

*Diseño de una herramienta computacional para estimar
el COP de una persona a través de una plataforma Wii
Balance Board.*

JENNY KATHERINE TORRES ZAMBRANO
JULIAN OSWALDO PEREZ DE LA CUADARA

ESTUDIANTES DE INGENIERÍA ELÉCTRICA POR CICLOS PROPEDÉUTICOS.



UNIVERSIDAD DISTRITAL FRANCISCO JOSÉ DE CALDAS
FACULTAD DE TECNOLÓGICA
PROYECTO CURRICULAR DE TECNOLOGÍA EN SISTEMAS
ELÉCTRICOS DE MEDIA Y BAJA TENSION POR CICLOS
PROPEDÉUTICOS.
BOGOTÁ, D.C.
ABRIL DE 2019

*Diseño de una herramienta computacional para estimar
el COP de una persona a través de una plataforma Wii
Balance Board.*

JENNY KATHERINE TORRES ZAMBRANO
JULIAN OSWALDO PEREZ DE LA CUADARA

ESTUDIANTES DE INGENIERÍA ELÉCTRICA POR CICLOS PROPEDÉUTICOS.

PROPUESTA PRESENTADA PARA OPTAR AL TÍTULO DE INGENIERO
ELÉCTRICO.

INGENIERÍA ELÉCTRICA POR CICLOS PROPEDÉUTICOS.

DIRECTOR

LELY ADRIANA LUENGAS CONTRERAS, PH.D.
DOCTORA EN INGENIERÍA

CO-DIRECTOR

HENRY ALBERTO HERNÁNDEZ MARTINEZ, MSc.
MAGÍSTER EN INGENIERÍA -INGENIERÍA DE TELECOMUNICACIONES-.

LÍNEA DE INVESTIGACIÓN

INTELIGENCIA ARTIFICIAL

GRUPO DE INVESTIGACIÓN

DIGITI



UNIVERSIDAD DISTRITAL FRANCISCO JOSÉ DE CALDAS
FACULTAD DE TECNOLÓGICA

PROYECTO CURRICULAR DE TECNOLOGÍA EN SISTEMAS
ELÉCTRICOS DE MEDIA Y BAJA TENSION POR CICLOS
PROPEDÉUTICOS.

BOGOTÁ, D.C.
ABRIL DE 2019

Título en español

Diseño de una herramienta computacional para estimar el COP de una persona a través de una plataforma Wii Balance Board.

Title in English

Design of a computational tool to estimate a COP through a Wii Balance Board platform.

Resumen: Este trabajo de grado presenta una metodología con el fin de establecer una estrategia, para diseñar un mecanismo de recolección de datos relacionados al balance de una persona en posición estática y bi-pedestada. Estos datos se recolectan mediante una interfaz gráfica de usuario, la cual se conecta con una plataforma Wii Balance Board mediante una interfaz bluetooth que implementa el protocolo L2CAP (Protocolo de control y adaptación del enlace lógico). La plataforma tiene cuatro (4) galgas extensiométricas que le permiten al usuario determinar el Centro de Presión Plantar (COP) a partir de la fuerza que se ejerce sobre ellas. El COP en este caso se estima usando una aproximación matemática basada en la ley de gravitación universal, que emula la atracción de una partícula hacia el campo con mayor fuerza gravitatoria.

Abstract: This work of degree presents a methodology in order to establish a strategy, to design a mechanism of data collection related to the balance of a person in static and bi-pedestada position. These data are taken by means of a graphical user interface, which is connected to a Wii Balance Board platform through a bluetooth interface that implements the L2CAP protocol (Control protocol and adaptation of the logical link). The platform has four (4) extensimetric gauges that determine the Plantar Pressure Center (COP) based on the force exerted on them. The COP in this case is estimated using a mathematical approach based on the law of universal gravitation, which emulates the attraction of a particle to the field with greater gravitational force.

Palabras clave: Fuerza gravitatoria, Interfaz, Bio-Mecánica.

Keywords: Gravitational force, Interface, Bio-Mechanics.

Agradecimientos

Por un lado, destacamos la colaboración que nos brindaron nuestros profesores directora y co-director de tesis, ya que, gracias a su apoyo este proyecto llego a buen termino. Por otro lado, a nuestros familiares y amigos quienes nos impulsan a continuar para conseguir nuestras metas profesionales.

Contribuciones

Durante el desarrollo de esta tesis se realizaron varias contribuciones, las cuales se describen a continuación.

- La primer contribución es la aplicación que permite determinar el Centro de Presión, ya que, la aplicación fue desarrollada en una herramienta de software libre (Ver código en el anexo I). Esta característica le permite a cualquier persona utilizarla, ya sea para continuar con esta investigación o para establecer si su balance y peso son adecuados.
- La segunda contribución es la forma en la cual se adaptó la ley de atracción gravitacional para la realización del modelo computacional, ya que, hasta la fecha los modelos que se encuentran en el estado del arte no permiten un análisis de las características del usuario en tiempo real por tiempo indefinido. A diferencia de ellos, en este trabajo se modela el Centro de Presión como una partícula que interactúa con el entorno, la cual se vea afectada por la fuerza gravitatoria de cuatro elementos (ver numerales 2 y 3). Este efecto se modela en tiempo real, ya que, la interfaz gráfica permite determinar el estado de los sensores de la Wii Balance Board en tiempo real.

Índice general

Índice general	I
Índice de tablas	III
Índice de figuras	IV
Introducción	VI
Objetivo general	VIII
1. Marco teórico	1
1.1. Introducción	1
1.2. Descripción general de los análisis de postura	1
1.2.1. Posturografía	2
1.2.2. Equipos usados en posturografía	3
1.2.3. Técnicas para realizar cinéticos	4
1.2.4. Calibración del equipo de medición	5
1.2.5. Presentación de resultados	5
1.2.6. Plataforma de fuerza	6
1.2.7. Funcionamiento interno de la Nintendo Wii Banlance Board	7
1.2.8. Ley de gravitación universal	8
1.3. Estado del arte	9
1.4. Resumen del Capítulo	13
2. Descripción de la aplicación desarrollada	14
2.1. Introducción	14
2.2. Pre-requisitos de la aplicación	14
2.3. Funcionamiento de la aplicación	15

2.4. Protocolo de comunicación	18
2.5. Resumen del capítulo	19
3. Experimentación y resultados	22
3.1. Introducción	22
3.2. Experimento	22
3.3. Resultados obtenidos	24
3.4. Análisis de los resultados	24
3.5. Resumen del Capítulo	29
4. Conclusiones	30
5. Trabajo Futuro	31
A. Instrucciones de uso	32
B. Código de la aplicación	37
Bibliografía	45

Índice de tablas

3.1. Error porcentual estimado por sujeto (Fuente los autores).	24
---	----

Índice de figuras

1.1. Laboratorio de posturografía basado en técnicas de procesamiento de imágenes (Tomado de [11]).	3
1.2. Equipo para medir la presión plantar (Tomado de [2]).	4
1.3. Protocolo de calibración del equipo de videogrametría (Tomado de [26]). . .	5
1.4. Masas de referencia para ajustar equipos de pesaje (Tomado de [15]).	6
1.5. Mapa conceptual resumen está sección (Basado en el trabajo de [1]).	6
1.6. Ejemplo de plataforma dinamométrica comercial (Tomado de [44]).	7
1.7. Elementos eléctricos y electrónicos usados en la medición de fuerza (Tomado de [10]).	8
1.8. Partes de la nintendo Wii Banlance Board (Tomado de [16]).	9
1.9. Ejemplo de la fuerza de atracción que ejerce la tierra sobre la luna (Tomado de [6]).	10
2.1. Terminal de Ubuntu (Fuente los autores).	15
2.2. PYTHON IDLE (Fuente los autores).	16
2.3. Funcionamiento resumido de la aplicación (Fuente los autores).	18
2.4. Componentes de B (Fuente los autores).	20
3.1. Bascula usada en los experimentos (Fuente los autores).	22
3.2. Gráficas generadas con la aplicación (Fuente los autores).	23
3.3. Segmento de los datos almacenados por la aplicación (Fuente los autores). .	23
3.4. Comportamiento del peso observado usando la aplicación (Fuente los autores). 24	
3.5. Comportamiento del centro de presión (Fuente los autores).	25
3.6. Posición de las cargas durante la realización de la primer prueba y los resultados obtenidos en ambas aplicaciones.	25
3.7. Posición de las cargas durante la realización de la segunda prueba y los resultados obtenidos en ambas aplicaciones.	26

3.8. Posición de las cargas durante la realización de la tercer prueba y los resultados obtenidos en ambas aplicaciones.	26
3.9. Posición de las cargas durante la realización de la primer prueba y los resultados obtenidos en ambas aplicaciones.	27
3.10. Posición de las cargas durante la realización de la segunda prueba y los resultados obtenidos en ambas aplicaciones.	28
3.11. Posición de las cargas durante la realización de la tercer prueba y los resultados obtenidos en ambas aplicaciones.	28
A.1. Posición de trabajo de la Wii Balance Board.	32
A.2. Terminal del sistema operativo Ubuntu.	32
A.3. Presentación de la aplicación.	33
A.4. Forma de cambiar las baterías (Esta imagen fue tomada de [57]).	34
A.5. Sincronización.	34
A.6. Éxito.	34
A.7. Fracaso.	34
A.8. Ajuste automático de la plataforma Wii Balance Board.	35
A.9. Uso de la plataforma Wii Balance Board.	36

Introducción

La posturografía es una herramienta que le permite a los expertos determinar diferentes patologías, entre las que se encuentran el mal de Alzheimer en sus diferentes estados, alternaciones del nervio periférico, atrofia muscular y espinal. Estas patologías normalmente se detectan en estados tempranos gracias a los análisis especializados que representan gráficamente patrones de comportamiento muscular o factores motrices [20, 54, 62].

El seguimiento de los movimientos corporales en la posturografía se ha convertido en una herramienta versátil que facilita el conocimiento de las capacidades motoras de una persona, ya que, la capacidad motora se puede determinar a partir de patrones cinemáticos o estáticos de una persona. El estudio más común es el análisis cinemático que recoge grandes cantidades de información, cuando una persona se mueve de un punto a otro. Normalmente, esta información se recolecta con equipos de vídeo o acelerómetros ubicados en diferentes secciones del cuerpo. A diferencia de ese tipo de análisis, la posturografía se realiza para conocer características específicas del sujeto, como la distribución de la presión en las zonas plantares o si existe alguna desviación de los huesos en las extremidades inferiores. En ambos casos, se guardan registros del comportamiento de las extremidades inferiores y superiores, la respiración, los latidos del corazón, entre otros datos, con el fin de asociar un patrón de comportamiento a algunas de las patologías antes mencionadas [43, 55].

El desarrollo de técnicas para realizar un análisis de los patrones de comportamiento de manera cuantitativa y cualitativa de rasgos característicos de una persona ha sido de gran interés para los investigadores y médicos, ya que, a partir de pocos parámetros determinan alteraciones biomecánicas o antropométricas ligadas al desequilibrio muscular y disposición corporal. Por una lado, algunas de estas técnicas se basan en el procesamiento de secuencias de imágenes y el seguimiento de los cambios de las zonas o regiones de interés para generar un informe al comparar diferentes registros almacenados en una base de datos con los proporcionados por la secuencia de imágenes. Por otro lado, al automatizar estos procesos de predicción de alteraciones biomecánicas mediante el procesamiento de imágenes digitales se pretende reducir la intervención humana durante el análisis de la información. Sin embargo, lograr una independencia total entre el conocimiento del médico especialista, el registro de la información y el análisis de los resultados sigue siendo un reto, debido a que implementar una rutina que sistematice totalmente el conocimiento de un especialista no ha sido posible [23, 32].

Otro análisis de laboratorio es el estudio del centro de presión plantar que permite determinar la distribución de cargas en las diferentes zonas de los pies. Esto es de gran importancia para determinar diferentes tipos de patologías, tales como úlceras asociadas al

pie diabético o deformidades en las extremidades inferiores que a largo plazo repercuten en varias secciones del cuerpo. Sin embargo, los análisis de presión plantar se realizan únicamente por laboratorios especializados y certificados que cuenten con médicos especialistas en la biomecánica del cuerpo humano, esta característica hace que los exámenes y pruebas sean costosos reduciendo la posibilidad de acceso a personas de recursos económicos limitados [34, 35, 39, 42].

Este trabajo de grado propone una solución a esta limitación de acceso con un prototipo electrónico, el cual le permita a una persona del común y sin conocimientos en ingeniería conocer su Centro de Presión (COP). Este prototipo facilitara al técnico especialista analizar los resultados obtenidos por la persona y así determinar si puede sufrir alguna clase de patología sin necesidad de equipo especializado. Además, el prototipo contara con una interfaz gráfica de usuario que genera el reporte automáticamente y de manera intuitiva. La contribución de este trabajo es el inicio para implementar equipo clínico que le permita al especialista realizar un diagnostico sin muchos recursos, ya que, el prototipo es lo suficientemente flexible como para que opere sin necesidad de un conocimiento técnico en ingeniería específico. Esta contribución se describe de una manera detallada en este documento, el cual se compone de tres (3) Capítulos organizados de la siguiente manera.

- **Capítulo 1:** Se presentan los conceptos generales acerca del análisis de la posición estática y bi-pedestada. Además, de un breve recuento de la importancia de la ingeniería en este campo.
- **Capítulo 2:** Se realiza una descripción del prototipo desarrollado y de la metodología empleada para el desarrollo y manufactura del mismo.
- **Capítulo 3:** Se hace una descripción de las diferentes pruebas que se realizaron al prototipo para obtener una confirmación metro-lógica de sus valores. Lo cual se reporta mediante gráficas y tablas de resultados.

Objetivo general

Diseñar una aplicación que permita monitorear el comportamiento de una plataforma WII FIT, a través de la cual se determine el centro de presión del usuario.

Objetivos específicos

1. Proponer un modelo experimental que permita determinar el centro de presión de una persona en posición bi-pedestada estática a partir de los datos medidos con la plataforma.
2. Construir una interfaz gráfica de usuario mediante la cual se pueda graficar el centro de presión de una persona en posición bi-pedestada y generar un reporte del comportamiento del mismo en diferentes intervalos de tiempo.
3. Comparar y analizar los datos medidos y los generados por el modelo experimental con al menos tres (3) personas diferentes.

CAPÍTULO 1

Marco teórico

1.1. Introducción

El objetivo de este trabajo es re-acondicionar una plataforma dinamométrica para encontrar el Centro de Presión (COP) de una persona. Este análisis se realiza a partir de la re-construcción de una señal en tiempo real, la cual se compone de la fuerza detectada con cuatro galgas extensiométricas. Debido a la gran cantidad de conceptos en este Capítulo se presentan todos los necesarios para entender el como funciona la aplicación propuesta. Este Capítulo está organizado de la siguiente forma: en la sección 1.2 se presenta una descripción de la posturografía, en la sección 1.2.6 se muestra una descripción del funcionamiento de una plataforma dinamométrica y la Nintendo Balance Board y en la sección 1.3 hay un resumen de los trabajos más parecidos al presentado en este documento.

1.2. Descripción general de los análisis de postura

En algunos casos, el diagnóstico de enfermedades musculoesqueléticas (generales o transitorias) se determinan mediante análisis de la postura corporal, ya que, estas enfermedades afectan a una gran cantidad de personas [25, 61]. Por lo general, los padecimientos asociados a la marcha se caracterizan por la presencia de dolor o debilitamiento general, lo cual lleva a adoptar posturas antiálgicas ocasionando patologías respiratorias, obesidad o problemas psicológicos [33, 46].

La posturografía es un análisis que se realiza para identificar en un paciente posibles patologías musculares u óseas. Esto le permite a un especialista realizar posibles prescripciones quirúrgicas, farmacológicas, terapéuticas u otras técnicas que minimicen el daño que sufre el paciente [41]. Debido a esto, la posturografía se ha abordado fuertemente desde dos frentes. Por un lado, este tipo de análisis se ha usado en otras áreas como la biomecánica deportiva, diseño de productos para el trabajo o productos que mejoren la ergonomía de la persona mientras corre, camina o trotta [4, 49]. Por otro lado, en investigaciones que tratan de entender la biomecánica del cuerpo humano se ha extendido para entender la biomecánica de las extremidades de los animales [12].

Los protocolos estandarizados usados en posturografía para realizar exámenes clínicos se han formulado de diferentes maneras debido a la variabilidad que existe entre los conjuntos de personas, lo cual le permite a este tipo de análisis actuar de una forma confiable y eficiente. Se debe tener en cuenta, que los protocolos están sujetos a las características propias del paciente, las variables a analizar, la disposición del laboratorio, la posición de los marcadores (en caso de que se aplique) y el software utilizado [18]. Estas características particulares de los análisis cinemáticos, también se aplican a los análisis estáticos. Como los análisis de interés en este trabajo son los de tipo estático, en el siguiente numeral se enunciarán varias características del análisis cinemático y su relación con el análisis estático.

1.2.1. Posturografía

La posturografía es útil para valorar la estabilidad del paciente, así como detectar posibles trastornos del equilibrio o la enfermedad vertiginosa; además de establecer su posible rehabilitación. Así mismo, también es útil para aquellas personas con dificultad de caminar, de balanceo, transferencia de peso o para la rehabilitación de tobillo, rodilla, cadera o espalda [66].

