del cerebro de forma gráfica. Esta señal se recoge con ayuda de electrodos

posicionados en la corteza cerebral, de acuerdo a la actividad que se desea estudiar.

Actualmente existen en el mercado gran cantidad de sensores EEG, la gran mayoría de estos diseñados y usados con fines médicos. Estos sensores buscan reemplazar las muy costosas máquinas de resonancia magnética (ver figura 11).



Figura 11. Resonancia magnética.

Entre las marcas más conocidas se encuentran: Emotiv Inc., NeuroSky y Enobio. Los dos primeros enfocados al estudio e investigación de señales cerebrales y Enobio diseñado con fines totalmente de estudios médicos.

La compañía Emotiv Inc. ofrece dos diferentes sensores EEG; EPOC (ver figura 12) e Insight (ver figura 13). Siendo la principal diferencia la cantidad de electrodos proporcionados en cada diadema. EPOC cuenta con 14 electrodos húmedos, esto significa que se necesita algún tipo de solución salina para poder recibir correctamente las señales de la corteza cerebral, colocados en las posiciones cerebrales F3, F7, F3, FC5, T7, P7, O1, O2, P8, T8, FC6, F4, F8, AF4. Mientras que

Insight cuenta con solo cinco electrodos secos, no requieren ningún tipo de solución salina, en las posiciones FP1, FP2, F4, F8 y FC6.



Figura 12. Sensor Insight.



Figura 13. Sensor EPOC.

Por su parte, la marca NeuroSky tiene en el mercado el sensor MindWave mobile (ver figura 14), diseñado para investigación y juegos recreativos. Este sensor es popular debido a su bajo precio y fácil comercialización. Cuenta con solo un electrodo seco en la posición FP1 de la corteza cerebral.



Figura 14. Sensor MindWave.

## **2.3 Procesamiento digital de señales**

El procesamiento digital de señales consiste en la extracción y manipulación de información a partir de estas señales. Esto se realiza con ayuda de un conjunto de herramientas matemáticas y de software. De igual manera se emplean estas herramientas para la conversión de señales del dominio del tiempo al dominio de la frecuencia, dependiendo la naturaleza de esta y del proyecto o análisis a realizar.

De manera general el sistema que se usa en para al procesamiento digital de señales consiste en la adquisición de la señal a utilizar. Esto puede ser con ayuda de una tarjeta de adquisición de datos o sensores que puedan captar la señal.

Posteriormente con ayuda de herramientas matemáticas o de software se obtienen los datos que se necesitan para emplear al sistema de estudio.

### **2.3.1 Filtros digitales**

Los filtros digitales son herramientas usadas para eliminar o modificar espectros de frecuencia en señales. Son usados comúnmente en aplicaciones biomédicas, de audio, procesamiento de imágenes, entre otros.

Estos filtros, al igual que los analógicos, pueden configurarse como pasa bajas, pasa altas, pasa bandas o rechaza bandas con distintas frecuencias de corte. Tienen características destacables de estabilidad y rapidez. Siendo superiores a los analógicos, debido a que estos dependen de sus componentes físicos, como resistencias, capacitores, etc.

En términos de su respuesta al impulso los filtros digitales se clasifican de dos formas: FIR (Finite Impulse Response) o filtros de respuesta finita al impulso; e IIR (Infinite Impulse Response) o filtros de respuesta infinita al impulso, que deben su comportamiento a la existencia de lazos de retroalimentación en su estructura.

También podemos clasificar los filtros de acuerdo al dominio en el que trabajan (ver figura 15).

		FILTER IMPLEMENTED BY:	
		Convolution <i>Finite Impulse Response (FIR)</i>	Recursion <i>Infinite Impulse Response (IIR)</i>
FILTER USED FOR:	Time Domain <i>(smoothing, DC removal)</i>	Moving average (Ch. 15)	Single pole (Ch. 19)
	Frequency Domain <i>(separating frequencies)</i>	Windowed-sinc (Ch. 16)	Chebyshev (Ch. 20)
	Custom <i>(Deconvolution)</i>	FIR custom (Ch. 17)	Iterative design (Ch. 26)

Figura 15. Tipos de filtros.

### 2.3.1.1 Filtro Windowed-sinc

El filtro Windowed – sinc se utiliza comúnmente como un filtro pasa bandas, debido a que presenta una gran estabilidad y pueden tener rendimientos muy altos para desempeñar su trabajo en el dominio de la frecuencia. Sin embargo, el costo por un excelente desempeño en el dominio de la frecuencia implica un rendimiento muy pobre en el dominio del tiempo.

### 2.3.2 Transformada Wavelet

La transformada Wavelet es una herramienta matemática diseñada para solucionar el problema de incertidumbre de Heisenberg que se hace presente en el análisis tiempo – frecuencia para señales no-estacionarias.

El problema de incertidumbre de Heisenberg establece “que no puede conocerse la información de tiempo y frecuencia de una señal en cierto punto del plano tiempo – frecuencia”. En otras palabras esto dice que no pueden determinarse exactamente que frecuencias existen en un instante dado.

Este es un problema de resolución y es la razón principal por lo cual se busca reemplazar la STFT (Short-Time Fourier Transform) por la WT (Wavelet Transform).

La solución más efectiva para resolver este problema es analizar la señal con una técnica llamada análisis multiresolución (MRA).

El MRA analiza la señal para diferentes frecuencias con diferentes resoluciones. Este análisis es la idea básica detrás de la transformada Wavelet.

El MRA está diseñado para proporcionar una buena resolución temporal y pobre resolución en frecuencia para las altas frecuencias y buena resolución en frecuencia y pobre en tiempo para las bajas frecuencias.

Existen tres tipos de transformada Wavelet: continua (CWT), semidiscreta (SWT) y discreta (DWT).

Su principal diferencia es en la forma en la que sus parámetros de desplazamiento (tiempo) y escala (frecuencia) son discretizados.

### **2.3.2.1 Transformada Continua Wavelet**

La CWT es una técnica desarrollada como alternativa a la STFT para solucionar el problema de resolución (ver Ecuación 1).

Su análisis es similar al de la STFT debido a que la señal es multiplicada por una función (Wavelet madre) de manera similar a la función ventana en la STFT.

$$C_{(\tau,s)} = \int_{-\infty}^{+\infty} f(t) \varphi_{\tau,s}(t) dt$$

Ecuación 1 CWT

Donde:

$$\varphi_{\tau,s}(t) = \frac{1}{\sqrt{s}} \varphi \left| \frac{t-\tau}{s} \right|$$

La señal transformada  $C_{(\tau,s)}$  es una función de dos variables,  $\tau$  y  $s$ , los parámetros de traslación y escala respectivamente.  $\varphi_{\tau,s}(t)$  es la función de transformación que se le denomina “Wavelet madre”.

### 2.3.2.2 Transformada Discreta Wavelet

La DWT (ver Ecuación 2) es la versión discreta de la transformada Wavelet, esta es capaz de entregar suficiente información tanto para el análisis como para la reconstrucción de la señal con una significativa reducción del tiempo de procesamiento, además, es mucho más fácil de implementar que la CWT.

Wavelets debe de disponer de algoritmos rápidos para su uso en computadoras, es decir, un método similar al de la FFT (Fast Fourier Transform) para encontrar los coeficientes de Wavelet  $C_{[j,k]}$  y para reconstruir la función que representan.

$$C[j, k] = \sum_{n \in \mathbb{Z}} f[n] \varphi_{j,k}[n]$$

Ecuación 2 DWT

Donde  $\varphi_{j,k}[n]$  es una wavelet madre discreta.

Este proceso es el más efectivo para obtener los coeficientes  $C_{[j,k]}$  de la DWT. A continuación se explicará en que consiste:

La idea básica es la misma que se emplea en la CWT, obtener una representación tiempo – escala de una señal usando técnicas de filtrado digital.

En el caso discreto, se utilizan filtros con diferentes frecuencias de corte para analizar la señal en las diferentes escalas; de este modo la señal se pasa a través de una serie de filtro pasa alto para analizar las altas frecuencias y pasa bajo para las bajas frecuencias.

El procedimiento para obtener la DWT comienza pasando la señal (secuencia) a través de un filtro digital de pasa bajo y media banda con respuesta impulso  $h[n]$ , este proceso consiste en realizar matemáticamente la convolución de la secuencia con la respuesta impulso del filtro:

$$x[n] * h[n] = \sum_{-\infty}^{\infty} x[k] \cdot h[n - k]$$

Ecuación 3 DWT

Un filtro pasa bajo de media banda elimina todas las frecuencias que están por encima de la mitad de la mayor frecuencia de la señal, por ejemplo, si la señal tiene como máximo una componente de 1000Hz., este filtro eliminaría todas las frecuencias sobre los 500Hz.

Una vez que la señal ha pasado por el filtro pasa bajo de media banda, la mitad de las muestras se pueden eliminar.

Con este propósito se elimina una de cada dos muestras (submuestreo por 2) con lo cual se reduce el número de puntos a la mitad y la escala de la señal se duplica.

La DWT analiza la señal descomponiéndola en una aproximación y en un detalle (nivel). Con este propósito se emplean dos conjuntos de funciones denominadas: funciones de escalamiento y funciones Wavelets, las que están asociadas a filtros pasa bajos y pasa altos, respectivamente.