La posturografía consiste en colocar al paciente sobre una plataforma en distintas condiciones: de pie, con los ojos abiertos, ojos cerrados, con la plataforma estable o móvil. Las pruebas para valorar el equilibrio duran entre 20 y 30 minutos. En caso de utilizarse durante el tratamiento del vértigo, las sesiones durarán entre 30 y 45 minutos. Se recomienda como mínimo realizar cuatro sesiones, dos semanales, de rehabilitación vestibular para valorar la evolución del paciente. Para realizar esta prueba no se necesita ninguna preparación previa, sólo debe estar descalzo sobre la plataforma del posturógrafo lo más tranquilo posible [3, 47].

Un caso general, es la posturografía dinámica computarizada (en inglés *Computerized Dynamic Posturography*, CDP) es una técnica para la valoración objetiva de la estabilidad postural en bipedestación, que se puede definir como el control del centro de masas con respecto a la base de soporte, tanto en situaciones estáticas como dinámicas. El control postural es una función básica para el organismo, que le permite desarrollar sus actividades de la vida cotidiana y que se consigue a través de la integración en el sistema nervioso central de la información somato-sensorial, visual y vestibular. La importancia relativa de cada una de esas referencias para el mantenimiento del equilibrio es lo que se determina con la CDP [58].

La prueba de organización somato-sensorial (SOT) nos permite analizar la aportación individual y combinada que cada sistema tiene en el mantenimiento del equilibrio. Como se ha comentado previamente, el posturógrafo consta de una plataforma y un entorno móviles. Ambos se pueden mover, controlados por el ordenador, alrededor de un eje alineado con la articulación del tobillo. De este modo se pueden asociar las oscilaciones del cuerpo del paciente, con los movimientos de la plataforma y el entorno, de tal forma que estos se desplazarán en el mismo grado y dirección. Con ello se puede anular en gran medida, la información somato-sensorial aportada por los receptores localizados en el tobillo (informan de las variaciones en el ángulo tibio tarsal) y se aporta una información visual errónea que indica que no hay movimiento, cuando realmente sí lo hay. Gracias a estas modificaciones en las referencias de los sistemas encargados del mantenimiento del equilibrio podemos evaluar la contribución relativa de los sistemas visual y somato-sensorial, y la habilidad

para adaptarse a situaciones sensoriales conflictivas en condiciones de prueba referenciada a las oscilaciones corporales (sway referenced) [45].

El estudio se ejecuta colocando al paciente sobre la plataforma y frente al entorno, se le pide que mantenga el equilibrio en seis condiciones diferentes en las que se realiza una distorsión selectiva de la información sensorial, haciendo tres intentos, de veinte segundos de duración, en cada una de ellas [51].

1.2.2. Equipos usados en posturografía

Los parámetros de interés se pueden medir con equipos especializados que permiten cuantificar cada variable durante un análisis de posturografía. Existen varios tipos de laboratorio, el más básico permite analizar los parámetros del usuario usando cintas métricas, un cronómetro y una observación continua por un técnico [5, 7]. Una forma de automatizar este proceso de observación es mediante equipos que miden parámetros cinemáticos y cinéticos con diferentes sensores tales como; dinamómetros, acelerómetros, ultrasónicos, goniómetros. Al implementar este tipo de equipos se puede cuantificar la información de una manera confiable comparada con los procesos de observación visual. Sin embargo, el costo de estos equipos y el personal que lo opera hace que el valor de un examen sea elevado, debido a esto se ha intentado desarrollar técnicas de bajo costo como las basadas en el procesamiento de imágenes (Fig. 1.1) o en videogrametría [63]. Aunque, la sistematización de estas técnicas se encuentra en desarrollo aun se pueden proponer algunos prototipos que intenten solucionar esta problemática, como la aplicación que se presenta en este documento.



FIGURA 1.1. Laboratorio de posturografía basado en técnicas de procesamiento de imágenes (Tomado de [11]).

El gran impacto que han tenido las técnicas basadas en procesamiento de imágenes ha sido estimulado la reducción de costos en este tipo de análisis, pero no permite la medición de características cinéticas. Normalmente, este tipo de análisis se realiza con plantillas o plataformas especializadas (Fig. 1.2), lo cual disminuye su accesibilidad. Teniendo en cuenta esta limitación en este trabajo se propone un sistema de bajo costo para realizar análisis cinéticos [21]. Sin embargo, los análisis cinéticos se pueden ver afectados por los análisis cinemáticos, debido a esto en el siguiente numeral se hace una descripción detallada de los análisis basados en procesamiento de imágenes, ya que, son de bajo costo y muy versátiles como el impacto que se busca lograr con esta aplicación.

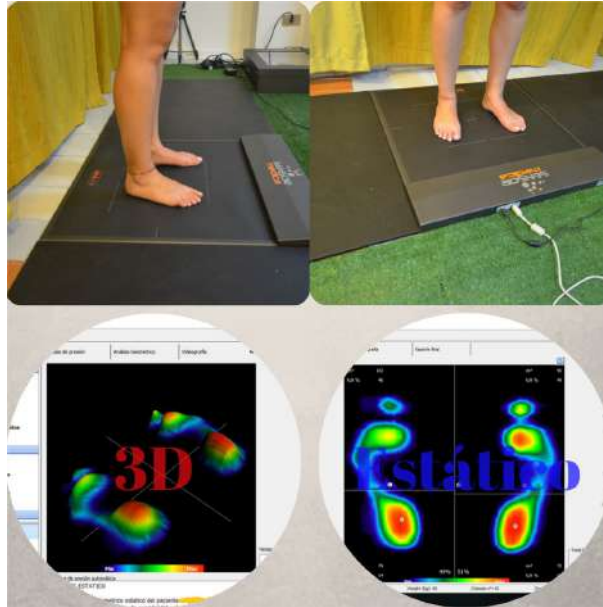


FIGURA 1.2. Equipo para medir la presión plantar (Tomado de [2]).

1.2.3. Técnicas para realizar cinéticos

Las técnicas de procesamiento de imágenes digitales se basan en mecanismos de segmentación que permiten la identificación de marcadores en secuencias de imágenes o vídeos [31]. Estos marcadores normalmente se ponen siguiendo un protocolo que se utilizan para detectar puntos anatómicos de interés de acuerdo con varios protocolos, entre los que se encuentran; el kadaba, el Newington, el Helen y el VCM (del inglés, Vicon Clinical Manager) [61, 46, 18]. En general, estos protocolos usan puntos de referencias oseas como la cabeza del peroné, la cabeza del quinto metatarsianos, la séptima vertebra cervical, la espina iliaca, el cóndilo femoral y la vertebra sacra. Además, los marcadores se implementan sobre el muslo derecho e izquierdo y la mitad de la pierna izquierda y derecha mediante marcadores como cintas reflectivas o LEDs (Fig. 1.1) [11].

Estos marcadores son interpretados por el software que guarda sus posiciones como secuencias de puntos y los reconstruye de manera gráfica para realizar un análisis. Sin embargo, se debe tener en cuenta que lo ideal es que los marcadores fueran adheridos a los huesos. Debido a esta característica se usan marcadores adheridos a la piel que tienen un diámetro de 40 ± 30 mm, para indicar un desplazamiento independiente de cada articulación. Al realizar esta operación se induce un error de medición traslacional de ± 10 mm y de ± 13 grados en medidas rotacionales [12, 47]. A diferencia de estos análisis, el

análisis cinético no cuenta con marcadores específicos, ya que, la persona normalmente se encuentra en una posición bipedestada y estática. En estos análisis se utilizan plataformas de fuerza, sensores de presión o plantillas susceptibles a la deformación, para registrar los datos y contribuyen a detectar posibles desviaciones de la columna u atrofas de tipo muscular [2]. En ambos casos (estudio cinemático y cinético), los equipos deben calibrarse y ajustarse siguiendo un protocolo como se describe a continuación.

1.2.4. Calibración del equipo de medición

Un aspecto importante que se debe tener en cuenta durante el proceso de adquisición de datos es la calibración y ajuste de los equipos de posturografía. Los parámetros de ajuste y calibración de los equipos en casi todas las ocasiones son dados por el fabricante, ya que, este tipo de equipos tienen sensores con una alta sensibilidad y cuando no se ajustan de manera adecuada el proceso de recolección de datos es deficiente.

Como se mencionó antes, en la posturografía se utilizan dos tipos de equipos; los equipos de videogrametría y las plataformas de fuerza. Por un lado, para calibrar los equipos de videogrametría se realizan tres (3) pasos (Fig. 1.3); linealizar las cámaras, ajustar el campo visual y alinear los marcadores con la fuente de luz para que se realice una lectura adecuada [26]. Por otro lado, las plataformas de fuerza se ajustan con base a masas patrón calibradas con referentes de tipo internacional y así asegurar una medición adecuada (Fig. 1.4). Esto se debe a que las plataformas de fuerza normalmente se componen de sensores de presión o galgas extensiométricas para realizar una medición [14], lo cual se describirá detalladamente en el siguiente numeral.

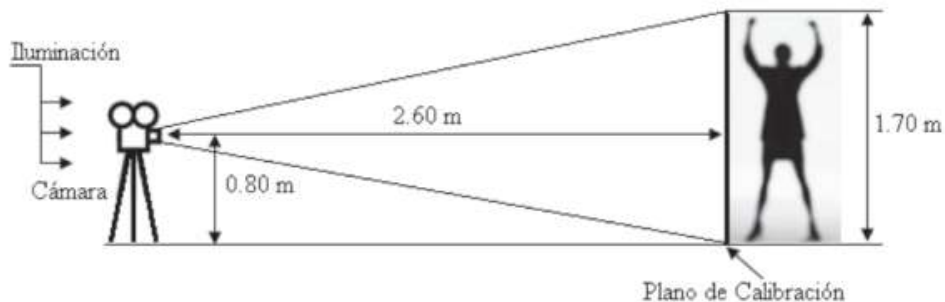


FIGURA 1.3. Protocolo de calibración del equipo de videogrametría (Tomado de [26]).

1.2.5. Presentación de resultados

Los informes que proporciona una equipo de posturografía normalmente se almacena en forma digital o impresa, los cuales se adaptan para que muestren únicamente la información de interés para el especialista. En ocasiones, la información del reporte se segmenta dependiendo del ciclo, para reducir su subjetividad al disminuir la variabilidad y el rango en el cual se pueden analizar las gráficas resultantes [3]. Además, están un gran numero de técnicas espaciales que permiten descomponer este tipo de análisis en problemas de múltiples variables. Sin embargo, la elección de la técnica nos es fácil, debido a la variabilidad que se presenta en el comportamiento que tiene cada paciente (Fig. 1.5). Por lo tanto, idear una herramienta que prediga el comportamiento que puede tener un paciente es aun un reto, lo cual le ha permitido a los investigadores proponer algunas soluciones parciales



FIGURA 1.4. Masas de referencia para ajustar equipos de pesaje (Tomado de [15]).

como presentar la información del comportamiento de un paciente mediante gráficas de tendencia, mapas de colores, entre otros métodos convencionalmente usados en estadística descriptiva [17]. Finalmente en la Fig. 1.5 se presenta un resumen de conceptos generales de las plataformas de fuerza y en las siguientes secciones se describirán sus componentes, funcionamiento y el principio físico usado para modelar el centro de presión.

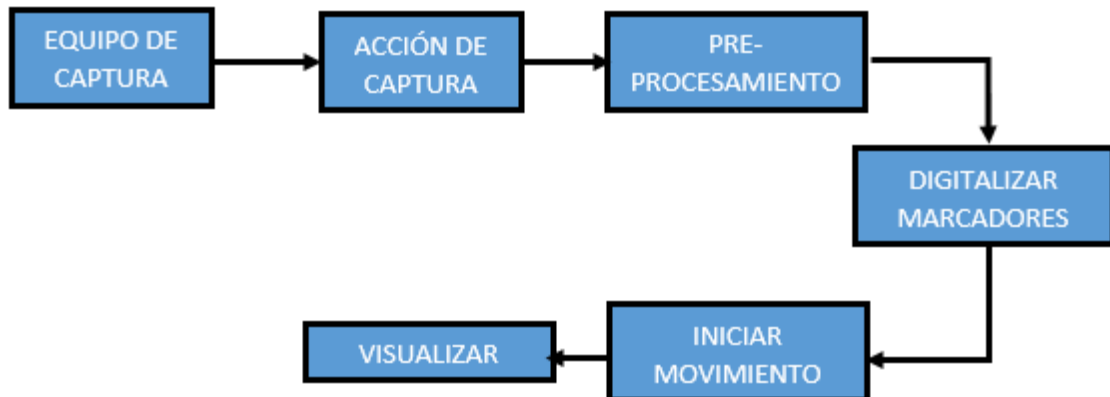


FIGURA 1.5. Mapa conceptual resumen esta sección (Basado en el trabajo de [1]).

1.2.6. Plataforma de fuerza

Las plataformas de fuerza o dinamométricas son sistemas que se utilizan en análisis cinéticos para medir el conjunto de fuerzas que se ejerce sobre los pies sobre la superficie de apoyo durante el desarrollo de diferentes actividades cotidianas (estar de pie). Estas plataformas se usan en grupos para detectar las fuerzas aplicadas mientras un usuario esta posición bipedestada y estatica (Fig. 1.6(b)) y su funcionamiento se basa en la tercera ley de Newton (principio de acción y reacción) que estima la fuerza sobre una superficie a partir de la fuerza que la origina, cuyas magnitudes tienen el mismo valor pero sentido contrario [60].

Las fuerzas aplicadas sobre la plataforma normalmente se transforman en señales eléctricas que proyectan el valor de la magnitud descompuesta en tres ejes de coordenadas cartesianas X, Y y Z (Fig. 1.6(a)). Estas fuerzas se detectan usando sensores piezo-eléctricos o galgas extensiométricas, las cuales se conectan a un amplificador de instrumentación o circuito de acondicionamiento que se encarga de llevar la señal de salida del sensor ($< 1V$) a un rango de trabajo lineal (comúnmente 5V) [56].

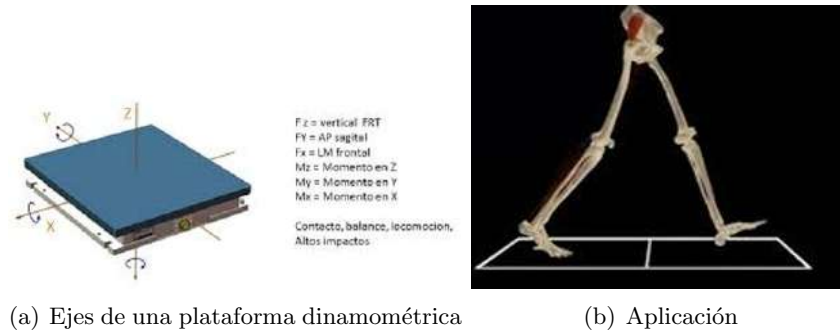


FIGURA 1.6. Ejemplo de plataforma dinamométrica comercial (Tomado de [44]).

Por un lado, los sensores piezo-eléctricos (Fig. 1.7(a)) son elementos que se usan para medir presión o aceleración a partir del efecto piezo eléctrico. Este efecto consiste en que a medida que se deforma un cristal este produce una tensión eléctrica que se asocia a la tensión superficial que tiene la superficie del cristal. Por otro lado, las galgas extensiométricas (Fig. 1.7(b)) son resistencias variables que cambian su valor a medida que la presión sobre ellas aumenta. En ambos casos, la señal de salida o señal de medida debe acondicionarse utilizando un circuito eléctrico que normalmente se compone de un arreglo de resistencias, condensadores, bobinas y amplificadores operacionales. En el caso del sensor piezoeléctrico normalmente se utiliza un amplificador de instrumentación para elevar la tensión de salida y llevarla a una región de operación lineal, en el caso de la galga extensiométrica se construye un puente de Weathstone para realizar una conversión de su valor de resistencia a tensión eléctrica en un rango de trabajo lineal. Dicho puente es un arreglo de cuatro resistencias que en condiciones de equilibrio su tensión de salida es cero y cuando se des balancea al cambiar el valor de una de ellas se presenta una tensión eléctrica en sus terminales (Fig. 1.8(a)). De manera similar que en el caso del sensor piezoeléctrico esta salida se lleva a un amplificador de instrumentación para elevar el voltaje [8, 9].

1.2.7. Funcionamiento interno de la Nintendo Wii Balance Board

Como se mencionó en el numeral anterior, la plataforma dinamométrica se utiliza para medir la fuerza de impacto durante un análisis cinético. De manera similar, la nintendo Wii Balance Board (Fig. 1.8(a)) se utiliza este efecto para medir las fuerzas que realiza una persona sobre su superficie. La magnitud de estas fuerzas se determina con cuatro (4) galgas extensiométricas (Fig. 1.8(b)) ubicadas en cada esquina de la plataforma las cuales detectan la inclinación del sistema mediante un sistema de barras acoplado entre si, además, tiene un sistema de alimentación de cuatro (4) baterías que se encuentra en la parte inferior de la plataforma. Cada galga tiene un circuito de acondicionamiento que eleva la tensión de salida a una que puede medir por el microcontrolador [57]. El microcontrolador (Fig. 1.8(c)) de la plataforma se encarga de conectarse con el computador

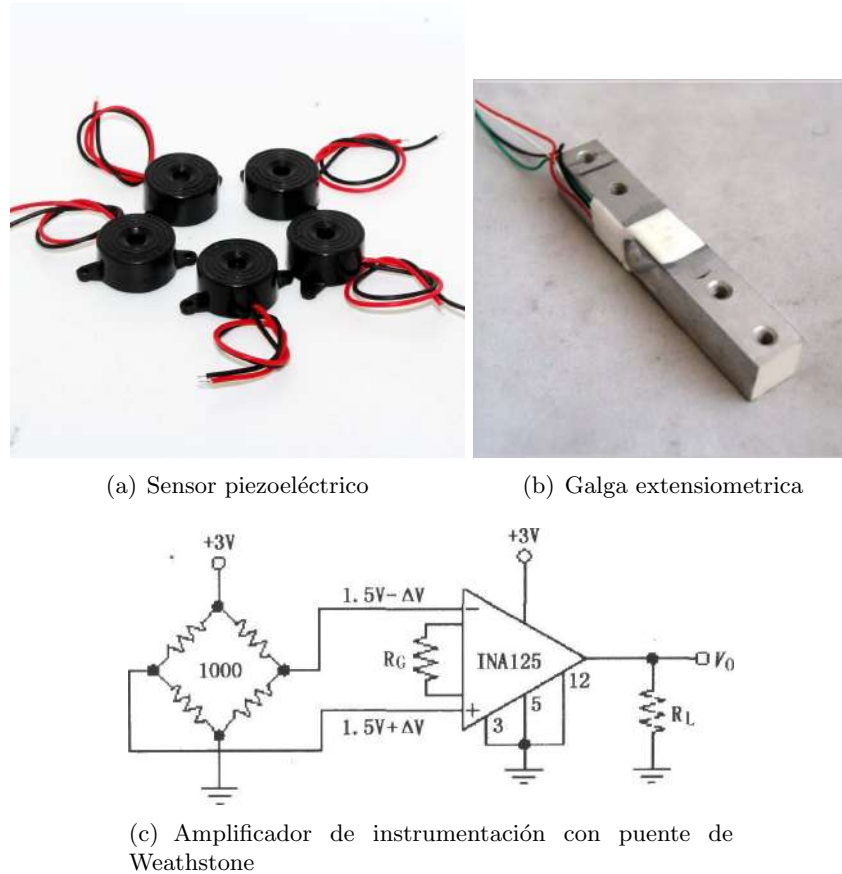


FIGURA 1.7. Elementos eléctricos y electrónicos usados en la medición de fuerza (Tomado de [10]).

mediante un bluetooth que utiliza el protocolo L2CAP (Protocolo de control y adaptación del enlace lógico), el cual se encarga de segmentar la información y enviarla mediante paquetes que se deben decodificar usando el protocolo de la Nintendo Wii. Además, este protocolo se encarga de fijar la Nintendo Wii balance board como dispositivo maestro, es decir, todo dispositivo que se conecte a la Nintendo Wii balance board será un dispositivo esclavo y recibirá la información que le envíe el dispositivo maestro [59]. Como cada galga extensiométrica se encuentra ubicada en puntos fijos de manera simétrica, se usó la ley de gravitación universal para determinar hacia qué costado se encuentra el centro de presión, por lo tanto, en el siguiente numeral se describe dicha ley.

1.2.8. Ley de gravitación universal

La ley de gravitación universal describe el comportamiento de dos cuerpos diferentes con masa, la cual establece una relación lineal entre dos objetos que se atraen y que tienen masa. Así esta relación que es el producto de las masas sobre la distancia entre ellas al cuadrado, indica que al aumentar la masa de uno de los dos cuerpos hay una fuerza de atracción mayor (Eq. 1.1) [36]. El símil que se hace con el proyecto desarrollado se menciona en el siguiente Capítulo, pero antes de terminar en el siguiente numeral se presenta una revisión del estado del arte acerca de las formas que se usan para medir el centro de presión.

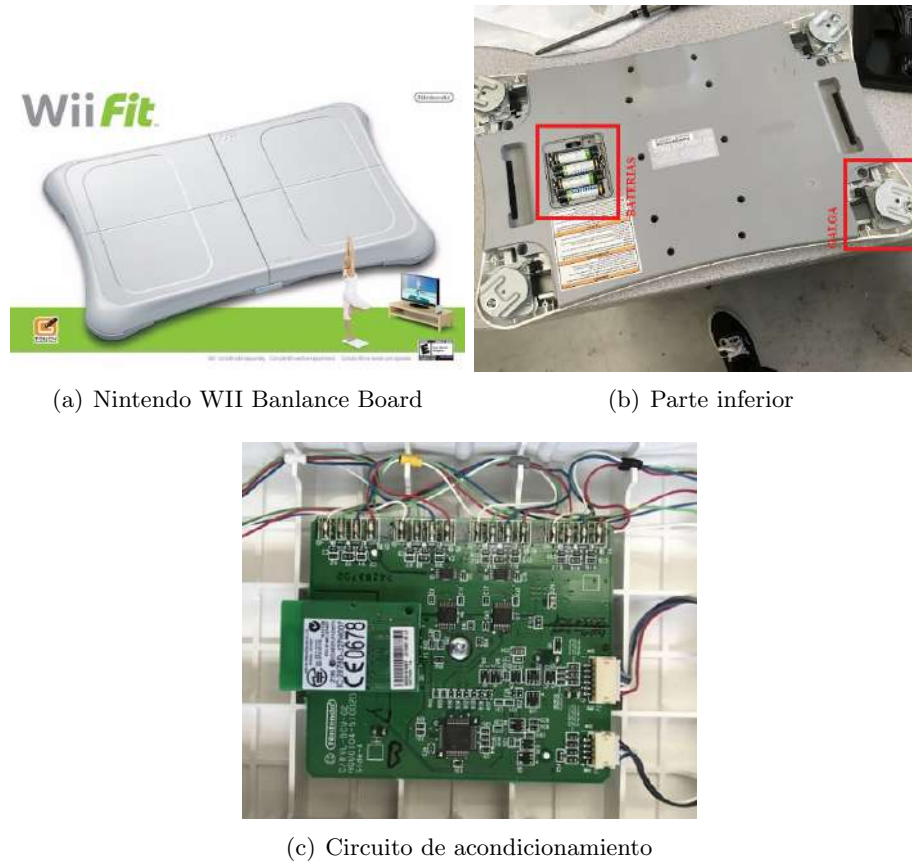


FIGURA 1.8. Partes de la nintendo Wii Banlance Board (Tomado de [16]).

$$F = G * \frac{m_1 * m_2}{r^2} \quad (1.1)$$

G=Constante de gravitación universal ($G = 6,67 \times 10^{-11} * Nm^2 * kg^{-2}$)

F=Fuerza de atracción

1.3. Estado del arte

En el año 2001 se evaluó el efecto del golpe en las características escalonadas, con el objetivo de comparar las medidas espaciales y temporales durante el cruce del obstáculo de la extremidad anterior entre sujetos con accidente cerebro vascular y sujetos sanos. Todo se desarrollo a la Unidad de rehabilitación geriátrica en un hospital y se midieron los datos espacio-temporales para 16 sujetos con accidente cerebro vascular y 16 sujetos sanos. Los sujetos con accidente cerebro vascular eran pacientes hospitalizados y tenían que poder caminar 10 metros con o sin ayuda. Además, estos sujetos debían pasar por encima de los obstáculos altos y anchos, de 1 a 8 cm, y los ensayos se grabaron en vídeo. Principalmente se midió la separación del dedo del pie, la distancia previa al obstáculo, la distancia posterior a la traba, la longitud del paso, la proporción de la longitud previa del escalón, el tiempo de progresión, el tiempo de la etapa previa al obstáculo, el tiempo de avance posterior a la traba y la proporción de la fase previa al paso. Esto permitió realizar

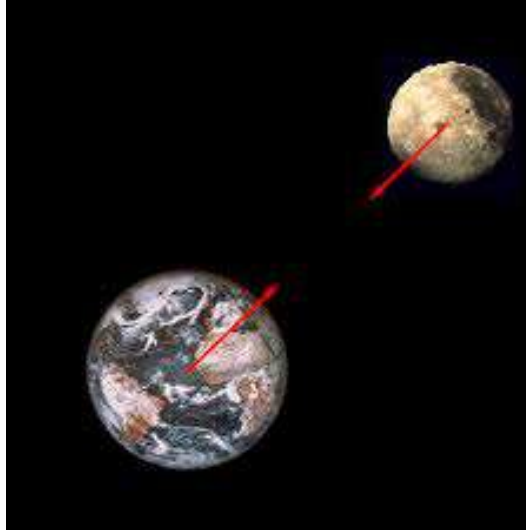


FIGURA 1.9. Ejemplo de la fuerza de atracción que ejerce la tierra sobre la luna (Tomado de [6]).

pruebas U de Mann-Whitney para determinar las diferencias entre los 2 grupos. Los sujetos con accidente cerebro vascular tuvieron un aclaramiento del dedo del pie significativamente mayor, distancias más pequeñas después de un obstáculo y mayores tiempos de pasos que los sujetos sanos. Los sujetos con accidente cerebro vascular no demostraron una reducción significativa en la distancia previa al obstáculo. Como se puede observar en este tipo de experimentos es útil contar con los equipos que se describen en este trabajo, ya que, permitieron concluir que al modificar la trayectoria de su extremidad principal durante el cruce de obstáculos, las personas con accidente cerebro vascular reducen el riesgo de un viaje debido al contacto con los dedos, pero la modificación puede exponerlos a otros riesgos de seguridad [53].

A partir de este trabajo también se analizó el efecto de la colocación del pie en la posición sentada para estar de pie en sujetos jóvenes sanos y en pacientes con hemiplejía. El objetivo de este trabajo fue determinar el efecto de alterar la colocación del pie de la extremidad dominante en sujetos jóvenes sanos y la extremidad no comprometida de los sujetos con hemiplejía en su capacidad para realizar sentarse para pararse (STS). Además, se diseñó un experimento biomecánico controlado en un Laboratorio de investigación de un centro universitario de ciencias de la salud. Este análisis se realizó con una muestra no aleatoria de 10 sujetos sanos y 10 con hemiplejía. Las edades medias respectivas fueron de 26 y 59 años. Todos los pacientes con hemiplejía deambulan y STS de forma independiente. El tiempo medio transcurrido desde fue de 3,6 años y los sujetos pasaron de estar sentados a una posición de pie bajo 3 condiciones diferentes: (1) condición normal, donde ambas extremidades se colocaron en 100 de flexión de la rodilla; (2) condición extendida de la extremidad, donde la extremidad dominante o no afectada se extendió a 75 de flexión de la rodilla; y (3) condición de elevación de la extremidad, en la que la extremidad dominante o no comprometida se colocó sobre un soporte de espuma densa igual al 25 % de la altura de la rodilla del sujeto. Se observó que la fuerza de reacción vertical y anteroposterior (GRF) y actividad del electromiograma tibial anterior bilateral y cuádriceps (EMG) y en los sujetos jóvenes, la extremidad no dominante colocada normalmente compensó la posición extendida o elevada de la extremidad dominante. Las amplitudes máximas de GRF y EMG fueron todas significativamente mayores para la extremidad no dominante. En pacientes con hemiplejía, la EMG de la extremidad afectada aumentó 39 % en las

condiciones de elevación y extensión de la extremidad en comparación con la condición normal. Los valores respectivos para el miembro no afectado disminuyeron. Los GRF fueron significativamente mayores para el miembro no afectado, excepto por la fuerza vertical en la posición extendida del miembro [24].

Otros trabajos de análisis biomecánico muestra el efecto del movimientos de los brazos, al realizar movimientos de salto vertical utilizando una plataforma dinamométrica. El conversor analógico a digital de la placa de fuerza y el amplificador de rango se pusieron a cero electrónicamente y se calibraron antes de cada sujeto. En este documento se pueden realizar dos saltos verticales con y sin un movimiento del brazo. Los datos y las señales informados se obtuvieron de un atleta que está familiarizado con la realización de un salto correcto. Se presentan las mejores grabaciones de salto por la plataforma de fuerza. En consecuencia, se concluye que los movimientos de los brazos contribuyen a los movimientos de salto de alto rendimiento [22].

Otro artículo relacionado al anterior presenta una plataforma novedosa para mediciones de fuerza isométrica/torsión monolateral en el pie de sujetos humanos. Se utilizó un enfoque mecatrónico centrado en el ser humano en el diseño y desarrollo de esta plataforma, que consiste en una estructura ergonómica móvil que incorpora dos sensores de fuerza/par de seis ejes. La altura de la plataforma desde el suelo se puede ajustar fácilmente a las propiedades antropométricas del cuerpo humano, sin ninguna reducción de la rigidez del sistema en general, como se requiere estrictamente para mediciones precisas de la fuerza isométrica/par. Gracias a un estudio ergonómico específico, se puede seleccionar un número discreto de niveles de altura para cumplir con los requisitos antropométricos de una gran mayoría de la población. La plataforma propuesta ha sido diseñada y construida para su aplicación en la evaluación funcional de pacientes después de un accidente cerebro vascular, y actualmente se está validando en tres centros clínicos diferentes en Europa. La aplicación futura para el ejercicio motor isométrico en la neuro-rehabilitación, y también como una interfaz genérica, basada en el pie hombre-máquina, se encuentran actualmente bajo evaluación de factibilidad [28].

Una modificación del anterior tiene el objetivo de demostrar que un sensor de fuerza inductiva multiplexado mide simultáneamente la fuerza normal y las fuerzas de corte en un pie. El sensor mide la fuerza normal y las fuerzas de corte al controlar los cambios de inductancia de tres bobinas de detección planas. La multiplexación por división de frecuencia de resonancia se aplicó a las señales de las bobinas de detección múltiples, lo que hizo posible medir simultáneamente las tres fuerzas (fuerza normal, fuerza de corte en los ejes x e y) en un pie utilizando solo un conjunto de electrónica de medición con alta sensibilidad y resolución. Resultados: Los resultados de las pruebas del sensor prototipo han demostrado que el sensor es capaz de medir una fuerza normal que varía de 0 a 800 N y que las fuerzas de corte varían de 0 a 130 N en tiempo real. Con su alta resolución, alta sensibilidad y la capacidad de monitorear las fuerzas en diferentes posiciones de un pie simultáneamente, este sensor se puede usar potencialmente para medir en tiempo real la fuerza plantar normal y la distribución de las fuerzas de corte en el pie del paciente diabético [27].

Otra variante de los trabajos anteriores se basa en la captura de la imagen de la presión plantar, ya que, es generalmente aceptada como una herramienta efectiva para analizar una patología del pie y un trastorno en la investigación del cuidado del pie durante mucho tiempo. En esta investigación, se propuso un estudio preliminar sobre el diseño y la construcción de un analizador de presión plantar personalizado y de bajo costo para la

imagen de la presión del pie en una clínica general de rehabilitación de pies. Una única plataforma de carga de pies descalzos está diseñada para medir una dispersión de la presión plantar en la que se combinan 72 sensores sensibles a la fuerza en forma de matriz fija de 6×12 de matriz. Un software GUI apropiado está especialmente diseñado para ajustarse a los requisitos reales de un médico clínico en el análisis de distribución de presión 2D y 3D. Finalmente, el sistema de base de datos para almacenamiento y recuperación en el proceso de seguimiento también se desarrolla bien. No solo es un resultado interesante obtenido de este estudio, esta investigación también nos guía para desarrollar un sistema de escáner para el cuidado de los pies más personalizado y apropiado para usar en el hospital general en el futuro [66].

Al combinar esta estrategia con otras se diseñó un dispositivo de alta resistencia que es capaz de soportar todo el peso de los robots humanos o humanoides cuando los sujetos están de pie o caminando. Este dispositivo fue modelado a través de la aplicación de flexión en voladizo en configuración paralela para soportar la gran fuerza e implementar la medición. Tiene tres grados de libertad que se usan convencionalmente para validar la fuerza de reacción del terreno (GRF) y el centro de presión (COP) y controlar los robots humanoides. Además, la dimensión del sensor en dirección vertical es tan pequeña como 10 mm, lo que hace que sea conveniente para fijarlo a los zapatos o al pie robot humanoide y se describen el principio de detección y el diseño de la estructura de este sensor [30]. Además, se han generado interfaces hápticas para realidad virtual han sido diseñadas para la interacción manual. Sin embargo, los pies pueden ser otra forma habitual de interactuar con el entorno real. Al usar los pies como una forma de manipular la realidad virtual, es necesario detectar la intención del usuario (por ejemplo, la fuerza de reacción del suelo (GRF), el centro de presión (COP) o la aceleración). Aunque se han realizado algunos estudios sobre el desarrollo de dispositivos de pie para detectar la intención del usuario, hubo algunas limitaciones que no pueden detectar la intención de movimientos extremos (por ejemplo, correr o saltar) debido a la baja capacidad de GRF y la debilidad al impacto. Los dispositivos para pies con cuatro sensores GRF, que pueden medir GRF grandes y robustos al impacto, y un acelerómetro comercial para detectar la intención humana. Para evitar que los movimientos del usuario sean perturbados, GRF y la aceleración del dispositivo de pie pueden transmitirse a la PC a través de un Bluetooth. Se estimó el CoP de GRF medida por el dispositivo desarrollado para investigar el rendimiento del dispositivo y medimos la aceleración de tres ejes (anterior-posterior, medial-lateral y vertical) [50].