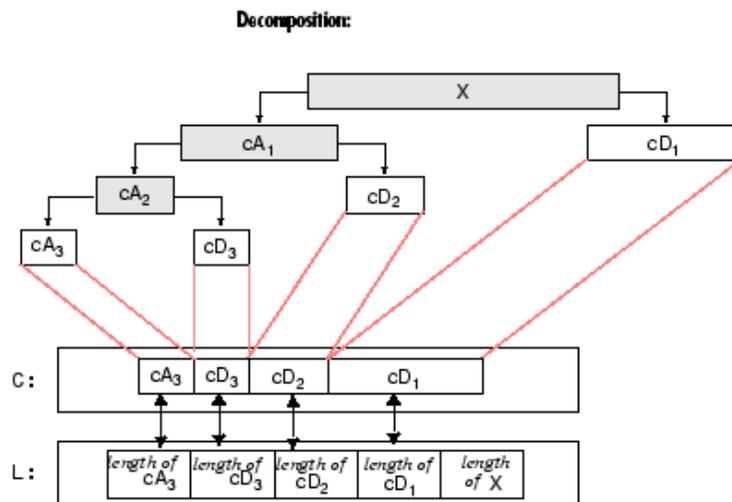


Figura 16. Filtro wavelet.

Donde:

- $x$  : Es la señal a descomponer.
- $cA_i$  : Coeficientes de aproximación en el nivel  $i$ .
- $cD_i$  : Coeficientes de detalles en el nivel  $i$ .
- $C$  : Vector de descomposición Wavelet.
- $L$  : Vector de registro o conteo.

## **2.4 Inteligencia Artificial**

Inteligencia artificial (IA) es la ciencia que estudia de forma sistemática el comportamiento de la inteligencia. Con el fin de imitar o simular habilidades humanas a través de la creación y el uso de maquinaria y computadoras.

Las principales áreas de estudio de la inteligencia artificial son los sistemas expertos, redes neuronales, lógica difusa, agentes inteligentes, robótica y lenguaje natural.

### **2.4.1 Aprendizaje automático**

Arthur Samuel (1959). Machine Learning: Campo de estudio que brinda a las computadoras la habilidad de aprender sin ser explícitamente programadas.

El aprendizaje automático o machine learning es una rama de la inteligencia artificial que se puede subdividir en dos áreas: aprendizaje supervisado y aprendizaje no supervisado.

El aprendizaje supervisado consiste en el entrenamiento de un sistema controlando su entrada y su salida. A partir de esto se hacen modificaciones a los parámetros propios de cada método de este tipo, para así poder lograr un sistema eficiente a diferentes tipos de bases de datos o problemas.

Ejemplos de métodos de aprendizaje automático son adaboost y SVM (por sus siglas en inglés *Support Vector Machine*).

El entrenamiento es la parte central de estos sistemas, es aquí donde se generan algoritmos controlados por un supervisor donde se modifican los parámetros

mencionados anteriormente. Estos parámetros pueden ser funciones de costo o pesos que se van actualizando de acuerdo al acierto o error en la salida del sistema.

El aprendizaje no supervisado se enfoca a la asociación o similitud entre de entrada y la información que se le proporcionó al inicio de su programación. Estos sistemas cuentan con un conjunto de reglas que le ayudan a familiarizar los patrones que se encuentran en un conjunto de datos.

#### 2.4.1.1 Adaboost

AdaBoost viene de **Adaptive Boosting** cuya idea es generar un clasificador fuerte y muy preciso a partir de clasificadores débiles.

Los clasificadores débiles se pueden ver como simples umbrales que sirven para determinar si una característica de un conjunto lo asocia a una clase u otra. Por lo cual se pueden tener tantos umbrales como características en una muestra y dependiendo del acierto o error en la clasificación de cada umbral se le asignan pesos, cuanto menor sea el peso menor es el error de este umbral.

Por esto un clasificador débil es un conjunto de umbrales con distintas distribuciones de pesos ver Ecuación 4.

$$D_1 = \frac{1}{m}$$

Ecuación 4 Clasificador débil.

Donde  $m$  es el número total de muestras para el entrenamiento y  $D$  es un vector de dimensiones iguales a la cantidad de umbrales determinados y que se va actualizando en cada iteración.

Este conjunto de pesos se aplica a cada una de las características de las muestras y determinan una hipótesis sobre la clase a la que pertenece la muestra, regularmente  $\{-1,1\}$ .

Posteriormente se calcula el error de la hipótesis por medio de la ecuación 5.

$$E_t = \sum h_t(x_i) \neq y_i$$

Ecuación 5 Hipótesis.

Donde:

$h_t$ : es el resultado  $\{-1,1\}$  del clasificador aplicado a la muestra.

$x_i$ : es la muestra de un conjunto de entrenamiento S.

$y_i$ : es la clase o etiqueta de entrenamiento  $\{-1,1\}$ .

Con ayuda del error encontrado se obtiene la variable alfa, cuya fórmula es:

$$\alpha_t = \frac{1}{2} \ln\left(\frac{1 - E_t}{E_t}\right)$$

Ecuación 6

Esta alfa se utiliza para la actualización de los pesos D que se aplicarán en la siguiente iteración del sistema.

El vector de pesos D actualizado se divide sobre la variable Z para normalizarlo, donde:

$$D_{t+1} = \frac{D_t(i) \exp(-\alpha_i y_i h_t(x_i))}{Z_t}$$

Ecuación 7

$$Z_t = \sum_{i=1}^m D_t(i) \exp(-\alpha_t y_i h_t(x_i))$$

Ecuación 8

Al acabar las iteraciones de entrenamiento se forma nuestro clasificador fuerte como:

$$H(x) = \sum_{t=1}^T \alpha_t h_t(x)$$

Ecuación 9

Donde T es el número de iteraciones determinadas.

#### 2.4.1.2 SVM

SVM (por sus siglas en inglés *Support Vector Machine*) es un método de aprendizaje automático supervisado. Este tiene gran eficiencia en la clasificación de datos no lineales. SVM también tiene un excelente comportamiento para datos líneas, ya que busca obtener fronteras de decisión, las cuales separan linealmente muchas soluciones.

Estas fronteras de decisión no son más que umbrales lineales que se usan para separar un clase de otra (ver figura 17).

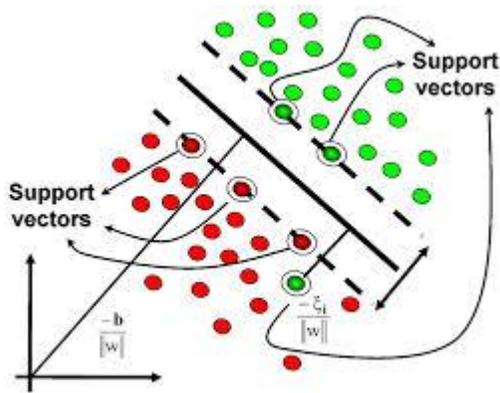


Figura 17. Ejemplo SVM.

La forma de encontrar las fronteras de decisión se basa en la función de costo, en la cual vamos ajustando la variable  $C$ . Con lo cual cuando nuestra  $C$  es muy grande nuestro sistema es muy sensible ante muestras que se encuentren fuera de nuestras fronteras o muy pequeña para ignorar estos datos.

Para problemas no lineales SVM se apoya de funciones conocidas como “kernels” los cuales actúan como nuevas características del conjunto de muestras. Esto sirve para llevar la muestra a una nueva dimensión donde sea más fácil clasificarla de forma lineal.

Un ejemplo de kernels es el Gaussiano, definido como:

$$\text{similaridad}(x, l) = e^{-\frac{\|x-l\|^2}{2\sigma^2}}$$

Ecuación 20

Donde tenemos el parámetro  $\sigma$  el cual, al igual que  $C$ , determina la “exigencia” entre muestras que se encuentran fuera de nuestras fronteras de decisión.

La similaridad no es más que la distancia entre una marca  $L$  con respecto a la muestra  $x$ .

## **2.5 Interfaz gráfica**

Es una ventana en una computadora o dispositivo electrónico, que sirve para la interacción entre el hombre y el dispositivo. Esta facilita la comunicación entre las dos partes ya que sirve como “traductor” entre un lenguaje y otro.

Son muy útiles debido a las grandes facilidades que proporcionan para controlar y entender diversos sistemas computacionales muy complejos. Estas ventanas están compuestas por al menos dos elementos esenciales: La ventana o interfaz que se presenta al usuario y los algoritmos que integran, controlan y muestran toda la información o procesos que se manejan en estas.

Dos herramientas poderosas que se combinan para crear interfaces complejas son el software matemático Matlab y VRML.

### **2.5.1 Matlab**

Matlab es un software ingenieril que cuenta con una gran cantidad de librerías y funciones capaces de hacer cálculos matemáticos y simulaciones reales muy complejas. De igual manera se pueden crear con gran facilidad funciones propias que ayuden a resolver problemas específicos.

Este software se encuentra disponible para diversos sistemas operativos y es muy usado principalmente para simulaciones, procesamiento digital de imágenes y señales. De forma paralela cuenta con cajas de herramientas o *toolbox* para crear interfaces gráficas.

Entre las cuales podemos destacar simulink y VRML.

### **2.5.2 VRML**

VRML (por sus siglas en inglés Virtual Reality Toolbox) sirve para conectar a simulink y Matlab con un software que permite la creación y manipulación de piezas o entornos gráficos.

Esta herramienta nos permite importar piezas diseñadas en casi cualquier programa tipo CAD. Esto le permite poder manipular el ensamble de las diferentes piezas que se importen, a través de nodos de conexión.

Posteriormente se puede exportar la figura ya ensamblada y articulada para su uso como interfaz gráfica en varios software como Matlab.

## 2.6 BCI

### 2.6.1 BCI: Concepto básico

BCI que por sus siglas en inglés significa Brain Computer – Interface, o en español ICC Interfaz cerebro – computadora. Consisten en sistemas que comunican hardware y software para así poder controlar diferentes tipos de dispositivos. La figura 18 nos permitirá mostrar de manera sencilla el proceso de BCI.



Figura 18. Proceso de sistema BCI.

### 2.6.2 BCI vs Neuroprótesis

BCI ha encontrado métodos no invasivos para ayudar a las personas con discapacidades para poder realizar tareas cotidianas con apoyo de prótesis robóticas o vehículos de apoyo.

Al igual que BCI o ICC (interfaz cerebro computadora) existen otros métodos que ayudan a combatir estas discapacidades. Uno de los que toma gran importancia son las neuroprótesis (ver Figura 19), esta consiste en la implantación de chips en el cerebro, lo que se busca con esto es reemplazar las funciones del cerebro que se encuentran dañadas, como motrices, visuales, etc.



Figura 19. Procesos de neuroprótesis.

Las principales diferencias entre las Neuroprótesis y BCI es que las neuroprótesis conecta al cerebro con un dispositivo físico, tal como lo son los chips de silicio que se utilizan, por lo cual son métodos invasivos. Por su parte BCI conecta el cerebro de manera no invasiva a un software que se encarga del procesar toda la información y mandarla al sistema a mover o actuador.

### 2.6.3 BCI: Aplicaciones

Este tipo de tecnología nos puede permitir usar dispositivos electrónicos, eléctricos y mecánicos que pueden sustituir algunas partes humanas, como pueden ser partes robóticas, o simplemente controlar algunos dispositivos ya existentes, pero que requieren el uso de extremidades que se han perdido total o parcialmente, como sillas de ruedas o automóviles.

## Capítulo 3 – Metodología

### 3.1 Base de datos

#### 3.1.1 Base de datos NeuroSky

Para crear la base de datos que servirá para realizar las pruebas del sistema a lo largo de este proyecto, se enlazará el sensor NeuroSky MindWave por medio de conexión bluetooth al software Matlab.. La base de datos consiste en 10 señales de relajación y 10 señales de concentración, con duración de 65 segundos cada una y una frecuencia de muestreo de 512 muestras por segundo.

Las señales utilizadas para el proceso de base de datos les será aplicado un filtro pasabandas Windowed.Sinc de  $[8 -30]$ Hz. Esto con el fin mantener el rango de frecuencias que corresponde a las ondas cerebrales a la que pertenecen los estados mentales de relajación y concentración.

Se crearán dos bases de datos, una de entrenamiento y una para pruebas. Ambas tendrán señales de concentración y relajación seleccionadas aleatoriamente. La base de datos de entrenamiento contendrá el 70% de las señales de concentración y relajación. La base de datos para prueba tendrá las 30 señales restantes.

Cada una de las señales será cortada en un número menor de muestras para su mejor análisis. A través del método Sliding window con diferentes tamaños y desplazamientos. Posteriormente se reordenarán al azar todas las muestras.

En la base de datos se aplicará la DWT buscando obtener solo los coeficientes de detalles de nuestro interés. Estos sería los pertenecientes en el rango de  $[0 - 30]$ Hz. Una vez obtenidos estos coeficientes se usarán para la generación de nuevas características con métodos estadísticos. Se obtendrán la media, media absoluta, desviación estándar, desviación estándar absoluta, potencia promedio y razón entre medias de cada nivel y medias absolutas.

Por último se integrarán cada uno de los datos obtenidos por nivel en un vector donde se encontrarán concatenados los valores de los coeficientes de detalle previamente seleccionados.

*VectorDWT*

$$= [An \ Dn \ \dots \ D1 \ media_n \ \dots \ media_1 \ mediabs_n \ \dots \ mediabs_1 \ dsvstd_n \ dsvstdabs_n \ \dots \ dsvstdabs_1 \ razón_n \ \dots \ razón_1 ]$$

### 3.1.2 Base de datos comparativa

Para poder hacer una validación correcta de los datos obtenidos en el sistema se realizó la investigación e implementación de una base de datos utilizada en sistemas con etapas parecidas. El objetivo era comparar los resultados de otro sistema y el nuestro a partir de los mismos datos de entrada.

La base de datos utilizada, adicionalmente a la anterior planteada, fue desarrollada por la universidad de Graz en Austria en el año 2012. El nombre de la base de datos es “Autocalibration and Recurrent Adaptation”, contiene señales por parte de 12 sujetos, donde cada uno aporta 13 señales EEG obtenidas con una tasa de muestreo de 512Hz y un filtro pasabandas de 0.5 a 100 Hz.

El estudio que utiliza esta base, y que sirve para la comparación de resultados, es una tesis de maestría de la universidad de Aalborg en Dinamarca. El nombre de la tesis es “Feasibility of using Error-related potentials as an appropriate method for adaptation in a brain-computer interface”.

### **3.2 Clasificación de datos Adaboost**

En esta etapa del proyecto se clasificará cada una de las muestras de entrenamiento para obtener un modelo efectivo para la clasificación de datos de prueba y datos de nuevo ingreso.

Para encontrar el mejor modelo se probarán algunas de las combinaciones posibles del número de iteraciones para entrenamiento de adaboost, el tamaño de la ventana de sliding window y el tipo de wavelet madre que se le aplicará al sistema.

Se obtendrán matrices de resultados, donde las filas representarán el número de iteraciones y las columnas el tipo de wavelet madre. Este proceso se repetirá para los diversos tamaños de ventanas que se desean estudiar.

Con los cuales se encontrará la mejor combinación de parámetros de iteraciones, wavelet madre. Todo esto para poder obtener el mejor modelo de entrenamiento para cada tamaño de ventana.

Una vez obtenidos los modelos, podremos obtener de estos los índices de columnas más importantes para la asociación o distinción de clases. Esto será de mucha utilidad para la selección de características.

### **3.3 Selección de características Adaboost**

Con la información obtenida por el modelo de adaboost, previamente entrenado para la clasificación de muestras de entrenamiento, se podrá tener acceso a los índices de las características con mayor peso para la clasificación de las muestras.

Con estos índices se generará nuevas muestras de entrenamiento y de test, que reducirán su número de características original al número y características determinadas con ayuda del modelo.

Tantas versiones de reducción de características de acuerdo a la cantidad de sliding Windows creadas previo a la DWT.

Este proceso se llevará acabo para poder entrenar de mejor forma SVM.

### **3.4 SVM**

En esta etapa del proyecto se generará un modelo para la clasificación de los datos de entrenamiento que tuvieron una reducción de características en el paso anterior. Al igual que en adaboost, se hará un entrenamiento previo para encontrar los mejor parámetros, en este caso C y sigma, que nos entregan el modelo con menor error.

Con ayuda de estos valores encontrados se generará el modelo de clasificación de SVM con ayuda del kernel gaussiano. Se generará un modelo por cada sliding window hecha. Posteriormente se aplicará a las muestras de entrenamiento para así obtener el error del modelo.

El mejor modelo obtenido se aplicará con la muestras de nuevo ingreso

### **3.5 Interfaz gráfica**

Se creará una interfaz gráfica en Matlab con ayuda de VRML. El interfaz es un vehículo que recorre una carretera. Esto se hará para poder probar los comandos creados con ayuda de las señales previamente obtenidas, descompuestas y clasificadas.

El vehículo podrá hacer dos movimientos, izquierda con concentración y derecha con relajación. De igual forma con ayuda de un guiño con alta amplitud se podrá cambiar el sentido de los movimientos a arriba o abajo.

## Capítulo 4 – Desarrollo del Proyecto

### 4.1 Base de datos

#### 4.1.1 Base de datos local

Las señales de concentración y relajación obtenidas tienen las siguientes características:

- Rango de frecuencia concentración: [14 - 30]Hz
- Método de obtención de concentración: Lectura a voz media.
- Rango de frecuencia concentración: [8 - 14]Hz
- Método de obtención de relajación: Cerrar ojos mientras se lleva un conteo en la mente sumado al control de la respiración.
- Número de sujetos: 1.
- Señales obtenidas por sujeto: 14 señales crudas.
- Duración de la señal: 1.04 segundos (32768 muestras).
- Frecuencia de muestreo: 512 muestras por segundo.

Estas señales fueron guardadas en formato .txt de la siguiente forma “EstadoSujeto#dePrueba.txt”. De igual manera a las señales utilizadas para el proceso de base de datos les fue aplicado un filtro pasabandas de [8 -30]Hz. En la tabla 1 se muestra el orden en el que fueron guardadas y el índice de cada señal.

Señal	Índice
RelaxRafael1.txt	1
RelaxRafael2.txt	2
RelaxRafael3txt	3

RelaxRafael4.txt	4
RelaxRafael5.txt	5
RelaxRafael6.txt	6
RelaxRafael7.txt	7
FocusRafael1.txt	1
FocusRafael2.txt	2
FocusRafael3.txt	3
FocusRafael4.txt	4
FocusRafael5.txt	5
FocusRafael6.txt	6
FocusRafael7.txt	7

Tabla 2. Base de datos.

Se concatenan todas las señales en una matriz. De esta forma se crea la matriz DB con dimensiones [14 x 33312]. Donde las primeras 7 señales son las correspondientes a los índices de relajación y las últimas 7 de concentración.

La función SlidingWindow.m se encargará de generar “fotos” o pequeños cortes a la señal. Esto se hizo con diferentes dimensiones de muestras y corrimientos. Además esta función genera un reacomodo de las muestras de la base de datos de forma aleatoria.

Aplicando la función anterior con una ventana de tamaño 8192 muestras, sin corrimientos, obtenemos una nueva base de datos con dimensiones [56 x 8192]. Obteniendo así un mayor cantidad de muestras para generar la base de datos para entrenamiento y la base de datos para pruebas.