Como se observa se han realizado múltiples trabajos relacionados a esta temática ([37, 38, 40, 48]) y han permitido la generalización de conceptos para realizar otras aplicaciones ([19, 29, 64]). Entre ellas, una turbina oceánica extrae la energía cinética de las corrientes oceánicas para generar electricidad. Las señales de vibración de la turbina contienen una gran cantidad de información con respecto a su estado, y la detección de cambios en estas señales es crucial para la detección oportuna de fallas. Las transformadas de wavelet proporcionan un medio para analizar estas señales complejas y extraer características que son representativas de la señal. Se requieren técnicas de selección de características una vez que se extraen estas características de wavelets para eliminar características redundantes o inútiles antes de presentar los datos a un algoritmo de aprendizaje automático para el reconocimiento y clasificación de patrones. Esto reduce la cantidad de datos a procesar y, a menudo, incluso puede aumentar la capacidad del aprendiz de la máquina para detectar el estado actual de la máquina. Este artículo compara empíricamente ocho algoritmos de selección de características en datos de vibración transformados en wavelets que se

originan en una plataforma de prueba en tierra para una turbina oceánica. Un estudio de caso muestra el rendimiento de la clasificación de siete aprendices de máquina cuando se entrena en los conjuntos de datos con un número variable de características seleccionadas del conjunto original de todas las características. Los resultados resaltan que al elegir una técnica de selección de características apropiada y aplicarla a las tres características más importantes (3,33 % del conjunto de características original), algunos clasificadores, como el árbol de decisión y el bosque aleatorio, pueden diferenciar correctamente entre estados defectuosos y no defectuosos. Casi el 100 % del tiempo. Estos resultados también muestran las diferencias de rendimiento entre los diferentes algoritmos de selección de características y las combinaciones de clasificadores [65].

1.4. Resumen del Capítulo

Como se observó en este Capítulo, los análisis de posturografía permiten determinar un conjunto de patologías en pacientes. Estos análisis se realizan con equipos especializados que en su mayoría son bastante costosos y quedan fuera del alcance de la gente común, debido a esto, se han propuesto diversas soluciones de bajo costo y entre ellas la que se presenta en este documento. En el siguiente, Capítulo se describe más detalladamente la aplicación desarrollada y su relación con los conceptos expuestos en este Capítulo.

Descripción de la aplicación desarrollada

2.1. Introducción

Debido a la necesidad de conocer parámetros cinemáticos en diferentes posiciones se propone una aplicación con entorno gráfico, la cual determina ciertas características del usuario. Esta aplicación se desarrollo enteramente en software libre y se describe detalladamente en este Capítulo, el cual se organiza de la siguiente manera; En la sección 2.2 se describe el sistema operativo y las librerías necesarias para el funcionamiento adecuado de la aplicación, en la sección 2.3 se realiza un análisis del código desarrollado (ver Anexo B) y en la sección 2.4 se presenta una descripción del protocolo de comunicación usado en este aplicación.

2.2. Pre-requisitos de la aplicación

Esta aplicación se desarrollo enteramente en software libre y funciona en el sistema operativo Ubuntu. El lenguaje de programación usado se llama PYTHON y se implementó en la versión 3.58, esta versión no es la más reciente pero es la única compatible con las librerías requeridas. PYTHON es un lenguaje de programación interpretado, es decir, las rutinas se usan convencionalmente dependen de otras función que funcionan en segundo plano durante el funcionamiento del compilador. Además, es un lenguaje que soporta la programación orientada a objetos y la creación de clases e instancias. En otras palabras, PYTHON permite definir un objeto mediante clases y asignarle métodos o funciones para su operación, cuyas instancias dependen de las siguiente librerías [52].

- **Pybluez:** Esta librería se encarga de establecer la comunicación bluetooth e implementa el protocolo L2CAP que es el usado por la nintendo Wii Balance Board para comunicarse con el computador.
- **Openpyxl:** Es una librería que le permite a PYTHON escribir archivos .xls o .xlsx.
- **Drawnow:** Es una función que le permite a PYTHON actualizar las gráficas en tiempo real y no en segundo plano como se realiza convencionalmente.

- **Pandas:** Es una librería que permite crear bases de datos y almacenarlos en segundo plano, para luego guardarlos en un archivo de texto plano.
- **Matplotlib:** Es la librería que le permite a PYTHON realizar graficas e tendencia.
- **Pygame:** Es la librería de video-juegos de PYTHON y permite leer Control-PAD o joystick.
- **Numpy:** Es la librería de procesamiento numérico, además, permite realizar operaciones vectoriales y matriciales como en programas como MATLAB.

Se utilizó el sistema operativo Ubuntu debido a que es más estable que el sistema operativo Windows, es decir, Ubuntu no se bloquea al conectarse con la nintendo Wii Balance Board como lo hace Windows. Otra ventaja es que las aplicaciones desarrolladas se ejecutan más rápidamente y para instalar las librerías puede usarse el terminal de Ubuntu (Fig. 2.1). Una vez instaladas las librerías es posible programar en el lenguaje PYTHON, en este caso, se utilizo el procesador de texto PYTHON IDLE que tiene dos interfaces; la primer interface (Fig. 2.2(a)) es el terminal donde se ejecutan las instrucciones directamente y la segunda es el procesador de texto plano que es donde se pueden digitar las instrucciones (Fig. 2.2(b)) de la aplicación en desarrollo para luego ejecutarse [52].

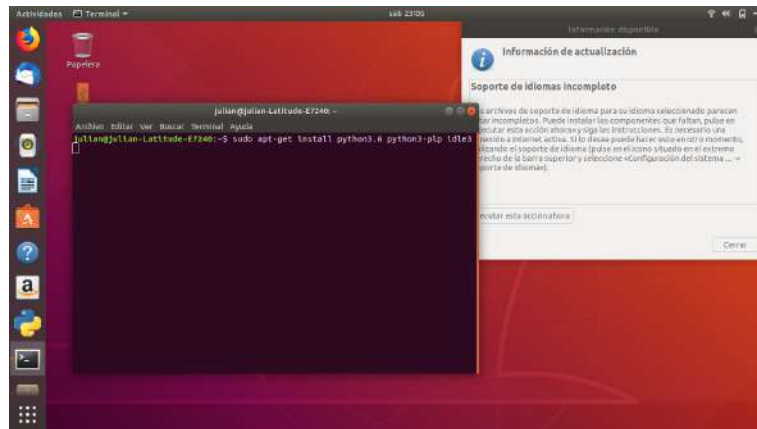
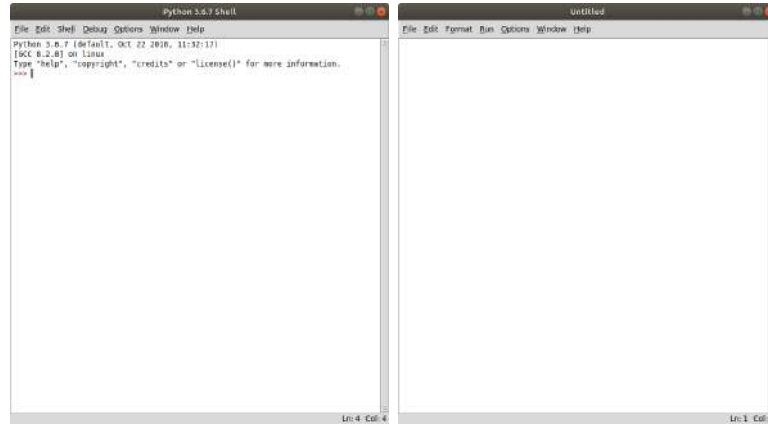


FIGURA 2.1. Terminal de Ubuntu (Fuente los autores).

2.3. Funcionamiento de la aplicación

Como se mencionó la aplicación desarrollada utiliza funciones de las librerías para ejecutar ciertas instancias de las clases. En total son dos tres clases y cada una de ellas se encuentra extendida a la clase principal para que herede sus funciones, es decir, hay una clase que contiene gran parte de las funciones de la aplicación desarrollada. Estas clases, la función principal de la aplicación y su relación con la nintendo Wii Balance Board (NWBB) se describen detalladamente a continuación.

- **Función principal:** Está función se encarga de inicializar las variables e invocar la clase que se encarga de construir la interfaz gráfica de usuario.
- **Clase 1 (Generación de interfaz gráfica de usuario):** Esta clase contiene todos los parámetros de la interfaz gráfica como los avisos, la forma de los botones,



(a) Terminal de PYTHON IDLE. (b) Procesador de texto de PYTHON IDLE.

FIGURA 2.2. PYTHON IDLE (Fuente los autores).

los colores de la ventana y todas las variables que se asocian a la creación de la interfaz gráfica. Esta clase se compone de las siguientes instancias:

- *Función `_init_`*: Esta función se encarga de generar todos los valores iniciales de las variables de la clase y de crear los objetos de la interfaz gráfica, tales como: botones, etiquetas o ventanas.
- *Función `Iniciar Bluetooth`*: Esta función se encarga de detectar los dispositivos bluetooth disponibles para establecer una conexión y tiene dos modos de funcionamiento. Por un lado, cuando detecta la NWBB se conecta con el dispositivo, permite la ejecución de la Clase 2 y cierra el puerto para que ningún otro dispositivo se conecte al computador. Por otro lado, cuando no detecta la NWBB simplemente envía un mensaje al usuario a través de la interfaz gráfica y detiene la ejecución del programa hasta que el dispositivo este disponible.
- **Clase 2 (Interfaz de acople)**: Esta clase simplemente inicializa la clase que permite la comunicación con la NWBB y mantiene la ejecución del programa de manera continua.
- **Clase 3 (Comunicación)**: Esta clase implementa todas las funciones que establecen el protocolo de configuración, ajuste y comunicación entre el computador y la NWBB. Las instancias que componen esta clase se definen a continuación:
 - *Función `_init_`*: Está función se encarga de inicializar todas las variables de la clase.
 - *Función `Conectar`*: Está función se encarga de establecer la conexión entre la NWBB y el computador, una vez se ha validado que el puerto (Bluetooth del computador) esta disponible por la Clase 1. Además de generar los avisos con las indicaciones de usuario para que se realice la calibración de la NWBB.
 - *Función `Enviar`*: Está función codifica y empaqueta la información en el protocolo de la NWBB antes de ser enviado a través del puerto.
 - *Función `Calibrar`*: Está función envía un mensaje a la NWBB para indicar que se inicio el proceso de calibración y deja el programa en espera hasta que la NWBB responda.

- *Función Estado*: Esta función termina el envío consecutivo de mensajes de la NWBB y permite conocer el estado de carga de la batería y el estado del botón principal de la NWBB.
- *Función Luz*: Esta función permite encender o apagar el LED azul del botón principal de la NWBB.
- *Función Reporte*: Esta función envía un mensaje a la NWBB que indica su operación continua, es decir, al enviar este mensaje la NWBB envía mensajes consecutivos con la información del peso detectado por cada galga.
- *Función loop*: Esta función procesa la información de la NWBB cuando hay algún dato en el buffer del puerto, es decir, cada vez que la NWBB envía algún mensaje al computador, esta función lo decodifica para traducirlo en números reales o información para el usuario. Esta información está codificada de acuerdo con las especificaciones del fabricante, por lo tanto, se describen detalladamente en el siguiente numeral.
- *Función Asignar Calibración*: Esta función almacena los valores que retorna la NWBB al ejecutar la función Calibrar, ya que, permiten corregir el margen de error al calcular la peso sobre cada galga con la función calcMass.
- *Función Sensores*: Esta función almacena en un arreglo temporal el valor en números reales del peso sobre cada galga, el centro de presión y el peso de la persona.
- *Función calcMass*: Esta función estima el valor del peso sobre cada galga en números reales, a partir de estos datos el peso de la persona y el centro de presión.
- *Función fuerzatracción*: Esta función determina el centro de presión a partir de la información recolectada por los sensores.
- *Función figura*: Esta función gráfica el comportamiento de cada galga, el peso y el centro de presión.
- *Función COP*: Esta función refresca la gráfica para que presente la información en tiempo real.
- *Función ExportarExcel*: Esta Función toma la información del arreglo temporal y la exporta en formato xls. o texto plano.
- *Función Cerrar*: Esta función cierra la interfaz gráfica y abre el puerto para que este disponible nuevamente.
- *Función _del_*: Elimina las variables almacenadas en el disco duro.
- *Función _enter_*: Retorna las funciones de la clase.
- *Función _exit_*: Destruye la Clase 3 únicamente para iniciar la comunicación Bluetooth nuevamente.

La forma en la que interactúan estas instancias entre sí, para que la aplicación funcione apropiadamente se presenta en la Fig. 2.3 que presenta una descripción general del funcionamiento. Sin embargo, el funcionamiento no sería posible sin el protocolo de comunicación y la forma de empaquetar la información para que el computador y la NWBB se puedan entender, por lo tanto, el protocolo implementado será descrito en el siguiente numeral.

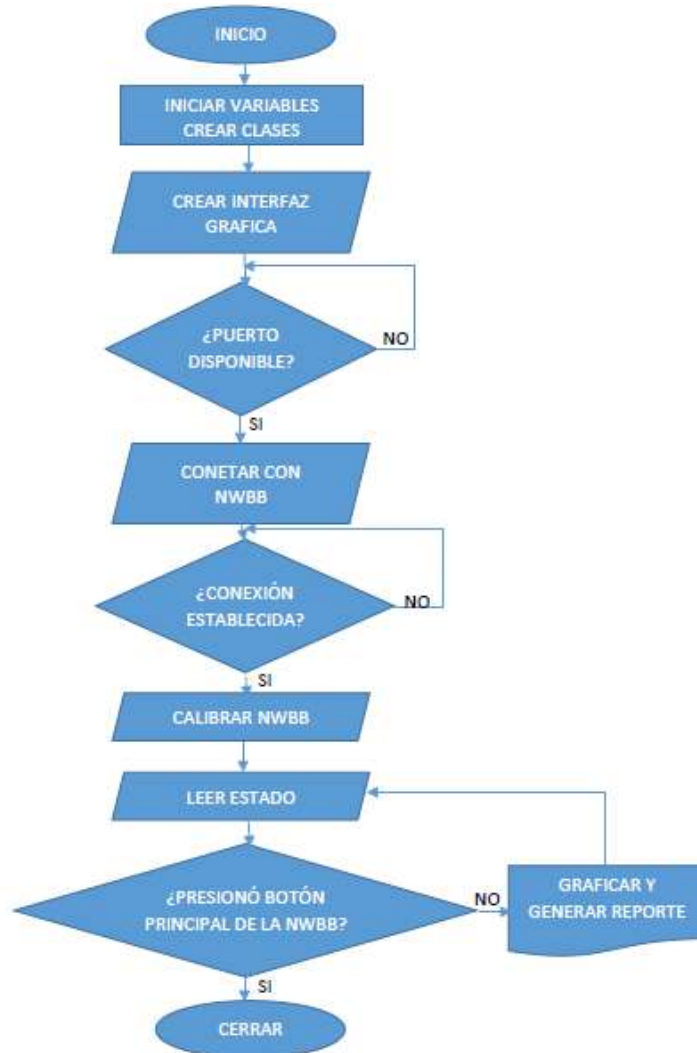


FIGURA 2.3. Funcionamiento resumido de la aplicación (Fuente los autores).

2.4. Protocolo de comunicación

Un protocolo de comunicación es una forma de estandarizar la transmisión de mensajes entre dos dispositivos, en este caso se implemento el protocolo L2CAP que es el nativo de la Nintendo Wii Balance Board (NWBB). Sin embargo, el fabricante establece que la comunicación entre el computador y la NWBB se realice mediante paquetes de datos que tienen referencias basadas en números. Estos números indican el tipo de contenido, como se debe interpretar la información dentro de cada mensaje y se describen a continuación [57].

- COMMAND_LIGHT = b'x11'
- COMMAND_REPORTING = b'x12'
- COMMAND_REQUEST_STATUS = b'x15'
- COMMAND_REGISTER = b'x16'

- `COMMAND_READ_REGISTER = b'x17'`
- `INPUT_STATUS = b'x20'`
- `INPUT_READ_DATA = b'x21'`
- `EXTENSION_8BYTES = b'x32'`
- `BUTTON_DOWN_MASK = 0x08`
- `LED1_MASK = 0x10`
- `CONTINUOUS_REPORTING = b'x04'`

Los números están en formato hexadecimal y se envían o reciben en forma de cadenas de texto. Por un lado, cuando se envía una cadena de texto se antepone el número `b'x52'` para indicarle a la WNBB que el computador va a enviar un mensaje, por ejemplo, para encender el LED del botón principal de la WNBB la cadena que debe enviarse debe tener la siguiente forma `"b'x52'+b'x11'+0x10"`. Por otro lado, para que la WNBB envíe un mensaje con información de interés se debe enviar el registro `b'x17'` + lo que se desea conocer. La única excepción a esta regla es la calibración, ya que, dos estructuras únicas que son:

1. `COMMAND_READ_REGISTER+b"x04xA4x00x24x00x18"`
2. `COMMAND_REGISTER+b"x04xA4x00x40x00"`

De las cuales, la primera le indica a la plataforma que se va a poner un objeto para calibrar y la segunda le pide a la plataforma que envíe el margen de error. Cuando se pide información a la WNBB siguiendo la estructura general, el dispositivo responde con un mensaje de la siguiente forma.