Dividimos nuestra base de datos con 70% de las muestras para el entrenamiento y 30% para pruebas. Obteniendo las siguientes bases de datos con dimensiones dataTrain [40 x 8192] y dataTest [16 x 8192]. De igual manera se crearon los vectores de clases para cada una de la muestras de las bases de datos.

A partir de la base de datos de entrenamiento se generaron 56 bases de datos para probar la mejor combinación de sliding window y wavelet madre. Las combinaciones fueron de la siguiente manera (ver tabla).

	Tamaños de ventanas de sliding window en segundos							
	2	4	6	8	10	12	14	16
Wavelet madre								
Coiflet 5	x	x	x	x	x	x	x	x
Symlet 4	x	x	x	x	x	x	x	x
Symlet 5	x	x	x	x	x	x	x	x
Symlet 6	x	x	x	x	x	x	x	x
Db6	x	x	x	x	x	x	x	x
Db40	x	x	x	x	x	x	x	x
Db45	x	x	x	x	x	x	x	x

Tabla 3. Combinaciones wavelet madre.

Una vez aplicada la DWT a cada muestra de la base de datos, el resultado se concatena en una matriz. Esta sería ya nuestra base de datos entrenamientos, que tiene una dimensión de [40 x 65536] y la base de prueba de [16 x 65536].

Para la base de datos se toman A7, D7, D6, D5 y D4(ver tabla 4), debido a que estos cumplen con el intervalo de frecuencias de [1 – 32] Hz.

Nivel	Rango (Hz)
D1	128 – 256
D2	64 – 128
D3	32 – 64
D4	16 – 32
D5	8 – 16
D6	4 – 8

D7	2 – 4
A7	1 – 2

Tabla 4. Niveles de detalles.

Para cada nivel de coeficientes de detalles se calculan la media, media absoluta, desviación estándar, desviación estándar absoluta, potencia promedio, la razón entre medias y la razón entre medias absolutas.

Estos datos estadísticos se agregan a los niveles obtenidos de coeficientes de detalles, convirtiéndose en nuevas características de la muestra.

Este proceso se conoce como la etapa de calibración del sistema y nos ayudó a poder tener mejores resultados con señales de diferentes sujetos.

*VectorDWT*

$$= [A7 \ D7 \ \dots \ D4 \ media_{A7} \ \dots \ media_{D4} \ mediabs_{A7} \ \dots \ mediabs_{D4} \ dsvstd_{A7} \ dsvstd_{D4} \ dsvstdabs_{A7} \ \dots \ dsvstdabs_{D4} \ razón_{A7} \ \dots \ razón_{D4} ]$$

Una vez obtenido el VectorDWT de cada muestra, se concatenan en una nueva matriz.

Por lo cual tendremos 56 combinaciones de conjunto de datos para esta base de datos, cada una con los diferentes valores de sliding window y wavelet madre.

#### 4.1.2 Base de datos comparativa

En el caso de la base de datos comparativa, no se utiliza un proceso de adquisición de datos, sin embargo, si se tiene que realizar el proceso de ventaneo, filtrado, calibración y extracción de patrones a como se realizó en el punto 4.1.1.

## 4.2 Clasificación de datos Adaboost

Con cada una de las combinaciones obtenidas en la etapa anterior se realizó un entrenamiento para buscar el mejor modelo de clasificación con adaboost. Esto se realizó mediante un ciclo en el cual se probaba la matriz con su combinación de tamaño de sliding window y wavelet madre (ver tabla 3) y los valores determinados en un vector que contenía el número de iteraciones para el entrenamiento. Las iteraciones fueron de 100, 200, 300, 400, 500, 600, 700, 800, 900 y 1000.

Por cada iteración se genera un resultado de precisión en una posición (i,j) donde i es el tipo de wavelet madre usada y j es la cantidad de iteraciones para el entrenamiento. Este proceso nos regresa matrices de [7 x 10] en las que se encuentran todas las precisiones para un tipo de slidingwindow.

La prueba se realizó para slidingwindow de 2, 4 ,6, 8, 10, 12, 14, 16 segundos. Al final de cada iteración se guardaron los resultados. Al finalizar todas las pruebas se buscó la mejor precisión en para cada uno de los sliding window usados. Al encontrar la mejor precisión también se obtuvo la combinación de wavelet madre e iteraciones más eficiente para dicho sliding window.

Por último se tomaron dichos parámetros con los que se obtuvo un modelo de adaboost por cada sliding window. Los modelos se guardaron y así se puedan usar para clasificar las pruebas y otras señales de nuevo ingreso.

## 4.3 Selección de características Adaboost

A partir de los modelos obtenidos, en el entrenamiento para la clasificación de datos con adaboost, podemos conocer las columnas de características que tienen un mayor peso en la clasificación de muestras. Con estas se reducen nuestras matrices DBnormal y DBestadístico en sus características para ser más eficientes al entrenar SVM.

El proceso de reducción de características inicia capturando todos los valores “*idx*” que se encuentran en la estructura del modelo. Posteriormente se ordenan de menor a mayor y eliminamos todos los repetidos.

Posteriormente se crearon las nuevas matrices con los índices sobrantes.

#### **4.4 Clasificación de datos SVM**

Recibiendo los datos de entrenamiento previamente seleccionados en el paso anterior, en SVM se normalizaron para regularizar todas las características de las muestras. Al igual que en adaboost se entrenó el sistema para encontrar los valores de C y sigma que proporcionen el menor error de entrenamiento.

Las combinaciones de C y sigma que se probaron fueron las siguientes:

C = [0.025 0.05 0.1 0.3 0.5 1 1.5 2 2.5 3 3.5 4 4.5 5 5.5 6 10 20 30];

Sigma = [0.025 0.05 0.1 0.3 0.5 0.7 1 1.5 2 2.5 3 3.5 4 4.5 5 5.5 6 6.5 7];

Una vez encontradas los mejores valores de estos parámetros se entrenó nuevamente con estos para encontrar el modelo de clasificación. Se encontró un modelo para cada sliding window y se probó el modelo con el conjunto de datos de prueba. Todos los modelos fueron guardados.

#### **4.5 Interfaz gráfica**

Para la interfaz gráfica del vehículo se implementó un código que uniera la captura de la señal cruda, el filtro pasabandas, la función de sliding window, la aplicación de

la DWT, el método de clasificación y las condicionales que determinan cuanto y hacia donde se debe de mover el nodo que sea en el interfaz.

Durante este proceso se tomó una señal cruda de 5 minutos de tiempo, aunque se mueve el cubo cada dos segundos. Esto para ayudar a que el usuario pueda obtener el valor deseado para la posición del nodo que se desea mover.

Para el vehículo el usuario tenía dos comandos, relajación para movimiento a la derecha y concentración para movimiento a la izquierda. Además se pudo cambiar de orientación al cubo para movimientos hacia arriba con relajación y hacia abajo con concentración. Esto con ayuda de un guiño voluntario por parte del usuario que sobre pase un umbral.

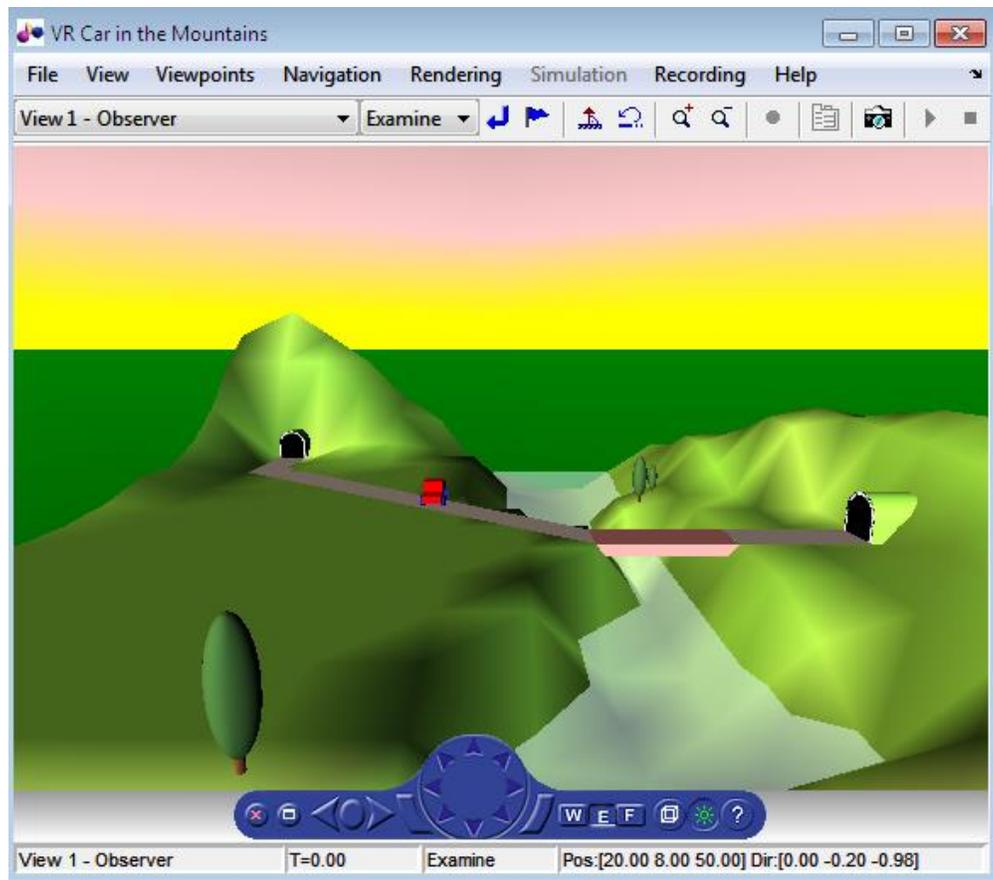


Figura 20. Resultados Coiflet 5.

## Capítulo 5 – Resultados

En las siguientes gráficas y tablas se muestran los porcentajes de aciertos obtenidos usando siete tipos diferentes de wavelet madre para la extracción de características y el método adaboost como clasificador de las muestras en la base de datos de prueba.

A la base de datos se le aplicó un sliding window de dos segundos y se realizaron desde 100 hasta 1000 iteraciones para el entrenamiento de adaboost. Las gráficas con los resultados para un sliding window de cuatro, seis, ocho, diez, doce y catorce segundos se pueden encontrar en el anexo I.

La figura 21 muestra el resultado de la wavelet madre Coiflet de nivel 5 para diez diferentes iteraciones.

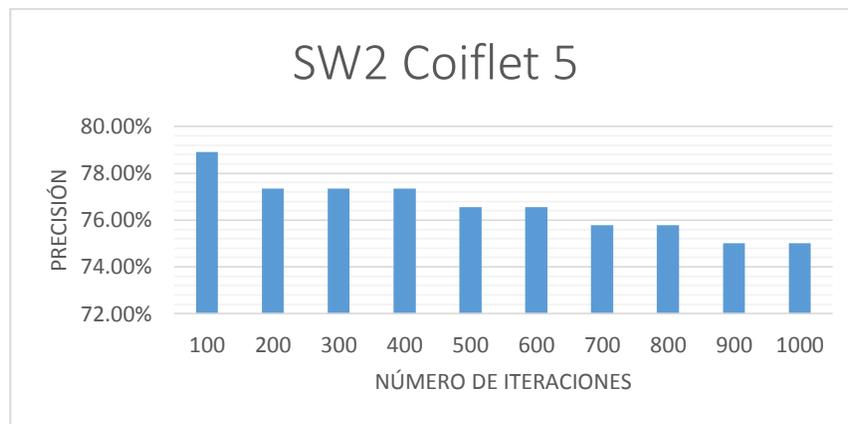


Figura 21. Resultados Coiflet 5.

Como se puede ver en la figura 21 Coiflet 5 presenta un mejor acierto con 100 iteraciones en el entrenamiento de adaboost.

La figura 22 muestra el resultado de la wavelet madre Symlet de nivel 4 para diez diferentes iteraciones.

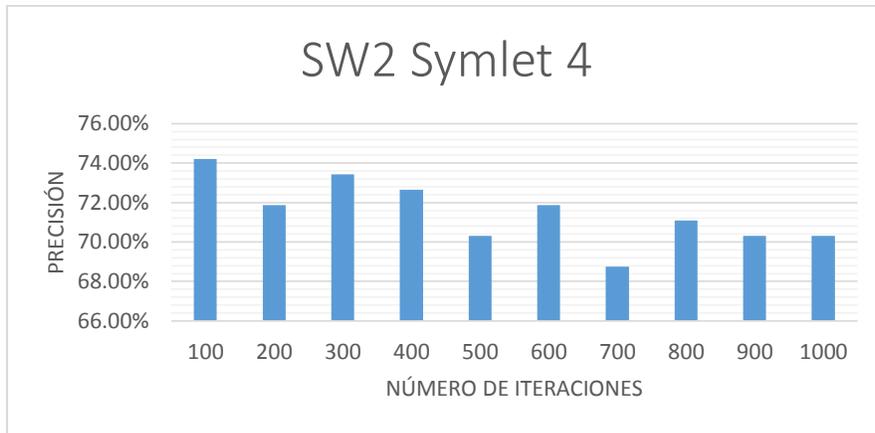


Figura 22. Resultados Symlet 4.

La figura 23 muestra el resultado de la wavelet madre Symlet de nivel 5 para diez diferentes iteraciones.

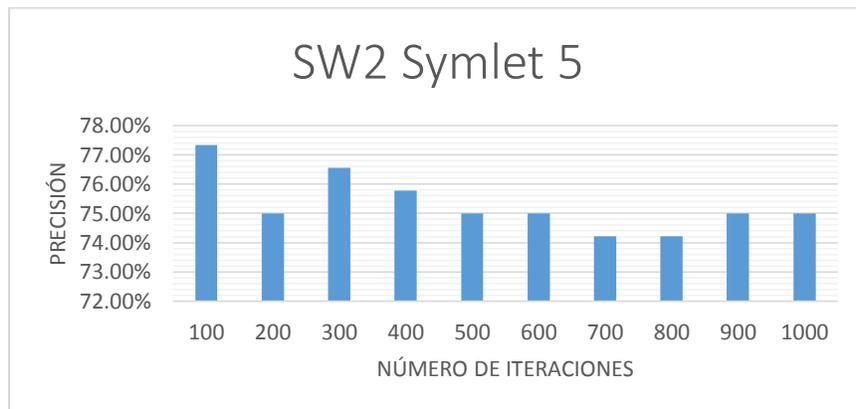


Figura 23. Resultados Symlet 5.

La figura 24 muestra el resultado de la wavelet madre Symlet de nivel 6 para diez diferentes iteraciones.

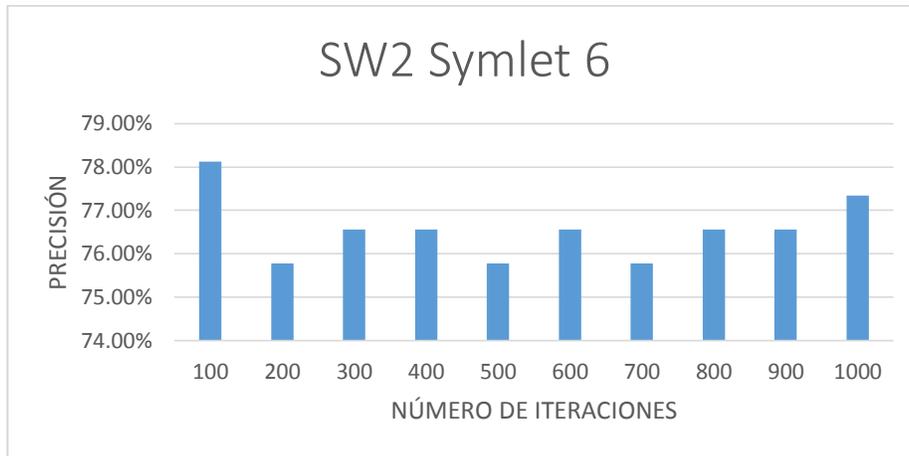


Figura 24. Resultados Symlet 6.

La figura 25 muestra el resultado de la wavelet madre Daubechies de nivel 6 para diez diferentes iteraciones.

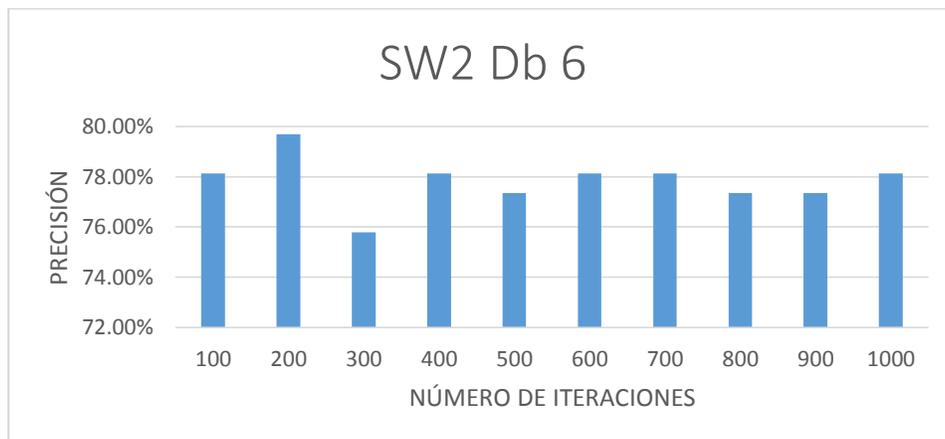


Figura 25. Resultados db 6.

La figura 26 muestra el resultado de la wavelet madre Daubechies de nivel 40 para diez diferentes iteraciones.

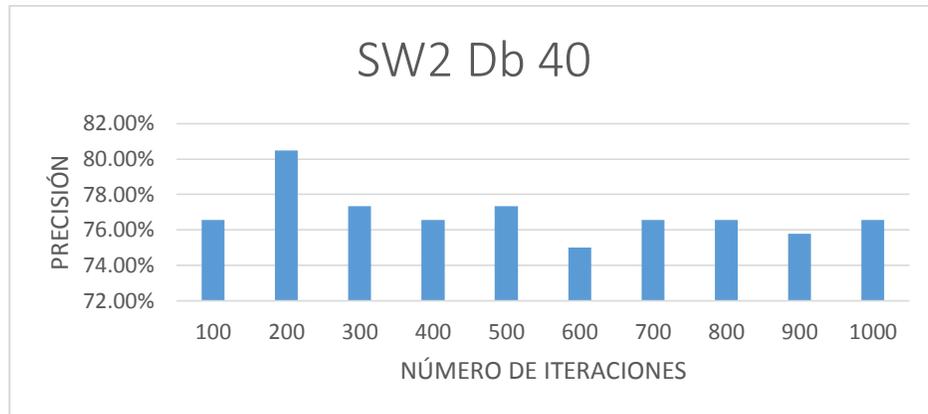


Figura 26. Resultados db 40.

La figura 27 muestra el resultado de la wavelet madre Daubechies de nivel 45 para diez diferentes iteraciones.