1. `xAx00x00xB`

Donde A es el registro mencionado en el primer listado de este numeral y B es la información de interés. La información de interés está dada por cinco (5) números enteros de ocho (8) bits, es decir, B tiene cinco componentes los cuales se deben separar teniendo en cuenta que el último componente indica el estado del botón principal de la WNBB (`0x00`=Apagado, `0x08`=Encendido). Los demás componentes de B indican el valor de cada galga que en su orden se muestran en la Fig. 2.4 y cada uno de ellos se corrige con base en los resultados de la calibración.

En el caso de B1 y B2 para estimar el valor de salida de cada galga se multiplica el valor de cada componente de B * 17 (Es un valor dado en la hoja técnica) y para estimar B3 y B4 a cada componente de B se suma y multiplica por 17 (Especificado en la hoja técnica) [57]. Finalmente el funcionamiento de esta aplicación se resume en el algoritmo 1.

2.5. Resumen del capítulo

En este Capítulo se presentó una descripción de la aplicación desarrollada, su funcionamiento e implementación. Sin embargo, se debe tener en cuenta que la versión extendida



FIGURA 2.4. Componentes de B (Fuente los autores).

Algoritmo 1: Estructura de la aplicación desarrollada (Fuente los autores).

```

1 Función APP(t)
2   Crear Clases
3   Inicializar variables
4   Iniciar Interfaz gráfica
5   mientras No pulse botón de salir hacer
6     si Bluetooth disponible entonces
7       Conectarse a la NWBB
8       si Conexión exitosa entonces
9         Calibrar NWBB
10        mientras No pulse botón principal de la NWBB hacer
11          Codificar mensaje para solicitar información a la WNBB
12          Enviar mensaje
13          Esperar respuesta de la NWBB
14          Clasificar información de llegada
15          Estimar valor de carga de los sensores
16          Calcular centro de presión
17          Calcular peso corporal
18          Graficar y almacenar en archivo de texto plano
19        fin
20      fin
21    en otro caso
22      | Publicar mensaje de error
23    fin
24  fin
25  en otro caso
26  | Publicar mensaje de error
27  fin
28 fin

```

del código y el manual de instrucción para su uso adecuado se encuentran en los anexos A y B. Esta aplicación funcionó de acuerdo con las expectativas y objetivos propuestos, lo cual se demuestra en el siguiente Capítulo mediante la presentación de los experimentos y resultados obtenidos.

Experimentación y resultados

3.1. Introducción

En el Capítulo anterior se describió la aplicación desarrollada y su funcionamiento. Para comprobar su funcionamiento se realizaron varios experimentos y sus resultados se registraron como se muestra en este Capítulo, el cual está organizado de la siguiente manera; en la sección 3.2 se describe la metodología empleada para realizar las pruebas, en la sección 3.3 se presentan las gráficas, tablas e imágenes relacionadas a los resultados obtenidos en las pruebas y en la sección 3.4 se realiza un análisis de los resultados obtenidos.

3.2. Experimento

Las pruebas realizadas a la aplicación son sencillas y consisten en una calibración indirecta, la cual permite determinar si el valor de peso detectado por cada galga es el indicado o tiene un margen de error. Este margen de error se estimó de dos maneras: La primera consiste en medir el peso de tres (3) personas diferentes en una báscula (en este caso fue una báscula digital Fig. 3.1) y comparar con el reportado por la aplicación. La segunda consiste en variar el centro de presión del usuario y verificar que el indicador de la gráfica efectivamente se desplace de acuerdo con la intensidad de la fuerza en cada galga.



FIGURA 3.1. Báscula usada en los experimentos (Fuente los autores).

Estos resultados se reportan en la aplicación en cuatro gráficas diferentes como se observa en la Fig. 3.2. En la primer gráfica se presenta el comportamiento de los cuatro sensores, el cual se actualiza cada segundo. En la segunda gráfica se muestra el comportamiento del peso (en kilogramos) del usuario u objeto que se encuentre sobre la nintendo Wii Balance Board. En la tercer gráfica se observa la ubicación del centro de presión del usuario (en términos de coordenadas rectangulares) y en la cuarta gráfica se representa la intensidad de la fuerza de atracción en cada sensor, es decir, en cada esquina de la gráfica hay un círculo que aumenta o disminuye su radio dependiendo de la fuerza superficial que detecte la galga que corresponda (cada esquina esta sincronizada con una galga diferente).

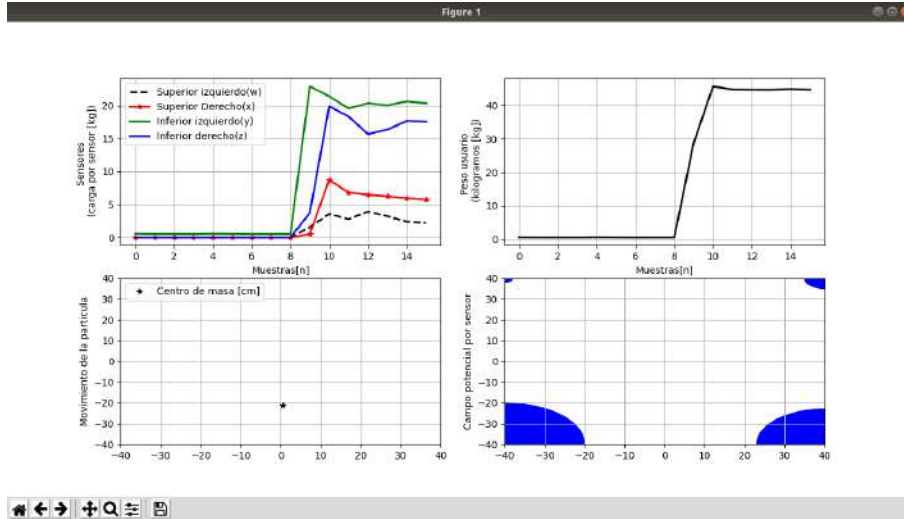


FIGURA 3.2. Gráficas generadas con la aplicación (Fuente los autores).

En segundo plano, la aplicación construye un arreglo numérico para exportar un histórico de los datos medidos y reportados en las gráficas en un archivo de texto plano (Fig. 3.3). Las pruebas reportadas en este documento se realizaron durante intervalos de tiempo de un (1) minuto, es decir, cada archivo de texto plano tiene 60 registros y se utilizaron para construir las gráficas presentadas en la siguiente sección.

Figure 3.3 shows a spreadsheet titled 'Centro de Masa.xlsx - LibreOffice Calc'. The table has columns A through H. Column A is 'Tiempo' (Time) from 0 to 30. Column B is 'Galga superior izquierda'. Column C is 'Galga superior derecha'. Column D is 'Galga inferior izquierda'. Column E is 'Galga inferior derecha'. Column F is 'Peso'. Column G is 'Centro de masa (x)'. Column H is 'Centro de masa (y)'. The data shows a significant increase in weight and center of mass coordinates starting around sample 10.

A	B	C	D	E	F	G	H
0	Galga superior izquierda	Galga superior derecha	Galga inferior izquierda	Galga inferior derecha	Peso	Centro de masa (x)	Centro de masa (y)
1	0	0	0.55518018018018	0	0.55518018018018	-0.392571670185775	-0.392571670185775
2	0	0	0.526463963963964	0	0.526463963963964	-0.372266238969269	-0.372266238969269
3	0	0	0.526463963963964	0	0.526463963963964	-0.372266238969269	-0.372266238969269
4	0	0	0.516891891891892	0	0.516891891891892	-0.365407761897101	-0.365407761897101
5	0	0	0.516891891891892	0	0.516891891891892	-0.365407761897101	-0.365407761897101
6	0	0	0.564752252252252	0	0.564752252252252	-0.399240147257994	-0.399240147257994
7	0	0	0.538639038639039	0	0.538639038639039	-0.379034718014144	-0.379034718014144
8	0	0	0.516891891891892	0	0.516891891891892	-0.365407761897101	-0.365407761897101
9	0	0	0.516891891891892	0	0.516891891891892	-0.365407761897101	-0.365407761897101
10	0	0	0.526463963963964	0	0.526463963963964	-0.372266238969269	-0.372266238969269
11	1.55667225013959	0.52019332566168	22.894904578448	3.72813660714286	28.69808264071808	-14.2731078649008	-17.3441436645308
12	3.5689580056393	8.72407123130035	21.385556514908	10.8678097345133	45.626260320282	2.57285429607296	-20.477205858177
13	2.7745299756504	6.0702020863061	19.524228950459	16.325774250282	44.6991284816249	1.99943288750525	-19.997440858705
14	3.92044024120905	6.47524991020969	20.551858214515	15.4613683920571	44.651177258369	1.50165583645396	-18.0942540827228
15	3.22724734785036	6.24050832911302	20.0184254606365	16.3454241671429	44.6279405069857	-0.466508309524	-19.018411719839
16	2.39195979894897	5.96662830840046	20.606923506421	17.65185840708	44.8304598047721	0.44259994412476	-21.1471039703151
17	2.21162862367393	5.72209426132487	20.22106387493	17.54935982309	44.6508387603648	0.31878385307952	-21.1660217240474
18	2.25059743160246	5.53624856156502	20.3601340035501	17.0752212389381	44.6550841772068	0.01466423891151	-20.978804077749
19	2.59128978224456	6.35788242370541	20.0658849604578	16.0797991071429	44.5964203326537	-0.15520501124725	-19.2308378447144
20	0	0	0	0	0	0	0
21	0	0	0	0	0	0	0
22	0	0	0	0	0	0	0
23	0	0	0	0	0	0	0
24	0	0	0	0	0	0	0
25	0	0	0	0	0	0	0
26	0	0	0	0	0	0	0
27	0	0	0	0	0	0	0
28	0	0	0	0	0	0	0
29	0	0	0	0	0	0	0
30	0	0	0	0	0	0	0
31	0	0	0	0	0	0	0
32	0	0	0	0	0	0	0
33	0	0	0	0	0	0	0
34	0	0	0	0	0	0	0
35	0	0	0	0	0	0	0
36	0	0	0	0	0	0	0
37	0	0	0	0	0	0	0
38	0	0	0	0	0	0	0
39	0	0	0	0	0	0	0
40	0	0	0	0	0	0	0

FIGURA 3.3. Segmento de los datos almacenados por la aplicación (Fuente los autores).

3.3. Resultados obtenidos

Inicialmente se midió el peso reportado por la bascula, se comparo con el valor máximo reportado por la aplicación (teniendo en cuenta que cada galga mide 1/4 del peso total, se deben sumar los 4 pesos medidos) y se construyó la tabla 3.1 para comparar el error de medida. Sin embargo, estos resultados se pueden observar gráficamente en la Fig. 3.4.

TABLA 3.1. Error porcentual estimado por sujeto (Fuente los autores).

Usuario	Valor medido (bascula)	Valor medido (Aplicación)	Error porcentual
Sujeto 1	45 kg	44.5125 kg	1.0833 %
Sujeto 2	62 kg	61.2324 kg	1.2380 %
Sujeto 3	90 kg	88.9725 kg	1.1416 %

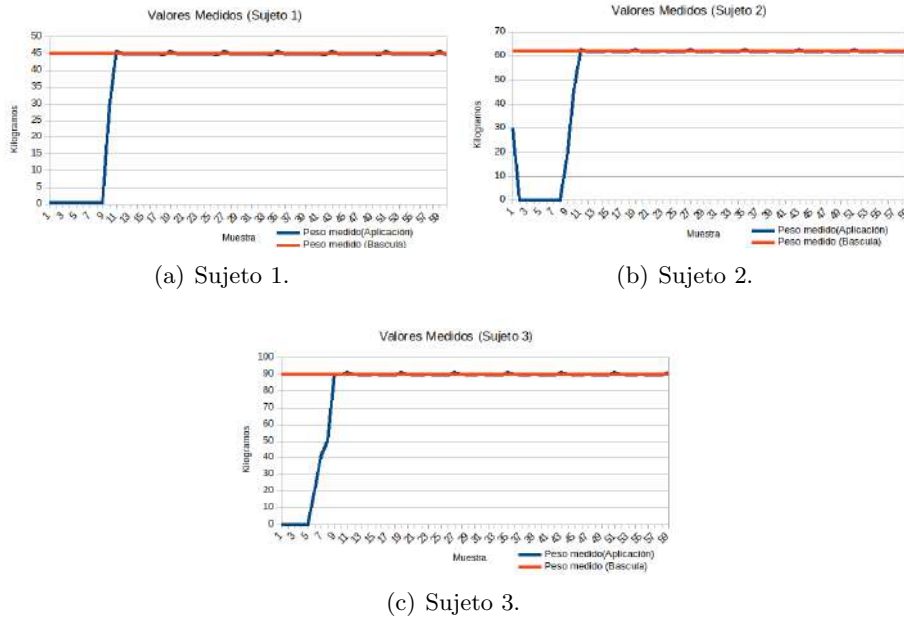


FIGURA 3.4. Comportamiento del peso observado usando la aplicación (Fuente los autores).

Finalmente, los resultados medidos del desplazamiento del centro de presión se muestran en la Fig. 3.5. Sin embargo, se debe tener en cuenta que en la gráfica se presentan todas las posiciones leídas a lo largo de la prueba y en la aplicación solamente se ve la ubicación actual, esto con el fin de realizar un análisis en el siguiente numeral.

3.4. Análisis de los resultados

Como se observa el margen de error entre el valor del peso medido por una bascula y el reportado por la aplicación es menor al 2 %, lo que indica que es lo suficientemente confiable como para estimar el centro de presión adecuadamente. Otra característica de la aplicación es que permite determinar el centro de presión en puntos del centro de la plataforma, lo cual no se ha logrado en aplicaciones como el CU BrainBLoX. El CU BrainBLoX permite estimar el centro de presión alrededor de un rectángulo ignorando

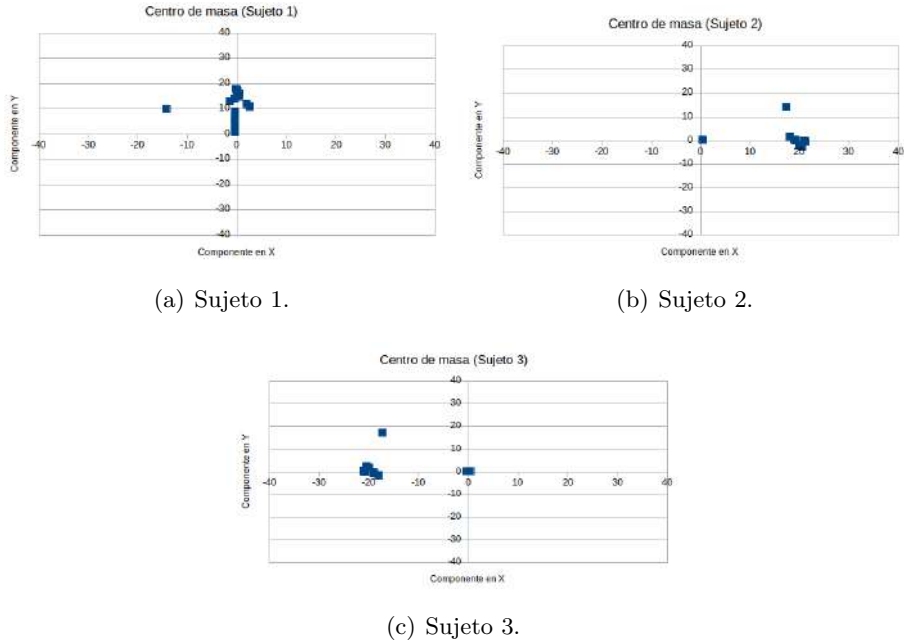


FIGURA 3.5. Comportamiento del centro de presión (Fuente los autores).

las medidas que pueda tener en el centro de la nintendo Balance Board. Este fenómeno se observa más claramente en las Figs. 3.6, 3.7, 3.8 , donde se ponen sobre la plataforma diferentes cargas y se presenta el área de trabajo del CU BrainBLoX (software disponible en [13]).