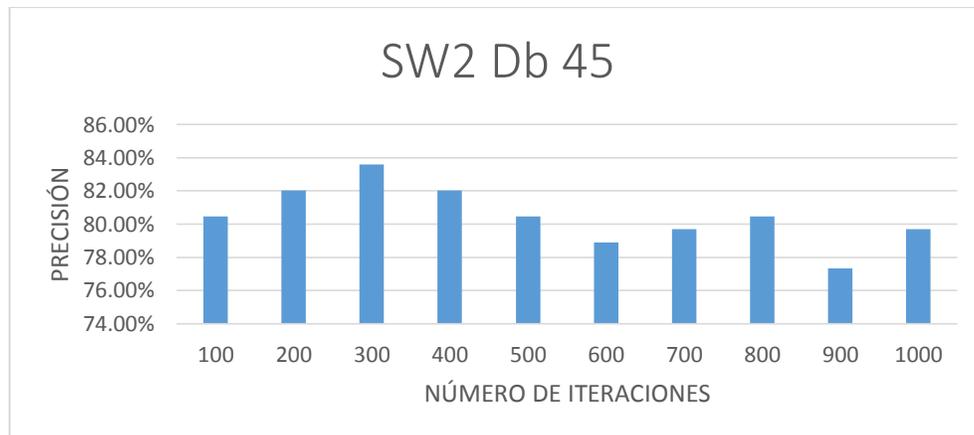


Figura 27. Resultados db 45.

En la figura 28 se comparan todos los resultados obtenidos en la aplicación de las siete wavelets madre al sistema de clasificación Adaboost para un sliding window de 2 segundos.

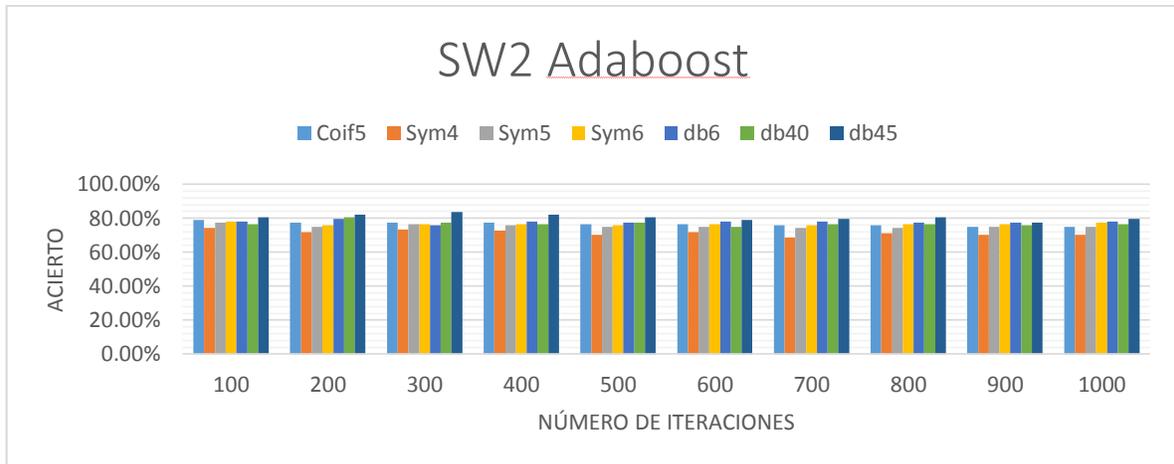


Figura 28. Resultados wavelet Adaboost.

En las siguientes gráficas y tablas se muestran los porcentajes de aciertos obtenidos usando siete tipos diferentes de wavelet madre para la extracción de características y el método SVM como clasificador de las muestras en la base de datos de prueba.

A la base de datos se le aplicó un sliding window de dos segundos y se realizaron desde 100 hasta 1000 iteraciones para el entrenamiento de SVM.

La figura 29 muestra el resultado de la wavelet madre Coiflet de nivel 5 para diez diferentes iteraciones.

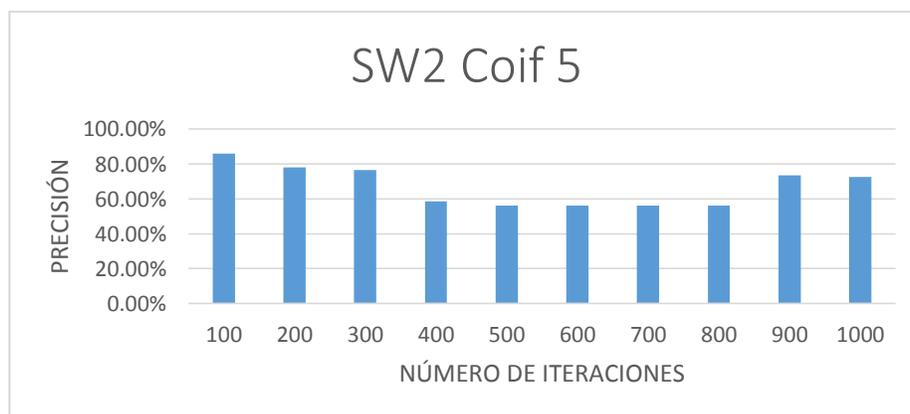


Figura 29. Resultados Coiflet 5 SVM.

La figura 30 muestra el resultado de la wavelet madre Symlet de nivel 4 para diez diferentes iteraciones.

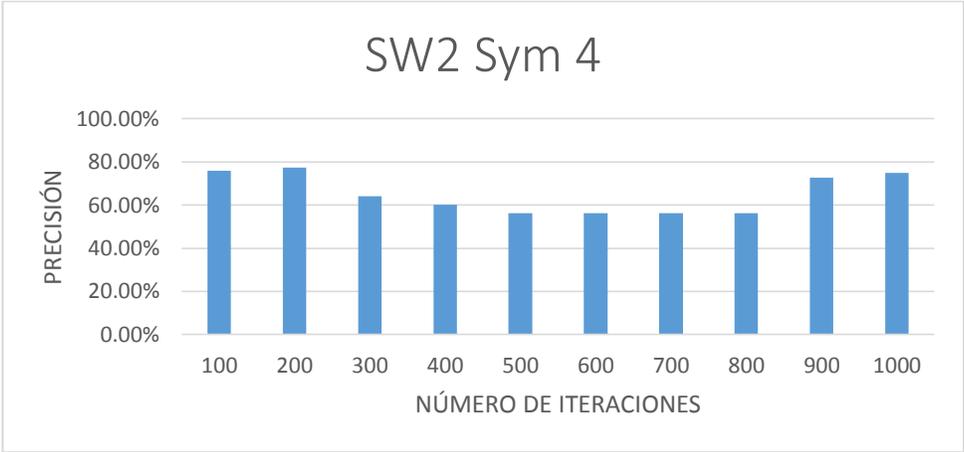


Figura 30. Resultados Symlet 4 SVM.

La figura 31 muestra el resultado de la wavelet madre Symlet de nivel 5 para diez diferentes iteraciones.

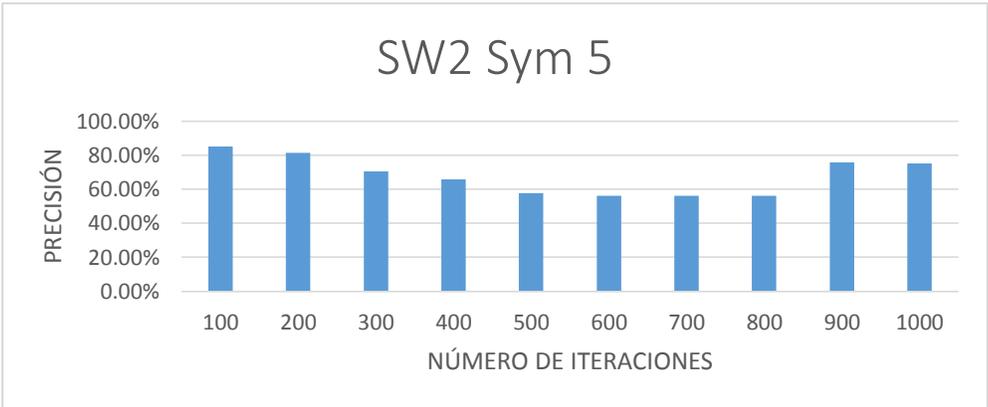


Figura 31. Resultados Symlet 5 SVM.

La figura 32 muestra el resultado de la wavelet madre Symlet de nivel 6 para diez diferentes iteraciones.

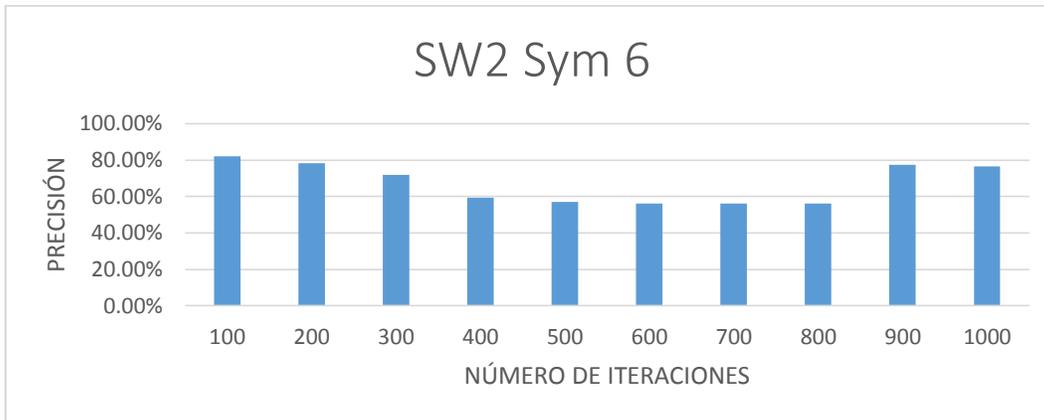


Figura 32. Resultados Symlet 6 SVM.