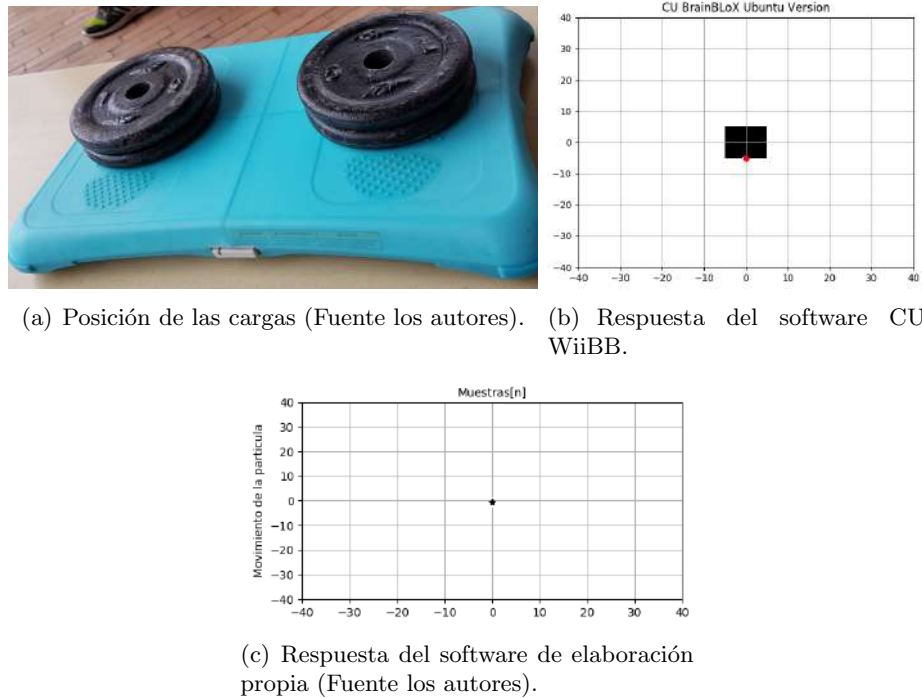
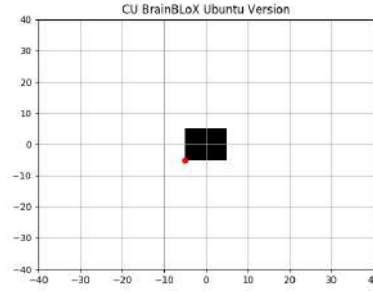
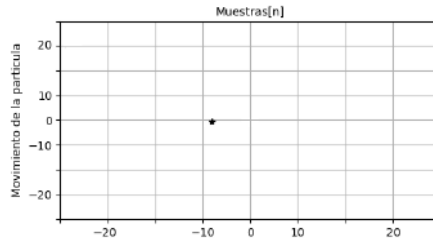


FIGURA 3.6. Posición de las cargas durante la realización de la primer prueba y los resultados obtenidos en ambas aplicaciones.

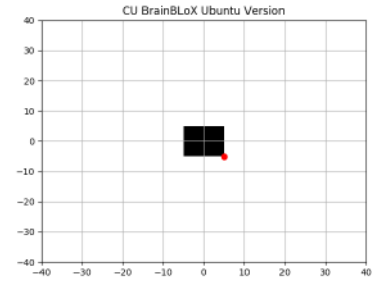


(a) Posición de las cargas (Fuente los autores). (b) Respuesta del software CU WiiBB.

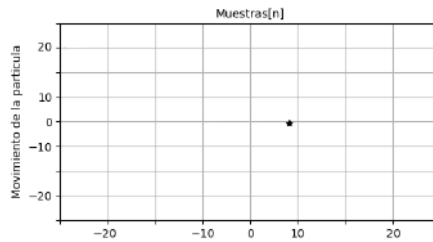


(c) Respuesta del software de elaboración propia (Fuente los autores).

FIGURA 3.7. Posición de las cargas durante la realización de la segunda prueba y los resultados obtenidos en ambas aplicaciones.



(a) Posición de las cargas (Fuente los autores). (b) Respuesta del software CU WiiBB.

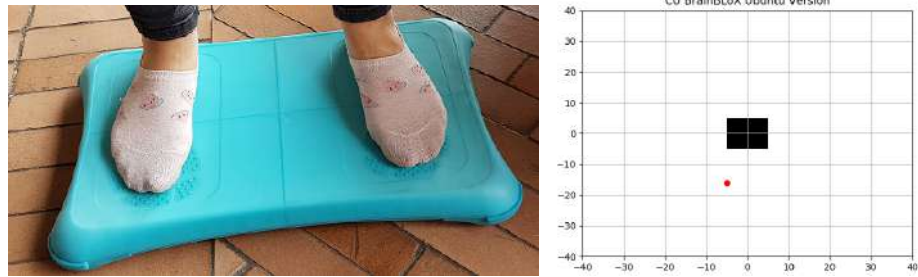


(c) Respuesta del software de elaboración propia (Fuente los autores).

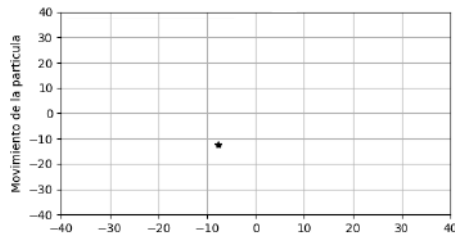
FIGURA 3.8. Posición de las cargas durante la realización de la tercer prueba y los resultados obtenidos en ambas aplicaciones.

Al revisar las gráficas de la Fig. 3.5 se observa variabilidad en los resultados, por lo tanto, se incorporó una metodología que permite la repetibilidad de los resultados con

cuatro (4) masas de 2.5 Kg. Estas masas son configurables y permiten aumentar o disminuir la presión sobre ciertos puntos de la plataforma, lo cual permite tomar una medida precisa de los resultados. Sin embargo, el rango de visibilidad es reducido, ya que, el umbral de las galgas se supera solamente por 0.5Kg. Por lo tanto, el protocolo de prueba implementado con las masas también se siguió con un paciente para recargar puntos de presión sobre la plataforma y así determinar el margen de error entre la aplicación desarrollada y la CU BrainBLoX. Este fenómeno se reporta en las Figs. 3.9, 3.10, 3.11 donde se observa un error menor al 5% entre las dos aplicaciones al medir objetos que pesen más de 10Kg.



(a) Posición de las cargas (Fuente los autores). (b) Respuesta del software CU WiiBB.



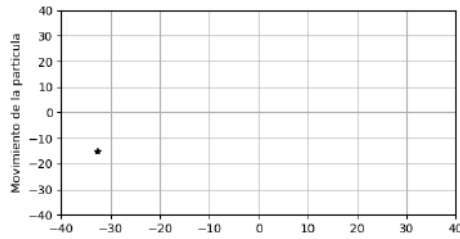
(c) Respuesta del software de elaboración propia (Fuente los autores).

FIGURA 3.9. Posición de las cargas durante la realización de la primer prueba y los resultados obtenidos en ambas aplicaciones.

Finalmente, las otras características del paciente reflejan un comportamiento repetible como se muestra en la Fig. ??, las cuales permiten determinar su peso y centro de presión plantar adecuadamente.



(a) Posición de las cargas (Fuente los autores). (b) Respuesta del software CU WiiBB.

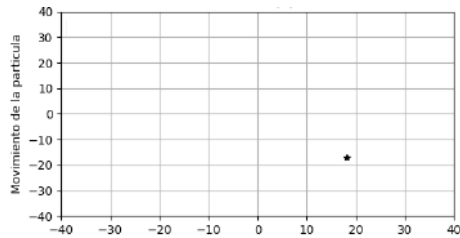


(c) Respuesta del software de elaboración propia (Fuente los autores).

FIGURA 3.10. Posición de las cargas durante la realización de la segunda prueba y los resultados obtenidos en ambas aplicaciones.

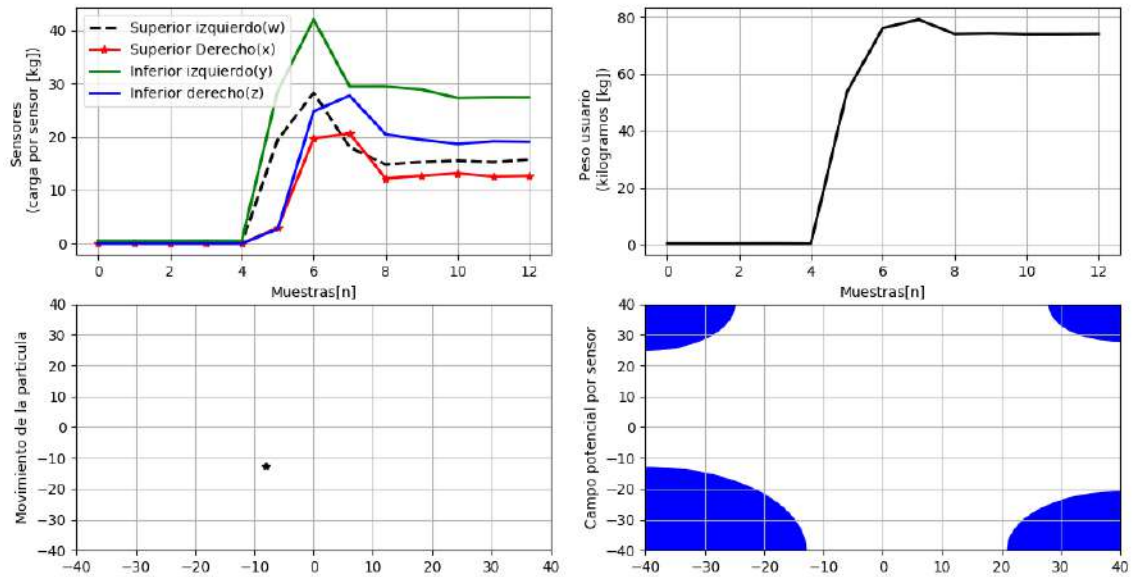


(a) Posición de las cargas (Fuente los autores). (b) Respuesta del software CU WiiBB.



(c) Respuesta del software de elaboración propia (Fuente los autores).

FIGURA 3.11. Posición de las cargas durante la realización de la tercer prueba y los resultados obtenidos en ambas aplicaciones.



(a) Posición de las cargas (Fuente los autores).

3.5. Resumen del Capítulo

Como se observó en este Capítulo la aplicación funciona correctamente y los sensores fueron calibrados para que el margen de error fuera cercano a cero (0). Estas características y los aportes más significativos de este trabajo se presentan detalladamente en las conclusiones de este trabajo.

Conclusiones

- La aplicación desarrollada permite encontrar el parámetros del usuario a partir de la información de los sensores, esto resulta ventajoso para realizar análisis cinéticos que se basen en el peso y en el centro de presión. Estos parámetros se estimaron mediante la implementación de un sistema de tres (3) clases, las cuales desempaquetan la información enviada por la nintendo Wii Balance Board, realizan una transformación matemática basada en la fuerza gravitatoria e incorporan un mecanismo para convertir la información de salida de la transformación matemática en el centro de presión.
- La transformación matemática se construye al modelar cada sensor como una masa con densidad y peso variable, al incrementar el valor de masa de cada sensor de forma independiente estos atraen la partícula de masa constante (centro de presión) dependiendo de su fuerza de atracción. Esta fuerza permite que se estimen valores del centro de presión hacia el centro de la plataforma, lo cual no se ha logrado perfectamente con otras aplicación que implementen una nintendo Wii Balance Board reportadas en el estado del arte.
- Esta aplicación es lo suficientemente versátil como para que otras personas sigan trabajando en esta línea de investigación, ya que, se basa enteramente en software libre y se desarrollo en el lenguaje de programación PYTHON que es muy intuitivo para cualquier persona que lo desee manejar.

Trabajo Futuro

Aunque la aplicación funcionó correctamente, para continuar con este trabajo se proponen las siguientes ideas de investigación:

- *Actualizar las librerías de PYTHON:* Como se mencionó esta aplicación funciona únicamente en el sistema operativo Ubuntu y para volverla aun más versátil se deberían actualizar las librerías para que funcione en el sistema operativo Windows.
- *Complementar el análisis de marcha:* Esta aplicación estima parámetros cinéticos únicamente, lo cual se puede complementar con el equipo TECHENAID del laboratorio de electrónica para realizar análisis cinéticos y cinemáticos, lo cual se puede lograr fácilmente, ya que, la aplicación desarrollada es modular y permite incorporar más módulos.
- *Implementar tecnologías basadas en IoT:* Estas tecnologías permitirían consultar los reportes realizados por la aplicación en tiempo real desde un computador que se encuentre en otra red.

Instrucciones de uso

La aplicación desarrollada al usuario los pasos necesarios para que funcione adecuadamente. Sin embargo, en este documento se realiza un recuento detallado de todas las indicaciones que se deben seguir para que la aplicación funcione adecuadamente. Estas indicaciones se mencionan a continuación.

- Encuentre una superficie seca, rígida, horizontal y libre de obstáculos sobre la que pueda usar la Wii Balance Board (ver Fig. A.1). Tenga en cuenta que su computador debe estar en un radio de máximo veinte (20) metros en línea de vista.



FIGURA A.1. Posición de trabajo de la Wii Balance Board.

- La aplicación funciona en el sistema operativo Ubuntu con el interprete PYTHON únicamente, así que abra el terminal, digite la palabra Python y por ultimo escriba la palabra "Programa.py"(ver Fig. A.2).

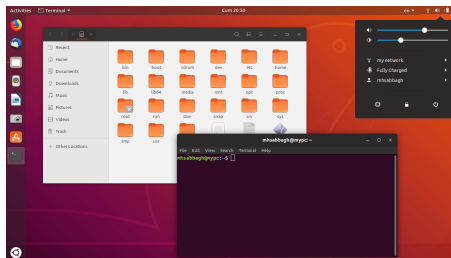


FIGURA A.2. Terminal del sistema operativo Ubuntu.

- Al abrir la aplicación fíjese que en la parte inferior debajo del rotulo ‘estado:’ aparecen las indicaciones que debe seguir a medida que la aplicación se este ejecutando. Una vez abierta la aplicación pulse el botón iniciar o el rojo para cerrar (ver Fig. A.3).



Autores:

Jenny Katherine Torres Zambrano
Julian Oswaldo Perez De la Cuadra

Directora:

Ing. Lely Adriana Luengas Contreras PhD.

Seleccione una opción para comenzar.



Estado:

Para iniciar pulse el botón verde (Iniciar).



FIGURA A.3. Presentación de la aplicación.

- Al intentar conectar la aplicación con la Wii Balance Board tenga en cuenta que las baterías deben estar puestas y cargadas (ver Fig. A.4).
- Una vez haya pulsado el botón verde, el programa le indicara que debe sincronizarse con la Wii Balance Board, esto se consigue pulsando el botón rojo que se encuentra en el compartimento de las baterías en los cinco (5) segundos despues de pulsar el botón verde (ver Fig. A.5).
- Si al intentar establecer la conexión el proceso fue exitoso la aplicación mostrara un mensaje de éxito (ver Fig. A.6), en otro caso le pedirá que revise la Wii Balance Board (ver Fig. A.7).
- Al establecer la conexión espere a que el sistema se ajuste y encuentre un nivel de cero, esto se realiza automáticamente siguiendo un protocolo definido en la aplicación (ver Figs. A.8(a), A.8(b) y A.8(c)). Cuando finaliza su calibración la aplicación le indicara que ponga la Wii Balance Board en el suelo (ver Fig. A.8(d)).

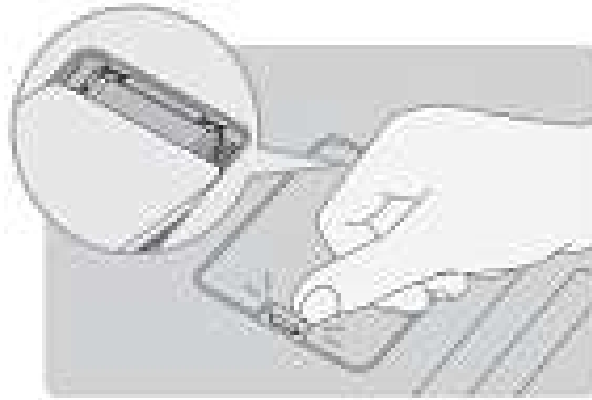


FIGURA A.4. Forma de cambiar las baterías (Esta imagen fue tomada de [57]).

Estado:

Pulse el botón SYNC de la Wii balance board (Detectando dispositivos espere 5 segundos).

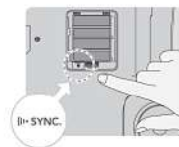


FIGURA A.5. Sincronización.

Estado:

Conectado a: Nintendo RVL-WBC-01



FIGURA A.6. Éxito.

Verifique el estado de las baterías y pulse el botón verde (Iniciar) nuevamente.



FIGURA A.7. Fracaso.

- Finalmente, cuando el programa lo indique párese sobre la Wii balance board teniendo en cuenta que para cerrar la aplicación se debe pulsar su botón principal.

Estado:

Tape el compartimento de baterías.



(a) Fase 1.

Estado:

Espere...



(b) Fase 2.

Estado:

Ajuste terminado.



(c) Fase 3.

Estado:

Ponga la Wii balance board en el suelo (como se muestra en la figura).



(d) Finalizado.

FIGURA A.8. Ajuste automático de la plataforma Wii Balance Board.

- **Nota:** Tenga en cuenta que los reportes y gráficas se almacenan en la carpeta donde este ubicado el programa.

Estado:

Párese sobre la Wii balance board.



(a) Fase 1.

Estado:

Presione el botón principal de la Wii balance board para terminar (cuando lo desee).



(b) Fase 2.



(c) Fase 3.

FIGURA A.9. Uso de la plataforma Wii Balance Board.

Código de la aplicación

```

1 from tkinter import *
2 from tkinter import ttk
3 import datetime
4 import bluetooth
5 import time
6 import matplotlib.pyplot as plt
7 import matplotlib.animation as animation
8 import numpy as np
9 import pandas as pd
10 from drawnow import *
11
12 #Modo interactivo de la libreria matplotlib
13 plt.ion()
14 # Parametros de la Wii Balance Board para acceder a los registros
15 # Estos valores estan en el manual técnico
16 CONTINUOUS_REPORTING = b'\x04'
17 COMMAND_LIGHT = b'\x11'
18 COMMAND_REPORTING = b'\x12'
19 COMMAND_REQUEST_STATUS = b'\x15'
20 COMMAND_REGISTER = b'\x16'
21 COMMAND_READ_REGISTER = b'\x17'
22 INPUT_STATUS = b'\x20'
23 INPUT_READ_DATA = b'\x21'
24 EXTENSION_8BYTES = b'\x32'
25 BUTTON_DOWN_MASK = 0x08
26 LED1_MASK = 0x10
27 BATTERY_MAX = 200.0
28 TOP_RIGHT = 0
29 BOTTOM_RIGHT = 1
30 TOP_LEFT = 2
31 BOTTOM_LEFT = 3
32 BLUETOOTH_NAME = "Nintendo RVL-WBC-01"
33 # Wiiboard parametros de muestreo
34 N_Muestras = 50
35 N_LOOP = 10
36 T_SLEEP = 2
37
38 global estado_galgas
39 #Funcion para convertir de byte a entero
40 def bytes_to_int(bytes):
41     resultado = 0
42     for b in bytes:
43         resultado = (resultado*256)+int(b)
44     return resultado
45
46 #Funcion para publicar sobre la etiqueta en tiempo real
47 def publicar(ventana,texto):
48     ventana.textol.set(texto)
49     raiz.update()
50
51 #Funcion para ver dispositivos bluetooth
52 def describir(ventana,duration=5, prefix=BLUETOOTH_NAME):
53     publicar(ventana,"Pulse el botón SYNC de la WII balance board (Detectando
54     dispositivos espere "+str(duration)+" segundos).\n")
55     devices = bluetooth.discover_devices(duration=duration, lookup_names=True)
56     return [address for address, name in devices if name.startswith(prefix)]
57
58 #Funciones basicas de la Tabla
59 class WiiTabla:
60     def __init__(Nintendo, ventana, direccion=None):
61         Nintendo.controlsocket = bluetooth.BluetoothSocket(bluetooth.L2CAP)
62         Nintendo.receivesocket = bluetooth.BluetoothSocket(bluetooth.L2CAP)
63         Nintendo.calibracion = [[1e4]*4]*3

```

```

64     Nintendo.estado_calibracion = False
65     Nintendo.running = True
66     Nintendo.conta=0
67     Nintendo.valor_actual=0
68     Nintendo.wl=[]
69     Nintendo.xl=[]
70     Nintendo.yl=[]
71     Nintendo.zl=[]
72     Nintendo.Peso=[]
73     Nintendo.ejex=[]
74     Nintendo.ejey=[]
75     if direccion is not None:
76         Nintendo.conectar(direccion,ventana)
77     #Funcion para establecer la conexion con la Wii Tabla
78     def conectar(Nintendo, direccion, ventana):
79         publicar(ventana,("Conectado a: "+BLUETOOTH_NAME+"\n"))
80         Nintendo.controlsocket.connect((direccion, 0x11))
81         Nintendo.receivesocket.connect((direccion, 0x13))
82         Nintendo.Luz(1)
83         ventana.img = PhotoImage(file="5.png")
84         ventana.mostrarestado2.create_image(72, 60, image = ventana.img)
85         time.sleep(5)
86         publicar(ventana,"Tape el compartimento de baterias.\n")
87         ventana.img = PhotoImage(file="3.png")
88         ventana.mostrarestado2.create_image(72, 60, image = ventana.img)
89         time.sleep(5)
90         publicar(ventana,"Ponga la WII balance board en el suelo (como se
muestra en la figura).\n")
91         time.sleep(5)
92         publicar(ventana,"Espere...\n")
93         Nintendo.Calibrar()
94         Nintendo.Estado()
95         Nintendo.Luz(0)
96         time.sleep(2)
97         publicar(ventana,"Ajuste terminado.\n")
98         time.sleep(2)
99         publicar(ventana,"Parese sobre la WII balance board.\n")
100        ventana.img = PhotoImage(file="4.png")
101        ventana.mostrarestado2.create_image(72, 60, image = ventana.img)
102        time.sleep(5)
103        publicar(ventana,"Presione el botón principal de la WII balance board
para terminar (cuando lo desee).\n")
104        time.sleep(5)
105        #Funcion para enviar
106        def Enviar(Nintendo, *datos):
107            Nintendo.controlsocket.send(b'\x52'+b''.join(datos))
108        #Funcion para ajustar el valor de cero de la balance board
109        def Calibrar(Nintendo):
110            Nintendo.Enviar(COMMAND_READ_REGISTER, b"\x04\xA4\x00\x24\x00\x18")
111            time.sleep(2)
112            Nintendo.Enviar(COMMAND_REGISTER, b"\x04\xA4\x00\x40\x00")
113            Nintendo.estado_calibracion = True
114        #Terminador para cerrar la comunicacion y eliminar el envio continuo
115        def Estado(Nintendo):
116            Nintendo.Enviar(COMMAND_REQUEST_STATUS, b'\x00')
117        #Funcion para fijar estado del LED
118        def Luz(Nintendo, LED=True):
119            Nintendo.Enviar(COMMAND_LIGHT, b'\x10' if LED else b'\x00')
120        #Funcion de reporte continuo para enviar registros de la balance board
121        def Reporte(Nintendo):
122            Nintendo.Enviar(COMMAND_REPORTING, CONTINUOUS_REPORTING,
EXTENSION_8BYTES)
123        #Funcion para leer de la Tabla
124        def loop(Nintendo, ventana):

```

```

125     w=[]
126     x=[]
127     y=[]
128     z=[]
129     p=[]
130     while Nintendo.running and Nintendo.receive_socket and ventana.salirapp:
131         datos = Nintendo.receive_socket.recv(25)
132         input_type = datos[1]
133         #Opcion para determinar estado del LED y la bateria
134         if input_type == bytes_to_int(INPUT_STATUS):
135             if(Nintendo.conta>N_Muestras):
136                 Nintendo.conta=0
137                 Nintendo.Estado()
138                 Nintendo.w1.append(np.max(w))
139                 Nintendo.x1.append(np.max(x))
140                 Nintendo.y1.append(np.max(y))
141                 Nintendo.z1.append(np.max(z))
142                 Nintendo.Peso.append(np.max(p))
143                 Nintendo.COP()
144                 Nintendo.ExportarExcel()
145                 w=[]
146                 x=[]
147                 y=[]
148                 z=[]
149                 p=[]
150                 bateria = float(bytes_to_int(datos[7:9])/BATTERY_MAX)*10.0
151                 #Detectar estado del LED
152                 led = datos[4] & LED1_MASK == LED1_MASK
153                 Nintendo.Reporte()
154                 #Leer cadena de calibración
155                 elif input_type == bytes_to_int(INPUT_READ_DATA):
156                     if Nintendo.estado_calibracion:
157                         Nintendo.Asignar_calibracion(datos)
158                     #Leer Estado de las celdas de carga
159                 elif input_type == bytes_to_int(EXTENSION_8BYTES):
160                     Nintendo.Sensores(datos[2:12])
161                     w.append(estados_galgas[0])
162                     x.append(estados_galgas[1])
163                     y.append(estados_galgas[2])
164                     z.append(estados_galgas[3])
165
166                 promedio_galgas=estados_galgas[0]+estados_galgas[1]+estados_galgas[2]+estados_galgas[3]
167                 p.append(promedio_galgas)
168                 print(estados_galgas)
169                 Nintendo.conta+=1
170                 Nintendo.Estado()
171                 if(estados_galgas[4]==8):
172                     Nintendo.running=False
173                     raiz.destroy()
174                     plt.close()
175                     Nintendo.cerrar
176
177                 #Funcion para asignar valores iniciales al programa para estimar el valor de
178                 las galgas
179                 def Asignar_calibracion(Nintendo, bytes):
180                     length = int((bytes[4]/16)+1)
181                     #Define la estructura en funcion de d, es decir, a medida que se ejecuta
182                     el bucle cal convierte
183                     # los ocho bytes en un numero de referencia para la wii board
184                     bytes=bytes[7:(7+length)]
185                     cal = lambda d: [bytes_to_int(d[j:j+2]) for j in [0, 2, 4, 6]]
186                     #Primer paquete de calibracion
187                     if length == 16:
188                         Nintendo.calibracion = [cal(bytes[0:8]), cal(bytes[8:16]), [1e4]*4]
189                     #Segundo paquete de calibracion

```

```

186         elif length < 16:
187             Nintendo.calibracion[2] = cal(bytes[0:8])
188             Nintendo.estado_calibracion = False
189         #Funcion para leer el valor de las galgas y asignarlo a la variable
global
190     def Sensores(Nintendo, bytes):
191         Nintendo.Luz(1)
192         global estado_galgas
193         estado_galgas=[]
194         estado = bytes_to_int(bytes[0:2])
195         #Conversiones de la cadena de la wii board
196         galgas=bytes[2:10]
197         rawTR = bytes_to_int(galgas[0:2])
198         rawBR = bytes_to_int(galgas[2:4])
199         rawTL = bytes_to_int(galgas[4:6])
200         rawBL = bytes_to_int(galgas[6:8])
201         topLeft = Nintendo.calcMass(rawTL, TOP_LEFT)
202         topRight = Nintendo.calcMass(rawTR, TOP_RIGHT)
203         bottomLeft = Nintendo.calcMass(rawBL, BOTTOM_LEFT)
204         bottomRight = Nintendo.calcMass(rawBR, BOTTOM_RIGHT)
205         #Vector para conocer el estado de las galgas
206         estado_galgas.append(topLeft)
207         estado_galgas.append(topRight)
208         estado_galgas.append(bottomLeft)
209         estado_galgas.append(bottomRight)
210         estado_galgas.append(estado)
211         Nintendo.Luz(0)
212     #Esta funcion calcula la presion por sensor de acuerdo con la hoja tecnica
213     def calcMass(Nintendo, raw, pos):
214         if raw < Nintendo.calibracion[0][pos]:
215             return 0.0
216         elif raw < Nintendo.calibracion[1][pos]:
217             return 17 * ((raw - Nintendo.calibracion[0][pos]) /
218                 float((Nintendo.calibracion[1][pos] -
219                     Nintendo.calibracion[0][pos])))
220         else: # if raw >= self.calibracion[1][pos]:
221             return 17 + 17 * ((raw - Nintendo.calibracion[1][pos]) /
222                 float((Nintendo.calibracion[2][pos] -
223                     Nintendo.calibracion[1][pos])))
224     #Funcion para graficar COP
225     def fuerzatraccion(Nintendo):
226         a=Nintendo.w1[Nintendo.valor_actual]#SI
227         b=Nintendo.x1[Nintendo.valor_actual]#SD
228         c=Nintendo.y1[Nintendo.valor_actual]#II
229         d=Nintendo.z1[Nintendo.valor_actual]#ID
230         Nintendo.valor_actual+=1
231         x1=a*np.cos((+135.0*np.pi)/180.0)
232         y1=a*np.sin((+135.0*np.pi)/180.0)
233         x2=b*np.cos((+45.0*np.pi)/180.0)
234         y2=b*np.sin((+45.0*np.pi)/180.0)
235         x3=c*np.cos((-135.0*np.pi)/180.0)
236         y3=c*np.sin((-135.0*np.pi)/180.0)
237         x4=d*np.cos((-45.0*np.pi)/180.0)
238         y4=d*np.sin((-45.0*np.pi)/180.0)
239         e=x1+x2+x3+x4
240         f=y1+y2+y3+y4
241         Nintendo.ejex.append(e)
242         Nintendo.ejey.append(f)
243         return [e, f, a, b, c, d]
244
245     def figura(Nintendo):
246         #Comportamiento de los sensores
247         plt.pause(.001)
248         plt.subplot(2,2,1)

```

```

249     plt.plot(Nintendo.w1,'k--',linewidth = 2, label = 'Superior
izquierdo(w)')
250     plt.grid(True)
251     plt.plot(Nintendo.x1,'r*-',linewidth = 2, label = 'Superior Derecho(x)')
252     plt.plot(Nintendo.y1,'g',linewidth = 2, label = 'Inferior izquierdo(y)')
253     plt.plot(Nintendo.z1,'b',linewidth = 2, label = 'Inferior derecho(z)')
254     plt.xlabel('Muestras[n]')
255     plt.ylabel('Sensores \n (carga por sensor [kg])')
256     plt.legend(loc=2)
257     #Comportamiento del peso
258     plt.subplot(2,2,2)
259     plt.plot(Nintendo.Peso,'k',linewidth = 2)
260     plt.grid(True)
261     plt.xlabel('Muestras[n]')
262     plt.ylabel('Peso usuario \n (kilogramos [kg])')
263     #Ley de atraccion
264     plt.subplot(2,2,3)
265     polos=Nintendo.fuerzatraccion()
266     plt.plot(polos[0],polos[1],'k*',linewidth = 2, label = 'Centro de masa
[cm]')
267     plt.grid(True)
268     plt.ylabel('Movimiento de la particula')
269     plt.xlim(-40,40)
270     plt.ylim(-40,40)
271     plt.legend(loc=2)
272     plt.figure(num=1, figsize=(1, 1))
273     #Campo potencial
274     plt.subplot(2,2,4)
275     c1=plt.Circle((-40, +40), int(polos[2]), color='b', fill=True) #SI
276     c2=plt.Circle(+40, +40), int(polos[3]), color='b', fill=True) #SD
277     c3=plt.Circle((-40, -40), int(polos[4]), color='b', fill=True) #II
278     c4=plt.Circle(+40, -40), int(polos[5]), color='b', fill=True) #ID
279     plt.gcf().gca().add_artist(c1)
280     plt.gcf().gca().add_artist(c2)
281     plt.gcf().gca().add_artist(c3)
282     plt.gcf().gca().add_artist(c4)
283     plt.grid(True)
284     plt.ylabel('Campo potencial por sensor')
285     plt.xlim(-40,40)
286     plt.ylim(-40,40)
287     plt.figure(num=1, figsize=(1, 1))
288
289     def COP(Nintendo):
290         drawnow(Nintendo.figura)
291
292     #Funcion para exportar datos a excel
293     def ExportarExcel(Nintendo):
294         datosaexportar = []
295         for i in range(len(Nintendo.w1)):
296             datosaexportar.append([])
297             for j in range(7):
298                 if(j==0):
299                     datosaexportar[i].append(Nintendo.w1[i])
300                 elif(j==1):
301                     datosaexportar[i].append(Nintendo.x1[i])
302                 elif(j==2):
303                     datosaexportar[i].append(Nintendo.y1[i])
304                 elif(j==3):
305                     datosaexportar[i].append(Nintendo.z1[i])
306                 elif(j==4):
307                     datosaexportar[i].append(Nintendo.Peso[i])
308                 elif(j==5):
309                     datosaexportar[i].append(Nintendo.ejex[i])
310                 elif(j==6):

```

```

311         datosaexportar[i].append(Nintendo.ejey[i])
312     df = pd.DataFrame(datosaexportar, columns = ['Galga superior izquierda',
313                                               'Galga superior derecha',
314                                               'Galga inferior izquierda',
315                                               'Galga inferior derecha',
316                                               'Peso',
317                                               'Centro de masa (x)',
318                                               'Centro de masa (y)'])
319     df.to_excel('Centrodemasa.xlsx', sheet_name='Galgas')
320
321 #Funciones para terminar la clase y cerrar el puerto
322 def cerrar(Nintendo):
323     Nintendo.running = False
324     if Nintendo.receivesocket: Nintendo.receivesocket.close()
325     if Nintendo.controlsocket: Nintendo.controlsocket.close()
326 def __del__(Nintendo):
327     Nintendo.cerrar()
328 def __enter__(Nintendo):
329     return self
330 def __exit__(self, exc_type, exc_val, exc_tb):
331     Nintendo.cerrar()
332
333 #Clase auxiliar
334 class WiiTabla2(WiiTabla):
335     def __init__(Nintendo, ventana, direccion=None):
336         WiiTabla.__init__(Nintendo, ventana, direccion)
337         WiiTabla.loop(Nintendo,ventana)
338
339 #Interfaz Grafica de usuario
340 class Aplicacion():
341     def __init__(ventana, maestro):
342         ventana.master=maestro
343         maestro.configure(bg = 'white')
344         maestro.title('Interfaz gráfica de usuario (GUI)')
345         ventana.texto1 = StringVar()
346         ventana.titulo = ttk.Label(maestro,
347                                   text = "Interfaz para Nintendo WII Balance
348 Board.\n\n",
349                                   foreground = "blue",
350                                   background = "white",
351                                   anchor = "c",
352                                   font = "Verdana 20 bold").pack()
353         ventana.titulo = ttk.Label(maestro,
354                                   text = "Autores:",
355                                   foreground = "black",
356                                   background = "white",
357                                   anchor = "c",
358                                   font = "Verdana 15 bold").pack()
359         ventana.titulo = ttk.Label(maestro,
360                                   text = "Jenny Katherine Torres Zambrano \n
361 Julian Oswaldo Perez De la Cuadra\n\n",
362                                   foreground = "black",
363                                   background = "white",
364                                   anchor = "c",
365                                   font = "Verdana 15").pack()
366         ventana.titulo = ttk.Label(maestro,
367                                   text = "Directora:",
368                                   foreground = "black",
369                                   background = "white",
370                                   anchor = "c",
371                                   font = "Verdana 15 bold").pack()
372         ventana.titulo = ttk.Label(maestro,
373                                   text = "Ing. Lely Adriana Luengas Contreras
374 PhD.\n\n\n\n",