La figura 33 muestra el resultado de la wavelet madre Daubechies de nivel 6 para diez diferentes iteraciones.

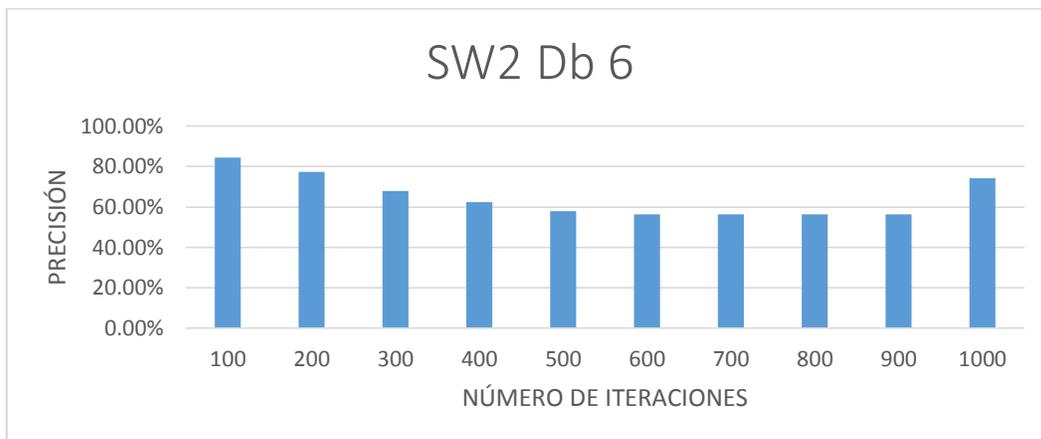


Figura 33. Resultados db 6 SVM.

La figura 34 muestra el resultado de la wavelet madre Daubechies de nivel 40 para diez diferentes iteraciones.

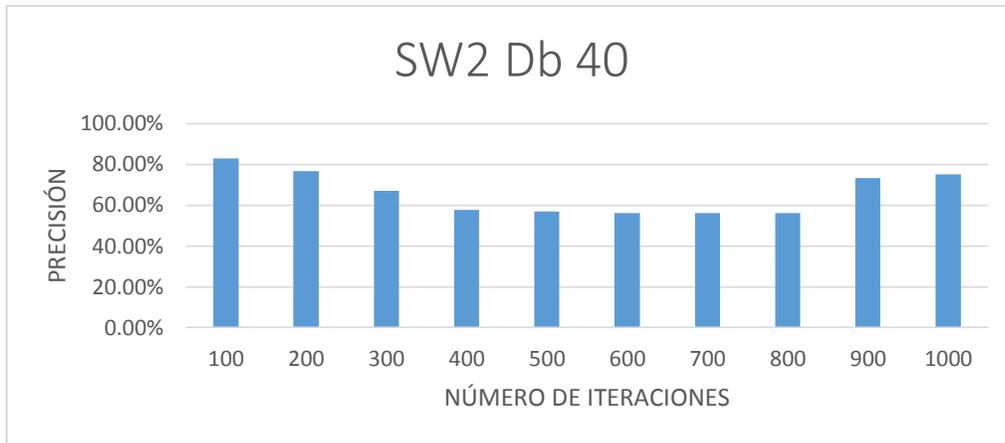


Figura 34. Resultados db 40 SVM.

La figura 35 muestra el resultado de la wavelet madre Daubechies de nivel 45 para diez diferentes iteraciones.

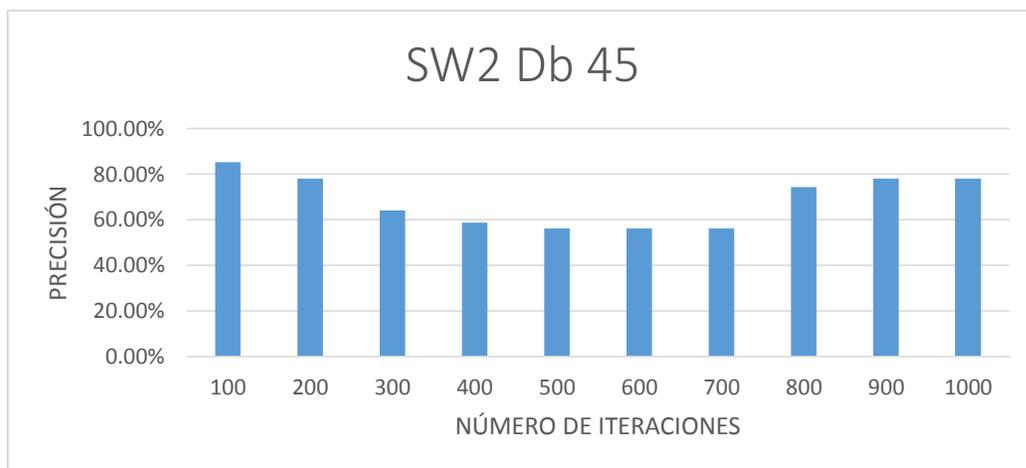


Figura 35. Resultados db 45 SVM.

En la figura 36 se comparan todos los resultados obtenidos en la aplicación de las siete wavelets madre al sistema de clasificación SVM para un sliding window de 2 segundos.

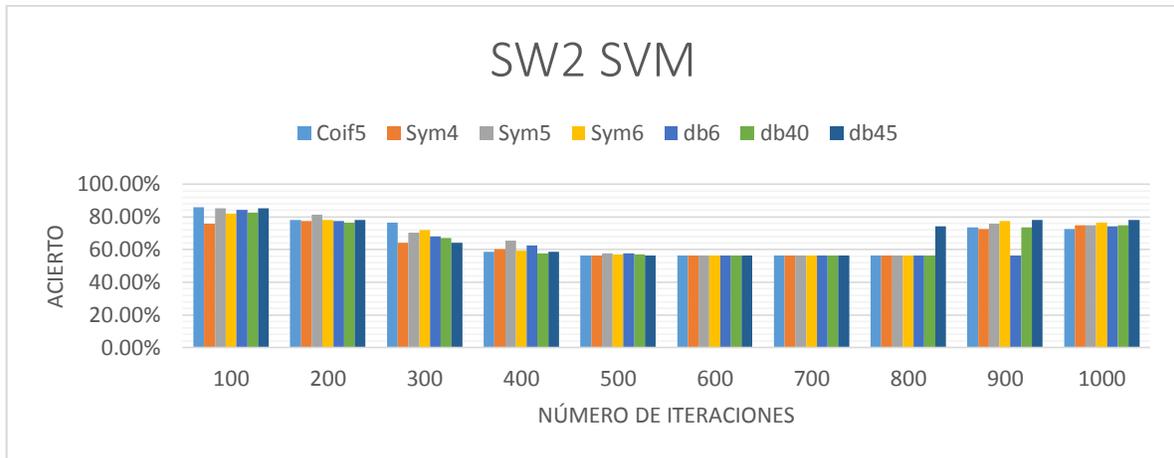


Figura 36. Resultados wavelet SVM.

En la tabla 5 se comparan los mejores resultados de Adaboost vs SVM.

Clasificación Adaboost vs SVM				
Segundos	Iteraciones	Wavelet madre	Acierto Adaboost %	Acierto SVM %
2	300	db45	83.59	76.56
4	600	Symlet 6	92.19	87.51
6	100	Coiflet 5	95.83	91.67
8	100	db40	100	100
10	100	Coiflet 5	96.88	96.87
12	100	Coiflet 5	96.88	93.75
14	100	Symlet 6	87.5	75
16	100	Coiflet 5	100	100

Tabla 5. Mejor Adaboost vs SVM.

Se demuestra que existe una superioridad por parte del algoritmo Adaboost para la clasificación muestras en este sistema.

En la tabla 6 se comparan los mejores resultados de SVM vs Adaboost.

Clasificación Adaboost vs SVM				
Segundos	Iteraciones	Wavelet madre	Acierto Adaboost %	Acierto SVM %
2	100	Symlet 4	83.59	87.6
4	200	Symlet 4	92.19	91.21
6	200	Coiflet 5	95.83	91.67
8	100	Db6	100	100
10	100	Db45	96.88	96.87
12	300	Coiflet 5	96.88	93.75
14	100	Symlet 6	87.5	75
16	100	Coiflet 5	100	100

Tabla 6. Mejor SVM vs Adaboost.

Se demuestra que a pesar de comparar los mejores resultados de SVM vs Adaboost, este último sigue siendo mejor. Por su puesto se tiene que tomar en cuenta que la diferencia no es muy amplia y quizá pueda no afectar demasiado en el resultado final de la clasificación.

De forma adicional, se realizó la aplicación de la base de datos comparativa. La cual nos da un parámetro para validar nuestro sistema a como se muestra a continuación.

En la tabla 7 y Figura 37 se muestran los resultados obtenidos por la tesis de referencia.

Resultados tesis de referencia						
OAP	77%	76%	80%	74%	78%	77%
CAP	83%	78%	80%	78%	78%	83%
GP	72%	70%	74%	73%	81%	72%
ICA	79%	79%	82%	81%	82%	79%
PCA	<b>88%</b>	85%	85%	85%	84%	<b>88%</b>
Single	80%	78%	79%	76%	83%	80%
LDA	57%	62%	55%	55%	56%	57%

Tabla 7. Tesis de referencia.

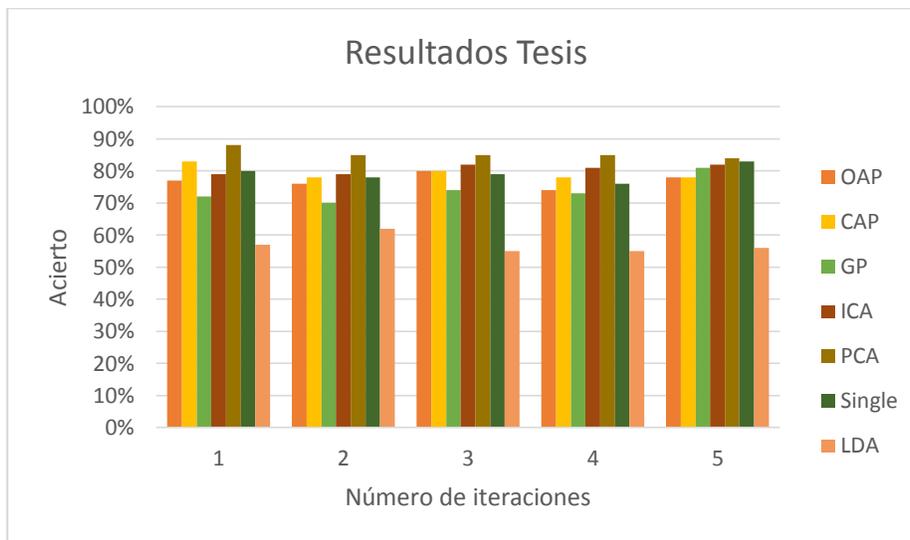


Figura 37. Tesis de referencia.

En la tabla 8 y Figura 38 se muestran los resultados obtenidos por el sistema local. Como podemos apreciar, nuestro sistema presenta resultados muy positivos a 400 iteraciones con un ventaneo de 12 segundos y utilizando la Wavelet madre Symlet 4.

A pesar de esto, cabe destacar que nuestro sistema requiere señales más largas y un mayor número de iteraciones en comparación con el resultado obtenido por los métodos aplicados en la tesis de referencia.

Resultados de sistema local con base de datos comparativa					
Iteraciones	100	200	300	400	500
Coif5	69%	72%	75%	72%	72%
Sym4	84%	84%	84%	<b>88%</b>	88%
Sym5	75%	72%	72%	75%	75%
Sym6	66%	66%	72%	72%	72%
db6	75%	75%	69%	66%	66%
db40	81%	72%	72%	72%	72%
db45	66%	75%	72%	75%	72%

Tabla 8. Resultados de sistema local.

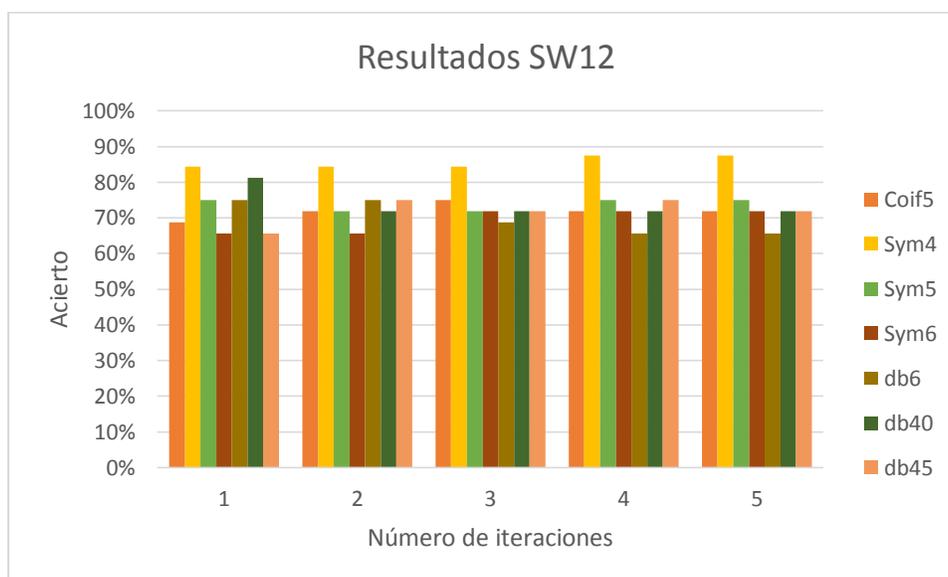


Figura 38. Resultados sistema local.

Se realizó un análisis de sensibilidad y especificidad de los resultados obtenidos por nuestro sistema como se muestra a continuación.

Análisis de sensibilidad y especificidad	
Mejor método de extracción	Symlet 4
Mejor método de selección de patrones	Adaboost
Mejor método de clasificación	Adaboost
Número de iteraciones	400
Tiempo de ejecución	12 segundos

Porcentaje de acierto	88%
Sensibilidad	94%
Especificidad	82%

Tabla 9. Análisis de sensibilidad y especificidad.

Este análisis nos permite tener fiabilidad en los resultados obtenidos por el sistema.

## Capítulo 6 – Conclusiones

Se utilizaron dos bases de datos de señales EEG. La primera contenía 10 señales de relajación y 10 señales de concentración adquiridas de 5 sujetos. Esta se generó a partir de la adquisición de cada señal con el sensor MindWave Mobile de la marca NeuroSky. La segunda base de datos ya existía y se utilizó para poder comparar un proyecto con un diseño similar al nuestro. Esta contaba con señales de estos mentales iguales a los adquiridos en la primera base de datos.

Las señales adquiridas en la base de datos necesitaban una etapa de procesamientos y extracción de datos para poder descomponerse en las señales eléctricas alfa, beta, delta y gama y teta. Siendo estas las portadoras de la información de los estados mentales de interés. La utilización de la etapa de pre-procesamiento y extracción de datos utilizando el método Wavelets, permitieron obtener las señales eléctricas con una mayor delimitación para poder clasificarlas entre cada una de ellas.

Los estados mentales utilizados para este proyecto fueron complicados de clasificar, debido a la cercanía que tienen entre ellas en términos de sus frecuencias. Para ello se utiliza una etapa de selección de características para poder detectar los patrones más relevantes para la clasificación de las señales. Adaboost obtuvo el mejor desempeño en el análisis de acierto y tiempo computacional requerido.

El algoritmo de clasificación Adaboost presentó excelentes resultados para el sistema. Esto se debe a que requiere que se ingrese una cantidad mínima de parámetros para poder obtener el mejor modelo.

La comparación de nuestro sistema con la etapa de auto-calibración y el de la tesis de referencia, utilizando la misma base de datos, nos permite dar veracidad a nuestro trabajo. Esto se debe a los porcentajes de aciertos altos, mayores al 80%, que pudimos obtener.

## Referencias

- [1]. Alegre, R. (2013). Interfaz Mental para el control de aplicaciones. Tesis de licenciatura. Universidad Politécnica de Valencia.
- [2]. Asadi, A., Mohsen, M., Ahmadi, A., Navidi, H., Broumandnia, A. (2012). Brain Computer Interface with Wavelets and Genetic Algorithms. *Young Researchers Club, Islamic Azad University*, 119 - 136.
- [3]. Bashashati A., Fatourehchi, M., Ward, R., Birch, G. (2007). A survey of signal processing algorithms in brain – computer interfaces base don electrical brain signals. *IOP Publishing Ltd, 07 (26)*, 32 – 47.
- [4]. Bashashati A., Fatourehchi, M., Ward, R., Birch, G. (2007). A survey of signal processing algorithms in brain – computer interfaces base don electrical brain signals [Figura]. Recuperado de <https://scholar.google.com.mx/>
- [5]. Chen, B. (2010). The Most Significant Gadgets of 2010 [Figura]. Recuperado de <http://www.wired.com/2010/12/top-tech-2010/?pid=928>
- [6]. Costagliola, S., Dal Seno, B., Matteucci, M. (2009). Recognition and Classification of P300s in EEG Signals by Means of Feature Extraction Using Wavelet Decomposition. *IEE, 1(9)*, 597 – 602.
- [7]. Fan, N., Balan, R., Rosca, J. (2011). Comparison of Wavelet and FFT based single channel speech signal noise Reduction Techniques. Siemens Corporate Research Inc.
- [8]. Guomundsdottir, K. (2011). Improving player’s control over the Neurosky Brain – Computer Interface. Bachelor’s Thesis. Reykjavik University School of Computer Science.
- [9]. INEGI. (2010). Discapacidad en México [Figura]. Recuperado de <http://cuentame.inegi.org.mx/poblacion/discapacidad.aspx?tema=P>

- [10]. INEGI. (2010). Discapacidad en México. INEGI. <http://cuentame.inegi.org.mx/poblacion/discapacidad.aspx?tema=P>
- [11]. Larsen, A. (2011). Classification of EEG Signals in a Brain – Computer Interface System. Master’s Thesis. Norwegian University of Science and Technology.
- [12]. Minguez, J. (2010). Tecnología de Interfaz Cerebro – Computador. Universidad de Zaragoza, España.
- [13]. Rojas, R. (2013). Semi-Autonomous Car Control Using Brain Computer Interfaces [Figura]. Recuperado de <https://scholar.google.com.mx/> .
- [14]. Sminkey, L. (2012). Informe Mundial sobre la discapacidad. Organización Mundial de la Salud. [http://www.who.int/disabilities/world\\_report/2012/es/](http://www.who.int/disabilities/world_report/2012/es/)
- [15]. Unhofer-Gesellschaft. (2007). Brain2Robot [Figura]. Recuperado de <http://phys.org/news/2007-11-brain2robot.html>
- [16]. Xu, Q., Zhou, H., Wang, Y., Huang, J. (2009). Fuzzy support vector machine for classification of EEG signals using wavelet – based features. *Medical Engineering & Physics*, 858 – 865.
- [17]. Yong, A., Hurley, J., Silvestre, M. (2011). Single trial EEG classification for Brain – Computer Interface using Wavelet Decomposition. Department of computer Science, University College Dublin.