```

```

372         foreground = "black",
373         background = "white",
374         anchor = "c",
375         font = "Verdana 15").pack()
376     ventana.titulo = ttk.Label(maestro,
377                               text = "Seleccione una opción para comenzar.
\n",
378                               foreground = "black",
379                               background = "white",
380                               anchor = "c",
381                               font = "Verdana 15").pack()
382     ttk.Style().configure("GB.TButton", foreground='black',
background='green')
383     ventana.boton_blue=ttk.Button(maestro,
384                                  text='Iniciar',
385                                  style="GB.TButton",
386                                  command=ventana.iniciar_bluetooth).pack()
387     ventana.titulo = ttk.Label(maestro,
388                               text = "\n",
389                               font = "Verdana 5 bold").pack()
390     ttk.Style().configure("RB.TButton", foreground='black', background='red')
391     ventana.boton_salir=ttk.Button(maestro,
392                                   text='Salir',
393                                   style="RB.TButton",
394                                   command=ventana.salirapp).pack()
395     ventana.titulo = ttk.Label(maestro,
396                               text = "\n Estado:",
397                               foreground = "black",
398                               background = "white",
399                               anchor = "c",
400                               font = "Verdana 15 bold").pack()
401
402     ventana.mostrarestado1 = ttk.Label(maestro,
403                                       textvariable = ventana.texto1,
404                                       foreground = "red",
405                                       background = "white",
406                                       anchor = "c",
407                                       font = "Verdana 10").pack()
408
409     ventana.img = PhotoImage(file="1.png")
410     ventana.mostrarestado2 = Canvas(maestro, width=140, height=120,
highlightthickness=0)
411     ventana.mostrarestado2.create_image(72, 60, image = ventana.img)
412     ventana.mostrarestado2.pack()
413
414     ventana.texto1.set("Para iniciar pulse el botón verde (Iniciar).
\n")
415     maestro.mainloop()
416
417     def iniciar_bluetooth(ventana):
418         ventana.img = PhotoImage(file="2.png")
419         ventana.mostrarestado2.create_image(72, 60, image = ventana.img)
420         wiiboards=descubrir(ventana)
421         if not wiiboards:
422             ventana.texto1.set("Verifique el estado de las baterías y pulse el
boton verde (Iniciar) nuevamente.\n")
423         ventana.img = PhotoImage(file="5.png")
424         ventana.mostrarestado2.create_image(72, 60, image = ventana.img)
425         else:
426             direccion=wiiboards[0]
427             ventana.texto1.set("Dirección detectada:"+str(direccion)+"\n")
428             ventana.img = PhotoImage(file="1.png")
429             ventana.mostrarestado2.create_image(72, 60, image = ventana.img)
430             WiiTabla2(ventana, direccion)

```



```
431     def salirapp(ventana):
432         raiz.destroy()
433
434     #Funcion principal
435     if __name__ == '__main__':
436         raiz = Tk()
437         Aplicacion(raiz)
```

Bibliografía

- [1] Analisis de variables multiples - Revista Chilena de Anestesia.
- [2] Análisis digital huella plantar - FootMedicalChile.
- [3] Assessment of hemiplegic gait using the wisconsin gait scale - turani - 2004 - scandinavian journal of caring sciences - wiley online library.
- [4] Botulinum toxin injection in patients with hereditary spastic paraparesis - Rousseaux - 2007 - European Journal of Neurology - Wiley Online Library.
- [5] EMG (Electromyogram) (for Parents).
- [6] Fisicaqueame2.
- [7] Gait Analysis: Normal and Pathological Function : Journal of Pediatric Orthopaedics.
- [8] Indl Instrumentation & Control 3e - Singh - Google Libros.
- [9] Industrial Instrumentation - K Krishnaswamy - Google Libros.
- [10] Instrumentation, Measurement And Analysis - B. C. Nakra, K. K. Chaudhry - Google Libros.
- [11] Laboratorios de análisis del movimiento, análisis de la marcha.
- [12] Leardini: Designing experimental protocols for routine... - Google Academico.
- [13] The neuromatics laboratory, colorado university, 2017.
- [14] Plataformas de presion, software analisis de presiones.
- [15] Servicios de CalibraciÃn - CalTechnix de Mexico, SA de CV - Laboratorio de calibracion acreditado.
- [16] Wii Balance Board Teardown, Nov. 2016.
- [17] Como Analizar Datos: La guia paso a paso para Aprovechar tus Datos, July 2017.
- [18] Videografia digital, Jan. 2017.
- [19] AMJADI, M., KIM, M. S., AND PARK, I. Flexible and sensitive foot pad for sole distributed force detection. In *14th IEEE International Conference on Nanotechnology* (Aug 2014), pp. 764–767.

-
- [20] BENEDETTI, M. G., MANCA, M., SICARI, M., FERRARESI, G., CASADIO, G., BUGANÈ, F., AND LEARDINI, A. Gait measures in patients with and without afo for equinus varus/drop foot. In *2011 IEEE International Symposium on Medical Measurements and Applications* (May 2011), pp. 591–592.
- [21] BENOIT, D. L., RAMSEY, D. K., LAMONTAGNE, M., XU, L., WRETENBERG, P., AND RENSTRÅM, P. Effect of skin movement artifact on knee kinematics during gait and cutting motions measured in vivo. *Gait & Posture* *24*, 2 (Oct. 2006), 152–164.
- [22] BOUKHENOUS, S., AND ATTARI, M. A vertical jumping performance with and without arms swing by using a dynamometric platform. In *International Workshop on Systems, Signal Processing and their Applications, WOSSPA* (May 2011), pp. 17–20.
- [23] BRINDZA, J., SZWEDA, J., LIAO, Q., JIANG, Y., AND STRIEGEL, A. Wiilab: Bringing together the nintendo wiimote and matlab. In *2009 39th IEEE Frontiers in Education Conference* (Oct 2009), pp. 1–6.
- [24] BRUNT, D., GREENBERG, B., WANKADIA, S., TRIMBLE, M. A., AND SHECHTMAN, O. The effect of foot placement on sit to stand in healthy young subjects and patients with hemiplegia. *Archives of Physical Medicine and Rehabilitation* *83*, 7 (2002), 924 – 929.
- [25] COMPLUTENSE, U. *Analisis de la marcha humana con plataformas dinamométricas: influencia del transporte de carga*. <http://purl.org/dc/dcmitype/Text>, Universidad Complutense de Madrid, 2002.
- [26] DIAZ, C. A., TORRES, A., RAMIREZ, J. I., GARCIA, L. F., AND ALVAREZ, N. DESCRIPCION DE UN DISPOSITIVO DESTINADO AL ANÁLISIS DE LA MARCHA EN DOS DIMENSIONES, CineMED. *Revista EIA*, 5 (June 2006), 85–92.
- [27] DU, L., ZHU, X., AND ZHE, J. An inductive sensor for real-time measurement of plantar normal and shear forces distribution. *IEEE Transactions on Biomedical Engineering* *62*, 5 (May 2015), 1316–1323.
- [28] DUHANEY, J., KHOSHGOFTAAR, T. M., AND SLOAN, J. C. Feature selection on dynamometer data for reliability analysis. In *2011 IEEE 23rd International Conference on Tools with Artificial Intelligence* (Nov 2011), pp. 1012–1019.
- [29] FUJIMOTO, Y., AND KAWANURA, A. Biped walking control with optimal foot force distribution by quadratic programming. In *Proceedings of IEEE/ASME International Conference on Advanced Intelligent Mechatronics* (June 1997), pp. 108–.
- [30] GU, G. M., PARK, K., KIM, E. J., LEE, D. Y., AND KIM, J. Foot pronation monitoring using wireless biaxial force sensing system. In *2015 IEEE International Conference on Rehabilitation Robotics (ICORR)* (Aug 2015), pp. 19–24.
- [31] GUERRERO, J. Técnicas de procesamiento de imágenes estereoscópicas.
- [32] HUANG, Q., YANG, J., YU, Z., XU, W., LI, J., AND LI, K. Measurement of human walking and generation of humanoid walking pattern. In *2007 IEEE International Conference on Robotics and Biomimetics (ROBIO)* (Dec 2007), pp. 127–132.
- [33] IERSEL, M. B. V., HAITSMAN, A., RIKKERT, M. G. M. O., AND BENRAAD, C. E. M. Quantitative Gait Analysis to Detect Gait Disorders in Geriatric Patients with Depression. *Journal of the American Geriatrics Society* *53*, 8 (2005), 1441–1442.

-
- [34] IKEMOTO, Y., AND YU, W. The roles of cpg phase modulation and reflexive muscular patterns in balance recovery reflexive responses to perturbation during walking - a simulation study. In *2007 29th Annual International Conference of the IEEE Engineering in Medicine and Biology Society* (Aug 2007), pp. 2389–2392.
- [35] JIANG, Y., WANG, S., TAN, R., ISHIDA, K., ANDO, T., AND FUJIE, M. G. Study of activation in motor cortex during mental imagery of walking using fmri. In *The 2011 IEEE/ICME International Conference on Complex Medical Engineering* (May 2011), pp. 637–640.
- [36] KANE, J. W., AND STERNHEIM, M. M. *Física*. Reverte, 1989. Google-Books-ID: lj5kLw2uxGIC.
- [37] KIM, G. . Development of 6-axis force/moment sensor for a humanoid robot’s foot. *IET Science, Measurement Technology* 2, 3 (May 2008), 122–133.
- [38] LI, K. W., AND WEN, H. Friction between foot and floor under barefoot conditions: A pilot study. In *2013 IEEE International Conference on Industrial Engineering and Engineering Management* (Dec 2013), pp. 1651–1655.
- [39] LIM, H., AND YOSHIDA, T. Online walking pattern generation for biped walking. In *ICCAS 2010* (Oct 2010), pp. 1816–1821.
- [40] LIND, R. F., LOVE, L. J., ROWE, J. C., AND PIN, F. G. Multi-axis foot reaction force/torque sensor for biomedical applications. In *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems* (Oct 2009), pp. 2575–2579.
- [41] LUCA VISMARA, M. R. Clinical implications of gait analysis in the rehabilitation of adult patients with "Prader-Willi" Syndrome: a cross-sectional comparative study ("Prader-Willi" Syndrome vs matched obese patients and healthy subjects) | Journal of NeuroEngineering and Rehabilitation | Full Text. *Journal of NeuroEngineering and Rehabilitation* (2007), 4–14.
- [42] LUO, R. C., PU, Y. H., CHEN, C. H., CHANG, J. R., AND LI, C. Y. Design and implementation of humanoid biped walking robot mechanism towards natural walking. In *2011 IEEE International Conference on Robotics and Biomimetics* (Dec 2011), pp. 1165–1170.
- [43] MA, R., LI, J., JIN, S., GUO, S., HASHIMOTO, K., AND DAI, S. A speed-independent feedback index for walking pattern recognition for a walking assistive robotic suit. In *2017 IEEE International Conference on Cybernetics and Intelligent Systems (CIS) and IEEE Conference on Robotics, Automation and Mechatronics (RAM)* (Nov 2017), pp. 514–517.
- [44] MARTINEZ, A. R., AND PODOLOGIA, D. Visión crítica de la biomecánica y tecnología aplicadas a la podología. 48.
- [45] MCNAIR, T. P. S. The penn state mcnair.
- [46] MOCK, M., AND SWEETING, K. Gait and posture - assessment in general practice. *Australian Family Physician* 36, 6 (June 2007), 398.
- [47] OBERG, T., KARSZNIA, P. A., AND OBERG, P. K. *Basic gait parameters: reference data for normal subjects, 10–79 years of*. 1993.

-
- [48] ONARAN, E. M., ERTAŞ, G., AND KÜÇÜK, H. A device to measure joint angles and foot force for lower extremity force distribution computations. In *2013 8th International Conference on Electrical and Electronics Engineering (ELECO)* (Nov 2013), pp. 414–417.
- [49] PAPADONIKOLAKIS, A., VEKRIS, M., KOROMPILIAS, A., KOSTAS, J., RISTANIS, S., AND SOUCACOS, P. Botulinum A toxin for treatment of lower limb spasticity in cerebral palsy Gait analysis in 49 patients. *Acta Orthopaedica Scandinavica* 74, 6 (Jan. 2003), 749–755.
- [50] PARK, J., AND KIM, J. Development of foot devices for virtual reality. In *2018 18th International Conference on Control, Automation and Systems (ICCAS)* (Oct 2018), pp. 1784–1787.
- [51] PATRICK, J. H. The case for gait analysis as part of the management of incomplete spinal cord injury. *Spinal Cord* 41, 9 (Sept. 2003), 479–482.
- [52] PYTHON. Welcome to Python.org.
- [53] SAID, C. M., GOLDIE, P. A., PATLA, A. E., AND SPARROW, W. A. Effect of stroke on step characteristics of obstacle crossing. *Archives of Physical Medicine and Rehabilitation* 82, 12 (2001), 1712 – 1719.
- [54] SALAMI, F., DIPAOLA, M., MARZEGAN, A., ISAIAS, I. U., AND FRIGO, C. A. Walking efficiency assessment through the analysis of mechanical energy and energy recovery index. In *2013 E-Health and Bioengineering Conference (EHB)* (Nov 2013), pp. 1–3.
- [55] SHI, P., AND YU, H. Walking patterns of knee and ankle joints during level walking and uphill walking. In *2016 IEEE International Conference on Signal and Image Processing (ICSIP)* (Aug 2016), pp. 56–59.
- [56] SOLE, A. C. *Instrumentación industrial*. Marcombo, 2005. Google-Books-ID: cV6ZOqQ0ywMC.
- [57] SUPORT, N. Wii balance board manual. *Nintendo of America INC.* (2006), 13–25.
- [58] TENORE, N., FORTUGNO, F., VIOLA, F., GALLI, M., AND GIAQUINTO, S. Gait analysis as a reliable tool for rehabilitation of chronic hemiplegic patients. *Clinical and Experimental Hypertension* 28, 3-4 (Jan. 2006), 349–355.
- [59] THOMPSON, T. J., KUMAR, C. B., AND KLINE, P. J. *Bluetooth Application Programming with the Java APIs Essentials Edition*. Morgan Kaufmann, Apr. 2008. Google-Books-ID: 0dsIPdbr4o8C.
- [60] VAZQUEZ, S. C. PLATAFORMAS DINAMOMETRICAS. APLICACIONES. 18.
- [61] VILLA MORENO, A., GUTIERREZ GUTIERREZ, E., AND PEREZ MORENO, J. C. CONSIDERACIONES PARA EL ANALISIS DE LA MARCHA HUMANA. TECNICAS DE VIDEOGRAMETRIA, ELECTROMIOGRAFIA Y DINAMOMETRIA. *Revista Ingenieria Biomedica* 2, 3 (June 2008), 16–26.
- [62] WANG, J., LIN, C., YANG, Y. C., AND HO, Y. Walking pattern classification and walking distance estimation algorithms using gait phase information. *IEEE Transactions on Biomedical Engineering* 59, 10 (Oct 2012), 2884–2892.

-
- [63] WATELAIN, E., FROGER, ROUSSEAU, M., LENSEL, G., BARBIER, F., LEPOUTRE, F.-X., AND THEVENON, A. Variability of video-based clinical gait analysis in hemiplegia as performed by practitioners in diverse specialties. *Journal of Rehabilitation Medicine* 37, 5 (Sept. 2005), 317–324.
- [64] WEN, H. Friction between foot and floor under barefoot conditions: A pilot study. In *2013 IEEE International Conference on Industrial Engineering and Engineering Management* (Dec 2013), pp. 1651–1655.
- [65] XUE-FENG, S., JIAN-HUA, W., ZHENG-LI, Z., AND DENG-HUA, Y. Restoration analyses of yongding river in beijing based on thrice water balance theory. In *2011 International Conference on Remote Sensing, Environment and Transportation Engineering* (June 2011), pp. 4373–4376.
- [66] YUAN, C., WANG, W., HAN, C., AND CHOI, H. A three degree of freedom force/torque sensor to measure foot forces. In *2012 12th International Conference on Control, Automation and Systems* (Oct 2012), pp. 2028–2032.